TRANSACTIONS OF THE
AMERICAN MATHEMATICAL SOCIETY
Volume 260, Number 2, August 1980

ON THE COMPUTATIONAL COMPLEXITY OF DETERMINING THE SOLVABILITY OR UNSOLVABILITY OF THE EQUATION $X^2 - DY^2 = -1$

BY

J. C. LAGARIAS

ABSTRACT. The problem of characterizing those D for which the Diophantine equation $X^2 - DY^2 = -1$ is solvable has been studied for two hundred years. This paper considers this problem from the viewpoint of determining the computational complexity of recognizing such D. For a given D, one can decide the solvability or unsolvability of $X^2 - DY^2 = -1$ using the ordinary continued fraction expansion of $\sqrt{D}$, but for certain D this requires more than $\frac{1}{3}\sqrt{D}\ (\log D)^{-1}$ computational operations. This paper presents a new algorithm for answering this question and proves that this algorithm always runs to completion in $O(D^{1/4+\epsilon})$ bit operations. If the input to this algorithm includes a complete prime factorization of D and a quadratic nonresidue n_i for each prime p_i dividing D, then this algorithm is guaranteed to run to completion in $O((\log D)^5(\log\log D)(\log\log\log D))$ bit operations. This algorithm is based on an algorithm that finds a basis of forms for the 2-Sylow subgroup of the class group of binary quadratic forms of determinant D.

1. Introduction. The problem of determining those D for which the equation

$$X^2 - DY^2 = -1 \tag{1.1}$$

(sometimes called the *non-Pellian equation*) is solvable in integers (X, Y) has a long history. It is well known that for any positive nonsquare D the solvability or unsolvability of (1.1) can be determined by expanding $\sqrt{D}$ as an ordinary continued fraction

$$\sqrt{D} = [a_0, \overline{a_1, \ldots, a_N}] \tag{1.2}$$

where the portion $[a_1, \ldots, a_N]$ is periodic. Then (1.1) is solvable or not according to whether N is odd or even. If N is odd, then

$$X_0/Y_0 = [a_0, a_1, \ldots, a_{N-1}] \tag{1.3}$$

is the minimal positive solution of (1.1). (These facts are usually stated for squarefree D, but are true for a general nonsquare D.)

A second approach to this problem involves using generalized residue symbol criteria derived from D to determine conditions on D which guarantee that (1.1) is solvable or unsolvable. This approach was initiated by Legendre in 1785. He proved that if D is a prime $p \equiv 1 \pmod 4$, then (1.1) is solvable, while if a $p \equiv 3$

Received by the editors July 3, 1979.

AMS (MOS) subject classifications (1970). Primary 10B05, 10C05, 68A20; Secondary 02E10, 15A03, 68A10.

Key words and phrases. Computational complexity, binary quadratic forms, form class group, composition of forms, Pell's equation, Diophantine equation.

0002-9947/80/0000-0360/$07.00

(mod 4) divides the squarefree part of D then (1.1) is unsolvable. Dirichlet [8] observed that if $D = pq$ with $p \equiv q \equiv 1 \pmod 4$ and $(p/q)_4 = (q/p)_4 = -1$ then (1.1) is solvable. For $D = p_1 \cdots p_N$ Tano [37] obtained quadratic residue criteria among the p_i which when they held would guarantee (1.1) is solvable. Scholz [32] applied methods of class field theory and obtained (among other results) that in the case $D = pq$ with $p \equiv q \equiv 1 \pmod 4$ that (1.1) is unsolvable when $(p/q)_4 \neq (q/p)_4$, but in the case $(p/q)_4 = (q/p)_4 = 1$ the equation (1.1) is sometimes solvable and sometimes not. Both Scholz [32] and Redei [31] observed that these residue symbol criteria were related to the structure of the 2-Sylow subgroup of an appropriate ring class group of $\mathbf{Q}(\sqrt{D})$. Redei [30], [31] introduced a "conditional Artin symbol" defined in terms of generators of certain class fields, by means of which he gave a set of necessary and sufficient conditions for (1.1) to be solvable. Narkiewicz [27, p. 463] treats the problem of determining those D for which (1.1) is solvable as still open, presumably due to the nonexplicit character of Redei's conditions. Explicit residue symbol conditions for special types of D are still being found, e.g. Kaplan [17], Pumplün [29].

In considering these two approaches, a natural question to ask is: Does the residue symbol approach provide a simpler characterization of those D for which (1.1) is solvable than that of simply testing each D by the continued fraction algorithm? This paper examines this question from the viewpoint of the worst-case complexity of computing for each D whether (1.1) is solvable or not.

We shall measure computational complexity in terms of *elementary operations*. An elementary operation is a Boolean operation on a single binary bit or pair of bits, or an input or shift of a binary bit. For example, it takes $[\log_2 D] + 1$ elementary operations to load the binary representation of D into a register. In counting elementary operations, we use $O(n)$ to indicate an upper bound of Cn operations, where C is an effectively computable positive constant which does not depend on the input of the algorithm being analyzed, but which may vary at each occurrence of the O-symbol. For ease in bounding operation counts, we establish the convention throughout the sequel that

$$\log N = \begin{cases} \log|N| & \text{if } |N| \geqslant q, \\ 2 & \text{if } |N| < q. \end{cases}$$

What is the worst-case complexity of determining whether (1.1) is solvable using the continued fraction algorithm? This algorithm is very efficient for finding the minimal solution of (1.1) when it exists, but this solution may be very large. For example, when $D = 5^{2k+1}$ with $k = 0, 1, 2, 3, \ldots$ the equation (1.1) is solvable and the minimal solution (X_k, Y_k) satisfies

$$X_k + Y_k\sqrt{5} = (2 + \sqrt{5})^{5^k}. \tag{1.4}$$

Now the partial quotients a_j, the continued fraction expansion of $\sqrt{D}$ in (1.2), can be shown to satisfy

$$0 < a_j < 2\sqrt{D}\,. \tag{1.5}$$

This implies that the continued fraction expansion of $\sqrt{D}$ for $D = 5^{2k+1}$ has a period length exceeding $\frac{1}{3}\sqrt{D}\,(\log D)^{-1}$.[1] Since computing each partial quotient of $\sqrt{D}$ requires at least $\log D$ elementary operations, these examples show the continued fraction algorithm may require at least $\frac{1}{3}\sqrt{D}$ elementary operations in the worst case. (These examples are complemented by a result of Hua **[16]** which implies an upper bound of $O(\sqrt{D}\ \log D)$ for the period length of the continued fraction expansion of $\sqrt{D}$.) On the other hand, these examples do not apply to Narkiewicz's version of the problem, which requires that D be squarefree. The worst-case behavior of the continued fraction algorithm of $\sqrt{D}$ for squarefree D is one of the outstanding problems of number theory, being closely related to determining the size of the class number of the real quadratic field $\mathbf{Q}(\sqrt{D}\,)$. By the Brauer-Siegel Theorem (Lang **[23]**) those $\sqrt{D}$ with long periods are associated to $\mathbf{Q}(\sqrt{D}\,)$ with small class numbers. (See also Yokoi **[41]**.) The empirical evidence convincingly supports the conjecture that there are infinitely many squarefree D for which the length of the period of the continued fraction expansion of $\sqrt{D}$ exceeds $D^{1/2-\varepsilon}$ for any fixed $\varepsilon > 0$ (see Hendy **[14]**).

We now consider the residue symbol approach. The main results of this paper are a new algorithm based on the residue symbol approach and a worst-case complexity analysis establishing that this algorithm is superior to the ordinary continued fraction algorithm (Corollary 1.3). In fact, we establish somewhat more. The residue symbol approach seems to require a complete prime factorization of D.[2] Indeed, most of the known residue symbol criteria are expressed in terms of the prime factors of D. The main complexity result of this paper is that the residue symbol approach can be extended to yield an algorithm determining the solvability of (1.1) whose main bottleneck is finding a factorization of D.

THEOREM 1.1. *There is an algorithm which when given a positive D together with*
(i) *a complete prime factorization of D,*
(ii) *a quadratic nonresidue n_i for each prime p_i dividing D,*
determines whether $X^2 - DY^2 = -1$ is solvable in integers or not, and which always terminates in $O((\log D)^5(\log\log D)(\log\log\log D))$ elementary operations.

The essential feature of this result is that the running time bound is polynomial in the length of the input data. (The input data is of length at least $\log_2 D$ binary bits.)

The hypothesis (ii) concerning quadratic nonresidues can be removed by an appeal to a conditional result of Ankeny **[3]**. This asserts that if the extended Riemann hypothesis (ERH) is true, then for any prime p there is a quadratic nonresidue $n \pmod p$ with $0 < n < C(\log p)^2$ where C is an effectively computable

---

[1] Examples of this kind are apocryphal and were undoubtably known to Dirichlet. We give proofs of these facts in Appendix A, since there do not seem to be readily accessible references.

[2] It is unlikely (but not impossible) that factorization can be avoided. For example, Dirichlet's necessary condition that an odd D have all primes which divide its squarefree part satisfy $p \equiv 1 \pmod 4$ is equivalent to $D = x^2 + y^2$ being solvable in integers. But determining whether or not D has such a representation seems no easier a problem than factoring D.

constant independent of p. We can test in $O((\log p)^3(\log\log p)(\log\log\log p))$ elementary operations whether a given m with $0 < m < p$ is a quadratic residue (mod p) or not, and this yields the following result.

COROLLARY 1.2 (ERH). *There is an algorithm which when given a positive D together with a complete prime factorization of D determines whether $X^2 - DY^2 = -1$ is solvable or not. If the extended Riemann hypothesis is true, this algorithm always terminates in $O((\log D)^5(\log\log D)(\log\log\log D))$ elementary operations.*

We also obtain an unconditional result by use of existing bounds on factorization and finding quadratic nonresidues. Pollard [**28**] gives a worst-case bound for factoring D of $O(D^{1/4+\varepsilon})$ elementary operations, for all $\varepsilon > 0$. Burgess [**5**] shows that all primes p have a quadratic nonresidue n with

$$0 < n < C(\varepsilon)p^{1/4e+\varepsilon} \tag{1.6}$$

for any $\varepsilon > 0$ and an effectively computable constant $C(\varepsilon)$ depending on ε. These yield the following result.

COROLLARY 1.3. *There is an algorithm which when given a positive D decides whether $X^2 - DY^2 = -1$ is solvable or not. This algorithm always terminates in $O(D^{1/4+\varepsilon})$ elementary operations, for any given $\varepsilon > 0$.*

The proof of Theorem 1.1 is based on the theory of integral quadratic forms. Basic notions and definitions of that theory are given in §2. That section establishes the well-known fact that (1.1) is solvable exactly when the indefinite binary quadratic form $X^2 - DY^2$ is equivalent to the form $-X^2 + DY^2$. Thus deciding the solvability of (1.1) can be viewed as a special case of deciding the equivalence or inequivalence of two binary quadratic forms. There is, however, no fast algorithm known for deciding the equivalence or inequivalence of two arbitrary indefinite quadratic forms. All known algorithms for deciding the equivalence of two binary quadratic forms of determinant $D > 0$ appear to take on the order of $D^{1/2}$ elementary operations in the worst case, even if a complete prime factorization of D is provided as input. The proof of Theorem 1.1 relies on a special property of the particular forms $X^2 - DY^2$ and $-X^2 + DY^2$. To explain this, recall that the set $\mathrm{Cl}(D)$ of equivalence classes of (properly primitive) quadratic forms of determinant D can be given the structure of an abelian group. The special property of the forms $X^2 - DY^2$ and $-X^2 + DY^2$ is that they are of order 1 or 2 in $\mathrm{Cl}(D)$. The proof of Theorem 1.1 actually gives a decision procedure for equivalence or inequivalence of forms known to be in the 2-Sylow subgroup $\mathrm{Cl}(D)_2$ of $\mathrm{Cl}(D)$. As an intermediate step, the algorithm produces a set of forms whose equivalence classes form a basis of $\mathrm{Cl}(D)_2$.

There has been extensive research on the problem of determining the structure of $\mathrm{Cl}(D)_2$. These include algorithms of Bauer [**4**], Hasse [**13**], Kaplan [**18**], Morton [**26**], Redei [**31**] and Shanks [**34**]. All of these algorithms appear to have worst-case running time bounds exponential in $\log D$, even when a complete factorization of D is provided. The algorithms of Bauer [**4**], Hasse [**13**], Kaplan [**18**] and Shanks [**34**] do not use the basis algorithm, and hence may be exponential in $\log D$ due to the

possible large size of $\mathrm{Cl}(D)_2$. The algorithms of Bauer [4], Hasse [13], Morton [26] and Redei [31] rely on finding nonzero solutions to certain diagonal ternary quadratic forms

$$aX^2 + bY^2 + cZ^2 = 0 \tag{1.7}$$

by the reduction procedure of Lagrange or direct search. Direct search is based on the bound of Holzer [15] that when (1.7) is solvable there exists a nonzero solution (X, Y, Z) to (1.7) with $|X| \leqslant \sqrt{|bc|}$, $Y \leqslant \sqrt{|ca|}$, $Z \leqslant \sqrt{|ab|}$. This yields a worst-case running time exponential in $\log(|a| + |b| + |c|)$. Analysis of the usual proof of convergence for Lagrange's procedure (Dickson [7, p. 129]) yields a worst-case running time bound exponential in $\log(|a| + |b| + |c|)$.[3] (There is however an algorithm for solving (1.5) due to Gauss [11, Article 292] which may be quite efficient, but has not been analyzed.) Finally the algorithm of Redei [31] requires constructing generators for certain class fields, and the possibility has not been ruled out that these generators require a number of binary bits exponential in $\log D$ to write down.

Our results on determining $\mathrm{Cl}(D)_2$ follow. Let Q denote a form of determinant D and $[Q]$ its equivalence class in $\mathrm{Cl}(D)$.

THEOREM 1.4. *There is an algorithm which when given any D not a perfect square and given*

(i) *a complete factorization of D,*

(ii) *a quadratic nonresidue n_i for each prime p_i dividing D*

determines a set of forms Q_j whose classes $[Q_j]$ form a basis of $\mathrm{Cl}(D)_2$ and determines the exact order of each $[Q_j]$ in that group. This algorithm terminates in $O((\log D)^5(\log\log D)(\log\log\log D))$ elementary operations in the worst case.

In particular, with the input (i), (ii) above, the complete set of 2-invariants of $\mathrm{Cl}(D)$ can be determined in $O((\log D)^5(\log\log D)(\log\log\log D))$ elementary operations (see Lagarias [20]).

THEOREM 1.5. *There is an algorithm which when given any D not a perfect square and given*

(i) *a complete factorization of D,*

(ii) *a quadratic nonresidue n_i for each prime p_i dividing D,*

(iii) *two quadratic forms Q_1, Q_2 of determinant D such that $[Q_1], [Q_2] \in \mathrm{Cl}(D)_2$, will decide the equivalence or inequivalence of Q_1 and Q_2. Let $L = \mathrm{Max}(\|Q_1\|, \|Q_2\|)$. This algorithm requires*

$$O\big((\log D)^5(\log\log D)(\log\log\log D) + (\log L)^2(\log\log L)(\log\log\log L)\big)$$

elementary operations in the worst case.

Here $\|Q\|$ is a measure of the size of the coefficients of the form Q defined in §2 by (2.6).

---

[3] Lagrange's procedure may find solutions much larger than Holzer's bound. It seems possible that those solutions may require a number of binary bits exponential in $\log(|a| + |b| + |c|)$.

The proof of these theorems splits naturally into two parts. The first part involves a purely group-theoretic *basis algorithm* for constructing a basis of an abelian p-group A given

(i) a generating set for the elements of order p in A,

(ii) a basis of the characters of order p on A,

(iii) an element $a \in A$ such that $X^p = a$ has at least one solution in A, a method for finding one such solution X.

Given a basis of A and (ii), (iii) above, there is a simple *representation algorithm* which can be used to decide whether two given elements of A are equal or not. Theorems 1.4 and 1.5 use the case $p = 2$. These algorithms are described in §3. They have been independently discovered by P. Morton [**26**], who observes they are implicit in the work of Redei [**30**], [**31**].

The second part involves worst-case complexity analyses of algorithms supplying the prerequisites (i)–(iii) of the basis algorithm above. These require analyses of many of the basic algorithms underlying the theory of integral quadratic forms. These include algorithms to reduce binary and ternary quadratic forms, to compose two binary forms, to evaluate the generic characters on a form, to decide whether a form is a square in $\mathrm{Cl}(D)$, and to extract a square root in $\mathrm{Cl}(D)$ of such a form if it is a square in $\mathrm{Cl}(D)$. Most of this analysis is carried out in Lagarias [**21**]. The required worst-case bounds are presented in §4. The proofs of Theorems 1.1, 1.4 and 1.5 follow in §5.

Finally we observe that Theorem 1.1 gives information on the complexity of recognizing the set

$$S = \{D \mid X^2 - DY^2 = -1 \text{ is solvable}\}.$$

An almost immediate corollary of this theorem is that the set S is in both the complexity classes NP and co-NP. General results on the complexity of recognizing certain subclasses of solvable Diophantine equations appear in Adleman and Manders [**1**], [**2**] and specific results concerning binary quadratic Diophantine equations appear in Lagarias [**22**] and Manders and Adelman [**24**].

2. Binary quadratic forms. A *binary quadratic form*

$$Q(X_1, X_2) = aX_1^2 + 2bX_1X_2 + cX_2^2 \tag{2.1}$$

is denoted $[a, 2b, c]$, and is said to be *integral* if its associated symmetric coefficient matrix

$$M_Q = \begin{bmatrix} a & b \\ b & c \end{bmatrix} \tag{2.2}$$

has integer entries. The *determinant* D of the form Q is given by

$$D = b^2 - ac = -\det(M_Q). \tag{2.3}$$

Such a form is *definite* if $D < 0$, *indefinite* if $D > 0$ is not a square and *degenerate* if D is a perfect square. An integral binary form $[a, 2b, c]$ is *properly primitive* if $(a, 2b, c) = 1$. Two forms Q_1 and Q_2 are *equivalent* if there is an integer unimodular matrix S such that

$$M_{Q_2} = S^t M_{Q_1} S. \tag{2.4}$$

In this case we write $Q_1 \approx Q_2$, and denote the equivalence class of Q by $[Q]$.

The relation between the solvability of the equation $X^2 - DY^2 = -1$ and the equivalence of two particular binary forms is well known (Smith [**36**, p. 197]).

PROPOSITION 2.1. *The following are equivalent.*

(i) $X^2 - DY^2 = -1$ *is solvable in integers.*

(ii) *The forms* $[1, 0, -D]$ *and* $[-1, 0, D]$ *are equivalent.*

PROOF. Suppose $X^2 - DY^2 = -1$. Then

$$\begin{pmatrix} X & Y \\ DY & -X \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & -D \end{pmatrix}\begin{pmatrix} X & DY \\ Y & -X \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & D \end{pmatrix},$$

hence $[1, 0, -D] \approx [-1, 0, D]$. Conversely, suppose

$$\begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & -D \end{pmatrix}\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & D \end{pmatrix}, \tag{2.5}$$

where $a_{11}a_{22} - a_{12}a_{21} = 1$. Comparing the upper left entries of both sides of (2.5) yields $a_{11}^2 - Da_{21}^2 = -1$. □

Gauss [**11**] observed that the set of equivalence classes $\mathrm{Cl}(D)$ of properly primitive integral binary quadratic forms[4] with a fixed nonsquare determinant D could be given the structure of an abelian group under an operation he called *composition*. He actually defined this operation on pairs of binary forms, and showed it was well defined on equivalence classes. We denote the composition of two forms Q_1, Q_2 by $Q_1 \circ Q_2$. The following result allows us to compose forms of a special type (see Cohn [**6**, Chapter 13], Lagarias [**21**]).

PROPOSITION 2.2. *Given binary forms* $Q_1 = [a, 2b, c]$ *and* $Q_2 = [a', 2b, c']$ *of determinant* D *with* $(aa', 2b) = 1$, *then* $Q_1 \circ Q_2 \approx Q_3$ *where* $Q_3 = [aa', 2b, c/a']$.

(Note that $a' | c$ so Q_3 is integral.) Proposition 2.2 gives the following result.

COROLLARY 2.3. *Let* $I = [1, 0, -D]$ *and* $-I = [-1, 0, D]$. *Then* $[I]$ *is the identity element in* $\mathrm{Cl}(D)$ *and* $[-I]$ *is of order* 1 *or* 2 *in* $\mathrm{Cl}(D)$.

PROOF. $I \circ I \approx I$ and $-I \circ -I \approx I$. □

In particular the form classes $[I]$ and $[-I]$ are in the 2-Sylow subgroup $\mathrm{Cl}(D)_2$ of $\mathrm{Cl}(D)$.

We shall need a measure of the size of a binary form in order to count elementary operations. We first define the *size* $\|A\|$ of a matrix $A = [a_{ij}]$ to be

$$\|A\| = \operatorname*{Max}_{i,j} |a_{ij}|. \tag{2.6}$$

The *size of a binary form* $Q = [a, 2b, c]$ is the size of its coefficient matrix (2.2).

In the subsequent algorithms we shall deal primarily with *reduced* binary quadratic forms. An indefinite form $Q = [a, 2b, c]$ is reduced provided

$$0 < b < \sqrt{D}\,, \qquad \sqrt{D} - b < |a| < \sqrt{D} + b. \tag{2.7}$$

A definite form is reduced provided

$$|2b| \leqslant a \leqslant c. \tag{2.8}$$

---

[4]We abbreviate this to *binary form* henceforth.

A degenerate form is reduced provided

$$a = 0, \qquad 0 \leqslant c \leqslant 2b - 1. \tag{2.9}$$

A reduced form $Q = [a, 2b, c]$ cannot have large coefficients in terms of its determinant D. The size of a reduced form satisfies the bounds

$$|D| \geqslant \|Q\| \geqslant \tfrac{1}{2}\sqrt{|D|} \tag{2.10}$$

regardless of whether it is definite, indefinite or degenerate.

3. Basis and representation algorithms for abelian p-groups. Let A be a finite abelian p-group with identity element 1, and the group operation denoted multiplicatively. A set $\{b_1, \ldots, b_g\}$ is an *ordered basis*[5] of A if

(i) ord $b_i = p^{n_i} \leqslant$ ord $b_{i+1} = p^{n_{i+1}}$ for all i,

(ii) each element $a \in A$ can be uniquely expressed in the form

$$a = \prod_{i=1}^{g} (b_i)^{f_i}, \qquad 0 \leqslant f_i < p^{n_i}, \text{ all } i. \tag{3.1}$$

Let h denote the exponent of A. We define the *p-invariants* e_i for $1 \leqslant i \leqslant h$ in terms of an ordered basis $\{b_i\}$ by the conditions that e_i be the number of basis elements of order equal or exceeding p^i. Note that $e_1 = g$, the number of generators of A. By the fundamental theorem of finitely generated abelian groups, such a group has an ordered basis, and the numbers $e_1, \ldots, e_h$, h are invariants of the group A independent of the ordered basis chosen.

The following notation is used in the remainder of this section.

$$A^p = \{a^p | a \in A\}, \quad A_j = \{a | a^{p^j} = 1\},$$

$$\hat{A} = \text{the character group of } A, \quad \hat{A}_1 = \{\chi | \chi^p = \hat{1} \in \hat{A}\}.$$

Characters χ of $\hat{A}_1$ will be considered as mapping into the additive group of the finite field $\mathbf{Z}/p\mathbf{Z}$, by use of the isomorphism

$$\exp(2\pi i k/p) \mapsto k \pmod{p}. \tag{3.2}$$

We shall present an algorithm for finding an ordered basis of a finite abelian p-group A when we are given the following.

(3.3)

(i) A generating set $\{t_i | 1 \leqslant i \leqslant m\}$ of the subgroup A_1 of elements of order $\leqslant p$ in A.

(ii) A basis $\{\chi_j | 1 \leqslant j \leqslant g\}$ for the group $\hat{A}_1$ of characters of order p of A,[6] and a means of evaluating $\chi_j(a)$ for any j, any $a \in A$.

(iii) An algorithm which when given $a \in A^p$ finds an element $b \in A$ such that $b^p = a$. That is, it extracts a pth root of a in A.

This algorithm is based on a criterion enabling us to recognize a basis.

We first recall a special case of the Burnside basis theorem [**12**, p. 176].

---

[5]If condition (ii) alone holds, we call such a set $\{b_i\}$ a *basis*.

[6]Later we will verify that a basis of A_1 has the same number of generators as a basis of A.

PROPOSITION 3.1. *Let A be an abelian p-group. The canonical projection homomorphism $\pi\colon A \to A/A^p$ takes any basis $\{b_1, \ldots, b_g\}$ of A to a basis $\{\pi(b_1), \ldots, \pi(b_g)\}$ of A/A^p. Conversely, if $\{\pi(b_1), \ldots, \pi(b_g)\}$ is a basis of A/A^p then $\{b_1, \ldots, b_g\}$ generates A, but is not necessarily a basis.* □

This shows A/A^p requires g generators. Since $A/A^p \cong \widehat{(A/A_p)} \cong \hat{A}_1$ by the duality of an abelian group and its character group [**12**, pp. 195–196], $\hat{A}_1$ requires exactly g generators.

To state the basis recognition criterion, we first note that since A/A^p is an elementary p-group (i.e. a direct sum of copies of $\mathbf{Z}/p\mathbf{Z}$), it may be regarded as a vector space over the finite field $\mathbf{Z}/p\mathbf{Z}$, which has dimension g by Proposition 3.1. The images $\pi(A_i)$ of the groups A_i under the canonical map π are subspaces of A/A^p. We need an auxiliary lemma specifying $\dim \pi(A_i)$, for which we establish the convention that $e_{h+1} = 0$.

LEMMA 3.2. $\dim \pi(A_i) = g - e_{i+1}$ *for* $0 \leqslant i \leqslant h$.

PROOF. Let $\{b_1, \ldots, b_g\}$ be a basis of A. Then $\{b_1^{(i)}, \ldots, b_g^{(i)}\}$ is a basis of A_i where

$$b_j^{(i)} = \begin{cases} b_j & \text{if ord } b_j \leqslant p^i, \\ (b_j)^{p^k} & \text{if ord } b_j = p^{i+k},\ k \geqslant 1. \end{cases}$$

By the definition of e_i, $b_j^{(i)} = b_j$ exactly when $j \leqslant g - e_{i+1}$. Then by Proposition 3.1, $\{\pi(b_1^{(i)}), \ldots, \pi(b_{g-e_{i+1}}^{(i)})\}$ are linearly independent. Furthermore $\pi(b_j^{(i)}) = 1$ for $g - e_{i+1} < j \leqslant g$ since $b_j^{(i)} \in A^p$ in that case. Since the images $\pi(b_j^{(i)})$ generate $\pi(A_i)$, $\dim \pi(A_i) = g - e_{i+1}$. □

THEOREM 3.3 (BASIS RECOGNITION CRITERION). *Let A be an abelian p-group and $\pi\colon A \to A/A^p$ the canonical projection homomorphism. Then $\{b_1, \ldots, b_g\}$ is an ordered basis of A if and only if*

(i) $\{\pi(b_1), \ldots, \pi(b_g)\}$ *is a basis of* A/A^p,

(ii) $\{b_1, \ldots, b_{g-e_i}\} \subseteq A_{i-1}$ *for* $1 \leqslant i \leqslant h$.

PROOF. $\Rightarrow$ (i) holds by Proposition 3.1 and (ii) by Lemma 3.2 and the definition of ordered basis.

$\Leftarrow$ By Proposition 3.1, (i) implies $\{b_1, \ldots, b_g\}$ is a generating set. So every element $a \in A$ has a representation

$$a = \prod_{i=1}^{g} b_i^{n_i}, \qquad 0 \leqslant n_i < \operatorname{ord}(b_i). \tag{3.4}$$

Now

$$|A| = p^{e_1 + e_2 + \cdots + e_h}. \tag{3.5}$$

On the other hand (ii) implies

$$\operatorname{ord} b_j \leqslant p^i \text{ when } g - e_i < j \leqslant g - e_{i+1} \tag{3.6}$$

for $1 \leqslant i \leqslant h$. This bound shows that if R is the number of distinct elements arising from (3.4) then

$$R \leqslant p^{e_1+e_2+\cdots+e_h} \tag{3.7}$$

and that equality can occur in (3.7) only if

$$\text{ord } b_j = p^i \quad \text{when } g - e_i < j \leqslant g - e_{i+1} \tag{3.8}$$

holds for all i. But (3.5) forces equality in (3.7), and this also shows that representation of an element $a \in G$ in (3.4) is unique, verifying (3.1) (ii). Finally (3.1) (i) follows from (3.8). □

We now turn to the basis algorithm. A key fact underlying the algorithm is that the $\mathbf{Z}/p\mathbf{Z}$-vector space A/A^p can be coordinatized using the basis of characters $\{\chi_1, \ldots, \chi_g\}$ of $\hat{A}_1$ provided by (3.3) (ii). We identify $\pi(a) \in A/A^p$ with the coordinates $(\chi_1(a), \ldots, \chi_g(a))$ of a g-dimensional $\mathbf{Z}/p\mathbf{Z}$-vector space V. This is well defined because if a, a' are two preimages of $\pi(a)$ then $a' = ab^p$ and $\chi(a') = \chi(a)\chi(b^p) = \chi(a)$ for all $\chi \in \hat{A}_1$. The vector space V is a coordinatized version of the second dual of A/A^p, and this identification is just the canonical isomorphism between an abelian group and its second dual. The projection map π: $A \to A/A^p \cong V$ in coordinate form is

$$\pi(a) = (\chi_1(a), \ldots, \chi_g(a)). \tag{3.9}$$

The *basis algorithm* runs as follows.

Cycle 1. Given the elements t_i, $1 \leqslant i \leqslant m$, form the $m \times g$ matrix

$$M^{(1)} = \left[\chi_j(t_i)\right] \tag{3.10}$$

with $\mathbf{Z}/p\mathbf{Z}$ entries whose rows are the coordinates (3.9) of $\pi(t_i)$. Use Gaussian elimination over $\mathbf{Z}/p\mathbf{Z}$ via elementary row operations to find an $m \times m$ matrix

$$R^{(1)} = \left[r_{il}^{(1)}\right], \qquad 1 \leqslant i, l \leqslant m, \tag{3.11}$$

over $\mathbf{Z}/p\mathbf{Z}$ of determinant 1 such that

$$T^{(1)} = R^{(1)}M^{(1)} \tag{3.12}$$

is in reduced row-echelon form. Set $r_1 = \text{rank}(T^{(1)})$, so that exactly the first r_1 rows of $T^{(1)}$ are nonzero. Using (3.11) and (3.12) we can rewrite $T^{(1)}$ as $T^{(1)} = [\chi_j(u_i^{(1)})]$ where

$$u_i^{(1)} = \prod_{l=1}^{m} (t_l)^{r_{il}^{(1)}}. \tag{3.13}$$

Note $0 \leqslant r_{il}^{(1)} < p$. Set

$$b_i = u_i, \qquad 1 \leqslant i \leqslant r_1. \tag{3.14}$$

These will be the first r_1 elements of the basis, and will remain fixed throughout the rest of the algorithm. If $r_1 = g$, halt. Otherwise go to Cycle 2.

Cycle $k + 1$ ($k \geqslant 1$). We suppose that we are given r_k with $r_k < g$ and a set $\{u_i^{(k)} | 1 \leqslant i \leqslant m\}$ such that the matrix

$$T^{(k)} = \left[\chi_j(u_i^{(k)})\right] \tag{3.15}$$

has its first r_k rows linearly independent over $GF(p)$ and the remaining rows zero. Each row i, for $1 \leqslant i \leqslant r_k$, has a column c_i of lowest index containing a nonzero entry, called its *leading column*. We also require that

$$\begin{aligned} &\text{(i) all leading columns } c_i \text{ are distinct,} \\ &\text{(ii) for } 2 \leqslant i \leqslant r_k, \text{ row } i \text{ is zeroed out in all} \\ &\quad \text{leading columns } c_j \text{ for } 1 \leqslant j < i. \end{aligned} \tag{3.16}$$

(These conditions are equivalent to the existence of a permutation of the columns that puts the matrix $T^{(k)}$ in reduced row-echelon form.)

Since the χ_j are a basis of $\hat{A}_1$, if $\chi_j(u) = 0$ for $1 \leqslant j \leqslant g$ then $u \in A^p$. By hypothesis this is the case for $u_i^{(k)}$, $r_k + 1 \leqslant i \leqslant m$. Use the pth root extraction algorithm to find $t_i^{(k+1)}$ such that

$$\left(t_i^{(k+1)}\right)^p = u_i^{(k)}, \qquad r_k + 1 \leqslant i \leqslant m. \tag{3.17}$$

Also set

$$t_i^{(k+1)} = u_i^{(k)}, \qquad 1 \leqslant i \leqslant r_k, \tag{3.18}$$

and consider the $m \times g$ matrix $M^{(k+1)} = [\chi_j(t_i^{(k+1)})]$. Now apply Gauss elimination to rows $r_k + 1$ through m, holding rows 1 through r_k fixed, to obtain $T^{(k+1)}$. This involves subtracting multiples of rows $1, 2, \ldots, l-1$ in this order to zero out the corresponding leading column entry of the row l being reduced. If row l is zeroed out, exchange it with a nonzero row of higher subscript and reduce the new row l. This procedure yields an $m \times m$ matrix

$$R^{(k+1)} = \left[r_{il}^{(k+1)}\right], \qquad 1 \leqslant i, l \leqslant m, \tag{3.19}$$

of determinant 1 such that $T^{(k+1)} = R^{(k+1)}M^{(k+1)}$. Here $T^{(k+1)}$ has rank $r_{k+1} \geqslant r_k$, its first r_k rows coincide with those of $T^{(k)}$, its first r_{k+1} rows satisfy (3.16) and all its remaining rows are zero. We calculate $T^{(k+1)} = [\chi_j(u_i^{(k+1)})]$ where

$$u_i^{(k+1)} = \prod_{l=1}^{m} \left(t_l^{(k)}\right)^{r_{il}^{(k+1)}}, \qquad 1 \leqslant i \leqslant m. \tag{3.20}$$

Set

$$b_i = u_i^{(k+1)} \quad \text{for } r_k < i \leqslant r_{k+1}. \tag{3.21}$$

If $r_{k+1} = g$, halt. If $r_{k+1} < g$, go to Cycle $k + 2$.

THEOREM 3.4. (i) *The algorithm halts at the end of cycle h, where h is the exponent of A.*

(ii) *The rank r_k of the matrix $T^{(k)}$ satisfies*

$$r_k = \dim \pi(A_k) = g - e_{k+1} \tag{3.22}$$

for $1 \leqslant k \leqslant h$.

(iii) *The set $\{b_1, \ldots, b_g\}$ obtained is a basis of A.*

PROOF. (i) and (ii). We have to establish two key facts.

FACT (A). At the kth cycle $\{u_1^{(k)}, \ldots, u_m^{(k)}\}$ is a generating set for A_k.

FACT (B). At the kth cycle $\{\pi(b_1), \ldots, \pi(b_{r_k})\}$ is a basis for $\pi(A_k)$.

The rank r_k of the matrix $T^{(k)}$ is the dimension of the subspace of $A/A^p \cong V$ spanned by $\{\pi(u_1^{(k)}), \ldots, \pi(u_m^{(k)})\}$ so the truth of fact (A) implies (ii). Also the truth of fact (A) for k implies fact (B) for k because the b_i were chosen so that $\{\pi(b_1), \ldots, \pi(b_k)\}$ is a basis of the row space of $T^{(k)}$. Finally fact (B) for $k = h$ and the fact that $\pi(A_h)$ has dimension g shows that the halting criterion is satisfied at exactly the end of cycle h, which is (i).

It suffices to prove fact (A), which we do by induction on k. It is true for $k = 1$ because the given t_i are a generating set and $R^{(1)}$ is invertible in (3.12). Assume fact (A), and hence fact (B), is true for $k = s$. Then a generating set for $A_s \cap A^p$ is $\{(b_1)^p, \ldots, (b_{r_s})^p, u_{r_s+1}^{(s)}, \ldots, u_m^{(s)}\}$. Suppose $a \in A_{s+1}$. Then $a^p \in A_s \cap A^p$. Hence

$$a^p = \left(\prod_{i=1}^{r_s} (b_i)^{pn_i} \right) \left(\prod_{i=r_s+1}^{m} (u_i^{(s)})^{n_i} \right) \tag{3.23}$$

for integer n_i, $1 \leqslant i \leqslant m$. By construction (3.17)

$$(t_i^{(s+1)})^p = u_i^{(s)}, \qquad r_s + 1 \leqslant i \leqslant m. \tag{3.24}$$

The $t_i^{(s+1)}$ are contained in the group B_{s+1} generated by $\{u_i^{(s+1)} | 1 \leqslant i \leqslant m\}$ because the matrix $R^{(k+1)}$ in (3.19) is invertible over $\mathbf{Z}/p\mathbf{Z}$. This also shows $A_s \subseteq B_{s+1}$ via (3.17), (3.18). Now consider the element

$$a_1 = \left(\prod_{i=1}^{r_s} b_i^{n_i} \right) \left(\prod_{i=r_s+1}^{m} (t_i^{(s+1)})^{n_i} \right). \tag{3.25}$$

Then $a_1 \in B_{s+1}$ by the preceding remarks. Since $a^p = (a_1)^p$ from (3.24), (3.25) we conclude $a = a_1 a_2$ for some a_2 with $a_2^p = 1$. Then $a_2 \in A_1 \subseteq A_s \subseteq B_{s+1}$, hence also $a \in B_{s+1}$. This establishes $A_{s+1} \subseteq B_{s+1}$. But the set $\{u_i^{(s+1)} | 1 \leqslant i \leqslant m\}$ is generated by the $t_i^{(s+1)}$ in (3.17), (3.18) and the induction hypothesis shows $t_i^{(s+1)} \in A_{s+1}$ for $1 \leqslant i \leqslant m$. Hence $B_{s+1} \subseteq A_{s+1}$, completing the induction step and proving fact (A).

(iii) We check that $\{b_i | 1 \leqslant i \leqslant g\}$ satisfies the basis recognition criterion. The truth of condition (i) is just fact (B) for $k = h$, and that of condition (ii) is implied by fact (A) and the already proved part (ii) of this theorem. By Theorem 3.3 we are done. □

REMARKS. (1) The complete set of p-invariants of A is determined by the basis algorithm via (3.22).

(2) The basis algorithm also produces the quantities $u_i^{(h)}$ for $g + 1 \leqslant i \leqslant m$. Since A is of exponent h,

$$(u_i^{(h)})^{p^h} = 1, \qquad g + 1 \leqslant i \leqslant m. \tag{3.26}$$

By use of the matrices $R^{(k)}$ and the t_i given in (3.17), (3.18), the equations (3.26) yield nontrivial relations among the originally given $\{t_i | 1 \leqslant i \leqslant m\}$. In fact they give a complete set of $m - g$ independent relations among the t_i. We do not pursue this further here.

Once an ordered basis $\{b_1, \ldots, b_g\}$ of A has been found, there is a simple algorithm to represent a given element $a \in A$ in terms of this basis, if the following are available.

(i) The order p^{n_i} of each basis element b_i,
(ii) a basis $\{\chi_1, \ldots, \chi_g\}$ of the characters of order p on A, and a means of evaluating $\chi_i(a)$ for any χ_i and any $a \in A$,
(iii) an algorithm which when given $a \in A^p$ finds an element $b \in A$ such that $b^p = a$. (3.27)

This *representation algorithm* proceeds as follows.

Initialization. Set $a_1 = a$. Go to Cycle 1.

Cycle k. Given a_k. Since $\{\pi(b_i) | 1 \leqslant i \leqslant g\}$ are a basis of V, we can find the unique relation

$$\pi(a_k) = \sum_{i=1}^{g} n_{ik}\pi(b_i), \qquad 0 < n_{ik} < p, \tag{3.28}$$

by Gaussian elimination. Set

$$c_k = \prod_{i=1}^{g} b_i^{n_{ik}} \tag{3.29}$$

and define a^* by $a_k^* = (c_k)^{-1}a_k$. Then $\pi(a_k^*) = 0$ so $a_k^* \in A^p$. Use the pth root extraction algorithm to find $a_{k+1} \in A$ such that $(a_{k+1})^p = a^*$. If $k < h$ go to Cycle $k + 1$. If $k = h$ go to termination step.

Termination. We have now found

$$a = c_1 c_2^p \cdot \cdots \cdot (c_h)^{p^{h-1}}(a_{h+1})^{p^h} = c_1 c_2^p \cdot \cdots \cdot (c_h)^{p^{h-1}}$$

since A has exponent h. Use the expressions (3.29) to give $a = \prod_{i=1}^{g}(b_i)^{m_i}$ and finally reduce each $m_i \pmod{p^{n_i}}$ where $\operatorname{ord}(b_i) = p^{n_i}$.

The representation algorithm allows us to test equality of two elements $a_1, a_2 \in A$. We compute their representations (3.1) in terms of the basis and equality occurs if and only if all their exponents f_i agree.

4. Ambiguous classes, genus characters and square root extraction. In order to use the basis and representation algorithms on the 2-Sylow subgroup $\mathrm{Cl}(D)_2$ of the form class group $\mathrm{Cl}(D)$, we need the following.

(i) Algorithms to compute the group operations of $\mathrm{Cl}(D)$ on individual forms, e.g., algorithms to compose two forms and to find the inverse of a form.
(ii) A basis $\{\chi_i | 1 \leqslant i \leqslant G\}$ of all characters of order 2 on $\mathrm{Cl}(D)$, and a means of evaluating $\chi_i(Q)$ for a given character χ_i and form Q.
(iii) A set of binary forms $\{Q_i | 1 \leqslant i \leqslant M\}$ whose classes $[Q_i]$ are in $\mathrm{Cl}(D)_2$ and generate all classes of order 2 in $\mathrm{Cl}(D)_2$.
(iv) An algorithm which when given a form Q whose class $[Q]$ is a square in $\mathrm{Cl}(D)$, finds a form G such that $G \circ G = Q$.

Worst-case complexity bounds on methods to do these are given in subsections B, C, D and E respectively. Subsection A treats the complexity of reducing a form, which plays a subsidiary role in the subsequent algorithms.

In order to simplify the statements of this section, we use $M(n)$ such that

$$M(n) = n(\log n)(\log\log n). \tag{4.1}$$

This is the Schönhage-Strassen bound on multiplication of two n-bit integers (see Knuth [**19**, p. 274]).

A. *Reduction of binary quadratic forms.* In order to keep the coefficients of the binary quadratic forms small during the course of the algorithms considered, we apply algorithms reducing such forms at intermediate stages. Note that reduced forms satisfy the size bounds (2.10). Reduction algorithms were analyzed in Lagarias [**21**, Theorems 4.1, 4.2, 4.5], and these yield the following result.

PROPOSITION 4.1. *We are given a binary form $Q = [a, 2b, c]$ which may be definite, indefinite or degenerate. In each case there exists an algorithm which finds a unimodular matrix S and an equivalent reduced form Q_{red} resulting from transforming Q by S. This algorithm takes $O(\log\|Q\| M(\log\|Q\|))$ elementary operations in the worst case and the size of S is bounded by*

$$\log\|S\| = O(\log\|Q\|). \tag{4.2}$$

B. *Composition of forms.* The group law on $\mathrm{Cl}(D)$ is given by composition. This operation is defined on individual forms. The following worst-case bound for composing two reduced forms appears in Lagarias [**21**, Theorem 5.5].

PROPOSITION 4.2. *Given any two properly primitive reduced forms Q_1, Q_2 of nonsquare determinant D there is an algorithm that deterministically computes a properly primitive reduced form Q_3 of determinant D such that $Q_1 \circ Q_2 = Q_3$, which requires no more than $O(\log D\, M(\log D))$ elementary operations in the worst case.*

The *inverse* of a form $Q = [a, 2b, c]$ is $\overline{Q} = [a, -2b, c]$. Now $\overline{Q} \approx \overline{\overline{Q}} = [c, 2b, a]$. It can be shown using Proposition 2.2 that $[\overline{Q}] = [Q]^{-1}$. In the case Q is indefinite, $\overline{Q}$ is reduced if and only if Q is reduced. Consequently it takes $O(\log D)$ elementary operations to compute the inverse of a reduced form Q.

C. *Genus characters.* The characters χ of order 2 on the class group $\mathrm{Cl}(D)$ of a nonsquare determinant D are called *genus characters*. They are essentially quadratic residue symbols (Legendre symbols) evaluated at integers N represented by any form Q in the class $[Q]$, with $(N, 4D) = 1$. We use the notation $\chi_p(N) = (N/p)$ to denote the Legendre symbol with respect to the odd prime p. We also need three *supplementary characters* (mod 8). These are

$$\chi_{-4}(N) = \begin{cases} (-1)^{(N-1)/2}, & N \equiv 1 \pmod 2, \\ 0, & N \equiv 0 \pmod 2, \end{cases}$$

$$\chi_8(N) = \begin{cases} (-1)^{(N^2-1)/8}, & N \equiv 1 \pmod 2, \\ 0, & N \equiv 0 \pmod 2, \end{cases}$$

$$\chi_{-8}(N) = \chi_{-4}(N)\chi_8(N).$$

The Legendre symbols described above are sufficient to give a basis for all genus characters. We need a complete factorization of D to describe this basis. Set $D = df^2$ where d is the squarefree part of D. Let $p_1, \ldots, p_r$ denote the odd primes dividing D, and let $q_1, \ldots, q_s$ be the odd primes dividing f which do not divide D. Let M denote the number of distinct prime divisors of D, so that $M = r + s + 1$

or $r + s$ according as 2 divides D or not. Let G denote the number of generators required for all genus characters. The following result gives a basis for the genus characters (Mathews [**25**, pp. 132–137]).

PROPOSITION 4.3. *The symbols in Table* I *all give rise to genus characters for the specified D. The corresponding genus characters are independent except for the single relation that (for each row) the product of the field characters is the trivial character. If the first character in each row is deleted, the remaining characters in that row form a basis for the genus characters for that type of D. They are exactly* 2^G *distinct genus characters where G is* $M - 1$ *or M as specified in Table* I.

By Gauss' principal genus theorem the genus characters include all characters of order 2 on $\mathrm{Cl}(D)$. So by the remark following Proposition 3.1, the number of generators of $\mathrm{Cl}(D)_2$ is G.

TABLE I. Basis for Genus Characters

Determinant $D = df^2$		Field Characters	Ring Characters	Number of Generators G
$d \equiv 1 \pmod 4$	$f \equiv 1 \pmod 4$	$\chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}$	$M - 1$
	$f \equiv 2 \pmod 4$	$\chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}, \chi_{-4}$	$M - 1$
	$f \equiv 0 \pmod 4$	$\chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}, \chi_{-4}, \chi_8$	M
$d \equiv 3 \pmod 4$	$f \equiv 1 \pmod 2$	$\chi_{-4}, \chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}$	M
	$f \equiv 2 \pmod 4$	$\chi_{-4}, \chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}$	$M - 1$
	$f \equiv 0 \pmod 4$	$\chi_{-4}, \chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}, \chi_8$	M
$d \equiv 2 \pmod 8$	$f \equiv 1 \pmod 2$	$\chi_8, \chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}$	$M - 1$
	$f \equiv 0 \pmod 2$	$\chi_8, \chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}, \chi_{-4}$	M
$d \equiv 6 \pmod 8$	$f \equiv 1 \pmod 2$	$\chi_{-8}, \chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}$	$M - 1$
	$f \equiv 0 \pmod 2$	$\chi_{-4}, \chi_8, \chi_{p_1}, \ldots, \chi_{p_r}$	$\chi_{q_1}, \ldots, \chi_{q_s}$	M

The following result bounding the complexity of computing genus characters is taken from Lagarias [**21**, Theorem 6.3].

PROPOSITION 4.4. *Suppose a complete prime factorization of D is given. Then for any properly primitive reduced form Q of determinant D,*

(i) *the values* $\chi_i(Q)$ *for all the relevant basis characters can be determined in* $O(\log D\, M(\log D))$ *elementary operations,*

(ii) *for any genus character* χ *given expressed in terms of the basis characters,* $\chi(Q)$ *can be evaluated in* $O(\log D\, M(\log D))$ *elementary operations.*

D. *Ambiguous forms.* An *ambiguous form* $[a, 2b, c]$ is a form for which $a|2b$. In the case $a|4D$ Gauss [**11**, Article 187] showed the following.

PROPOSITION 4.5. *Ambiguous forms occur only in classes of order* 2 *in the class group. When* $D > 0$ *there are exactly two reduced ambiguous forms in each class of order* 2.

This justifies calling classes of order 2 *ambiguous classes*. Since the number of ambiguous classes equals the number of genus characters, when $D > 0$ there are exactly 2^{G+1} reduced ambiguous forms.

LEMMA 4.6. *Every reduced ambiguous form is equivalent to a properly primitive form of shape* $[a, 0, c]$ *or* $[a, a, c]$.

PROOF. The transformation $\left[\begin{smallmatrix} 1 & 0 \\ \lambda & 1 \end{smallmatrix}\right]$ takes $[a, 2b, c]$ to $[a, 2b + 2\lambda a, c']$. Choose λ such that $0 \leqslant 2b + 2\lambda a < 2a$. Since $a|2b$, we have $2b + 2\lambda a = 0$ or a. □

The converse of Lemma 4.6 is true but we do not need it here.

Table II below exhibits a set of ambiguous forms which we will show generate all classes of order 2 under composition. The forms in Table II are usually not reduced forms, however.

TABLE II. Generating Set of Ambiguous Forms

$$D = p_1^{2a_1+1} \cdots p_r^{2a_r+1} q_1^{2b_1+2} \cdots q_s^{2b_s+2} 2^t \text{ and}$$

the p_i, q_j are distinct odd primes.

$$Q_i = \left[p_i^{2a_i+1}, 0, -\frac{D}{p_i^{2a_i+1}} \right], \qquad 1 \leqslant i \leqslant r,$$

$$Q_{r+j} = \left[q_j^{2b_j+2}, 0, -\frac{D}{q_j^{2b_j+2}} \right], \qquad 1 \leqslant j \leqslant s,$$

$$Q_{r+s+1} = \begin{cases} \left[2, 2, \dfrac{1-D}{2} \right] & \text{if } D \equiv 3 \pmod 4, \\ \left[2^t, 0, -\dfrac{D}{2^t} \right] & \text{if } D \equiv 0 \pmod 2, \end{cases}$$

$$Q_{r+s+2} = \left[4, 4, 1 - \frac{D}{4} \right] \quad \text{if } D \equiv 0 \pmod 8.$$

LEMMA 4.7. *For any nonsquare* $D > 0$ *Table* II *gives a set of* $G + 1$ *ambiguous forms whose classes under composition generate all classes of order* 2 *in* $\mathrm{Cl}(D)$.

PROOF. The forms Q_{r+s+1}, Q_{r+s+2} are present only for D in the specified congruence classes. A comparison of Tables I and II shows there are $G + 1$ forms in Table II in every case.

It suffices to show that all properly primitive forms of shape $[a, 0, c]$ or $[a, a, c]$ can be obtained by composition from the Q_k, since Proposition 4.5 and Lemma 4.6 guarantee that every class of order 2 contains such a form.

Consider $Q = [a, 0, c]$ first. We may assume $a > 0$ without loss of generality since $ac = -D < 0$ and $[a, 0, c] \approx [c, 0, a]$. Since $a|D$, the prime-power factors of a divide $p_i^{2a_i+1}$, $q_j^{2b_j+2}$ or 2^t. If $p_i|a$ then $(a, c) = 1$ shows $p_i \nmid c$ while $ac = -D$ shows $p_i^{2a_i+1} \| a$. Similarly if $q_j|a$ then $q_j^{2b_j+2} \| a$, if $2|a$ then $2^t \| a$. Applying Proposition 2.2 repeatedly then shows Q is composed of exactly the Q_i whose first coefficient divides a.

Next consider $Q = [a, a, c]$. These can only occur if $D \equiv 0, 3, 7 \pmod 8$. Certainly $2|a$, and if $D \equiv 3 \pmod 4$ then $D = \frac{1}{2}a(\frac{1}{2}a - 2c)$ shows $a \equiv 2 \pmod 4$. If $D \equiv 0 \pmod 8$ then either $4\|a$ or $2^t\|a$, according to whether $4|\frac{1}{2}a$ or $4|(\frac{1}{2}a - 2c)$. Since $[a, a, c] \approx [4c - a, 4c - a, c]$ via $\left[\begin{smallmatrix} -1 & -1 \\ 2 & 1 \end{smallmatrix}\right]$, and $(a, c) = 1$ guarantees c is odd, we may assume without loss of generality that $4\|a$ in this case. Next we claim that

$$[a, 0, c] \circ [a', a', c'] = [aa', aa', (a' - 2c')/2a] \tag{4.3}$$

whenever $(a, a') = 1$. To see this, note $2|a'$, $2\nmid a$ so that

$$[a, 0, c] \approx [a, aa', c''] \quad \text{via} \begin{bmatrix} 1 & a'/2 \\ 0 & 1 \end{bmatrix}$$

and

$$[a', a', c'] \approx [a', aa', c'''] \quad \text{via} \begin{bmatrix} 1 & (a-1)/2 \\ 0 & 1 \end{bmatrix}.$$

These last two forms can be composed by Proposition 2.2 to prove the claim. If $D \equiv 3 \pmod 8$, then $Q = [2a', 2a', c']$ where a' is odd. By (4.3),

$$Q = [2, 2, (1 - D)/2] \circ [a', 0, -D/a'].$$

Here the first form is Q_{r+s+1} and the second has previously been produced from the Q_k by composition. For $D \equiv 0 \pmod 8$ we assumed $Q = [4a', 4a', c']$ with a' odd, so by (4.3),

$$Q = [4, 4, 1 - D/4] \circ [a', 0, -D/a']$$

and as before Q is composed from the Q_k's. □

E. *Square root extraction algorithm.* Gauss [**11**, Article 286] gave an algorithm for finding a square root under composition of a form that is a square in the form class group $\mathrm{Cl}(D)$. The following result gives a worst-case complexity bound for square root extension (Lagarias [**21**, Corollary 6.9]).

PROPOSITION 4.8. *Given a properly primitive reduced form $Q = [a, 2b, c]$ of nonsquare determinant D, suppose that*

(i) *a complete factorization of D is provided,*

(ii) *a quadratic nonresidue n_i is given for each prime p_i dividing D,*

(iii) $[Q]$ *is the square of some element of* $\mathrm{Cl}(D)$.

There is an algorithm which produces a properly primitive reduced form G such that $[G] \circ [G] = [Q]$. This algorithm terminates in $O((\log|D|)^3 M(\log|D|))$ elementary operations in the worst case.

5. Complexity bounds. The first step is to bound the worst-case complexity of the basis and representation algorithms.

THEOREM 5.1. *Suppose that A is a p-group requiring G generators, of exponent H, and with p-invariants $\{e_i | 1 \leqslant i \leqslant H\}$. Suppose a set of M elements $\{a_1, \ldots, a_M\}$ which generate the elements of order p in A is given. The basis algorithm finds a basis $\{b_1, \ldots, b_G\}$ of A. The following are upper bounds on the number of operations used in the basis algorithm in the worst case.*

(i) $O(M^2GH)$ *additions, multiplications and inverses of elements in* $\mathbf{Z}/p\mathbf{Z}$.
(ii) $O(M^2H \log p)$ *multiplications of group elements.*
(iii) $O((M - G)GH + (\sum_{i=1}^{H} e_i)G)$ *character evaluations.*
(iv) $O((M - G)H + \sum_{i=1}^{H} e_i)$ *extractions of a pth root in* A.

PROOF. Gauss elimination on the rows of an $M \times G$ matrix over $\mathbf{Z}/p\mathbf{Z}$ requires $O(M^2G)$ additions, multiplications and inverses in $\mathbf{Z}/p\mathbf{Z}$. (Finding a^{-1} in $\mathbf{Z}/p\mathbf{Z}$ is equivalent to solving $aX \equiv 1 \pmod{p}$. This can be done via the Euclidean algorithm in $O(\log p\ M(\log p))$ elementary operations, see Lagarias [**21**, Proposition 3.3].) Since at most H cycles occur, this gives (i). Multiplication of group elements occurs in (3.20). The bound $O(M^2H \log p)$ follows from the fact that M elements (3.20) are calculated during each cycle, each requires $O(M \log p)$ separate multiplications, and there are H cycles. The bound $M \log p$ comes from noting that any t^r with $0 \leqslant r < p$ can be calculated in $O(\log p)$ multiplications by expanding r in binary and using repeated squarings to calculate $t, t^2, t^4, \ldots$. For (iii), we need only note that the character evaluations required are those for all G characters for each new element $t_i^{(k+1)}$ for $r_k + 1 \leqslant i \leqslant M$ created via (3.17) at the kth cycle. By Theorem 3.4 (ii), $r_k = G - e_k$, there are $M - G + e_k$ of these new elements. For (iv) the only pth root extractions are those at the same point (3.17), and there are $M - G + e_k$ at the kth cycle. □

The bounds (i), (ii) in Theorem 5.1 can be improved, but this would not improve the results in the main theorems.

THEOREM 5.2. *Suppose* A *is a* p*-group requiring* G *generators, of exponent* H*, and with* p*-invariants* $\{e_i | 1 \leqslant i \leqslant H\}$. *The representation algorithm expresses a given element* $b \in G$ *in terms of an ordered basis* $\{b_1, \ldots, b_G\}$ *as*

$$b = \prod_{i=1}^{G} (b_i)^{n_i}, \qquad 0 \leqslant n_i < \operatorname{ord}(b_i).$$

The following are upper bounds on the number of operations used in the worst case.
(i) $O(GH)$ *additions, multiplications and inverses of elements in* $\mathbf{Z}/p\mathbf{Z}$.
(ii) $O(GH \log p)$ *multiplications of group elements.*
(iii) $O(GH)$ *character evaluations.*
(iv) $O(H)$ *pth root extractions in the group* A.

PROOF. Similar to that of Theorem 5.1. □

The remaining information we need is an upper bound for the number of generators G, the exponent H and the sum of the 2-invariants for $\mathrm{Cl}(D)_2$.

LEMMA 5.3. *The number of generators* G*, exponent* H*, and 2-invariants* $\{e_i | 1 \leqslant i \leqslant H\}$ *of* $\mathrm{Cl}(D)_2$ *satisfy*

$$G = O(\log D / \log \log D), \tag{5.1}$$

$$H = O(\log D), \tag{5.2}$$

$$\sum_{i=1}^{H} e_i = O(\log D). \tag{5.3}$$

PROOF. (5.3) implies (5.2), since each $e_i \geqslant 1$. Let $Q = \sum_{i=1}^{H} e_i$. We have

$$2^Q = |\mathrm{Cl}(D)_2| \leqslant |\mathrm{Cl}(D)| \leqslant D^3. \tag{5.4}$$

The last inequality uses the fact that each form class contains at least one reduced form, and the number of reduced forms is at most D^3 by (2.10). The bound for G is implied by Proposition 4.3 which shows that G is at most the number of distinct prime divisors of D. But an integer D has at most $O(\log D/\log\log D)$ distinct prime divisors. □

The inequality (5.4) is extremely crude. It is known that for squarefree $D > 0$, $|\mathrm{Cl}(D)| = O(D^{1/2}\log D)$ but this only improves the constant in the O-symbol in (5.3). Using genus theory one can show there are infinitely many squarefree D with $G > c_0 \log D/\log\log D$ for a small fixed constant c_0. Little is known about H other than that there are D for which it is arbitrarily large. We now prove the main theorem.

PROOF OF THEOREM 1.4. This essentially amounts to applying the complexity bounds of §4 in Theorem 5.1 and using Lemma 5.3. The initialization of the algorithm requires a generating set of ambiguous forms, and we note $M = G + 1$ using Lemma 4.7. Given a complete factorization of D it takes $O((\log D)\, M(\log D))$ elementary operations to obtain the entries of Table II and another $O((\log D)^2\, M(\log D))$ to reduce them. Turning to Theorem 5.1, we see that addition, multiplication and inversion in $\mathbf{Z}/2\mathbf{Z}$ take 1 elementary operation each, and $\log p$ disappears from (ii). Applying Lemma 5.3 gives

$$O\left((M - G)GH + \left(\sum_{i=1}^{H} e_i\right)G\right) = O\left(\frac{(\log D)^2}{\log\log D}\right)$$

and

$$O\left((M - G)H + \left(\sum_{i=1}^{H} e_i\right)\right) = O(\log D).$$

Using the complexity bounds of §4, the bottlenecks are group multiplications and square root extractions, both of which are bounded by

$$O((\log D)^4\, M(\log D)) \tag{5.5}$$

elementary operations. In fact the square root extractions are the true bottleneck, since (ii) could be sharpened in Theorem 5.1. This completes the proof. (Recall (4.1).) □

PROOF OF THEOREM 1.5. We first reduce Q_1 and Q_2 using Proposition 4.1. If $L = \mathrm{Max}(\|Q_1\|, \|Q_2\|)$ this requires at most $O((\log L)\, M(\log L))$ elementary operations. Next we obtain a basis for $\mathrm{Cl}(D)_2$. By Theorem 1.4 this requires $O((\log D)^4\, M(\log D))$ elementary operations. Finally we apply the representation algorithm to the reduced forms obtained from Q_1, Q_2. If they have identical representations, they are equivalent, otherwise not. The representation algorithm requires at most $O((\log D)^3\, M(\log D))$ elementary operations by a straightforward analysis using Theorem 5.2, Lemma 5.3 and the bounds of §4. □

PROOF OF THEOREM 1.1. By Corollary 2.3 the forms $[1, 0, -D]$ and $[-1, 0, D]$ are in $\mathrm{Cl}(D)_2$. Apply Theorem 5.5 to determine whether they are equivalent or inequivalent. This takes $O((\log D)^4 M(\log D))$ elementary operations, since $L = D$ in this case. By Proposition 2.1 we are done. □

Appendix A. Period length of certain continued fractions. This appendix constructs examples of D for which the continued fraction expansion of $\sqrt{D}$ has a long period. The method is based on principles due to Dirichlet **[10]**.

LEMMA A-1. *Suppose that $d > 1$ is squarefree and that*

$$X^2 - df^2Y^2 = -1 \tag{A.1}$$

is solvable in integers. Then

$$X^2 - df^{2k}Y^2 = -1 \tag{A.2}$$

is solvable in integers for all $k \geqslant 1$. Let (X_k, Y_k) be the minimal positive solution to (A.2) *and set $\varepsilon_k = X_k + Y_k f^k \sqrt{d}$. Suppose that $(Y_1, df) = 1$. Then*

$$\varepsilon_k = (\varepsilon_1)^{f^{k-1}}, \quad \text{for } k \geqslant 1. \tag{A.3}$$

PROOF. The equation

$$X^2 - dY^2 = -1. \tag{A.4}$$

is solvable by hypothesis, so let $\varepsilon = u + v\sqrt{d}$ denote its minimal positive solution. It is well known the complete set of positive solutions (u_n, v_n) to (A.4) is given by

$$u_n + v_n\sqrt{d} = (\varepsilon)^n \tag{A.5}$$

where n is odd. The minimal solution (X_k, Y_k) to (A.2) is given by $(u_n, v_n/f^k)$ where $n > 0$ is chosen as the least odd integer for which

$$v_n \equiv 0 \pmod{f^k}. \tag{A.6}$$

Now

$$v_n = (\varepsilon^n - \bar{\varepsilon}^n)/\sqrt{d} \tag{A.7}$$

where $\bar{\varepsilon} = u - v\sqrt{d}$. We view (A.6) as a congruence over the ring of integers of $\mathbf{Q}(\sqrt{d})$, via (A.7). This gives

$$\varepsilon^n \equiv \bar{\varepsilon}^n \left(\mathrm{mod}\left(f^k\sqrt{d}\right)\right) \tag{A.8}$$

where n is odd. Now (A.4) implies $\bar{\varepsilon} = -(\varepsilon)^{-1}$ is a unit in $\mathbf{Q}(\sqrt{d})$, so (A.8) together with n odd is equivalent to

$$\alpha^n \equiv -1 \left(\mathrm{mod}\left(f^k\sqrt{d}\right)\right) \tag{A.9}$$

where $\alpha = \varepsilon(\bar{\varepsilon})^{-1} = -\varepsilon^2$. Let n_k denote the minimal solution to (A.9) if it exists. By hypothesis n_1 exists, and by properties of the index calculus if n_{k+1} exists then $n_k | n_{k+1}$. The hypothesis $(Y_1, df) = 1$ is equivalent to

$$\alpha^{n_1} = -1 + \beta f\sqrt{d} \tag{A.10}$$

where β is an integer of $\mathbf{Q}(\sqrt{d})$ relatively prime to $f\sqrt{d}$. The conclusion of the lemma is equivalent to

$$n_k = n_1 f^{k-1}. \tag{A.11}$$

Note that f must be odd, or else (A.1) would be unsolvable.

We now establish (A.11) by induction on k. Suppose it is true that $n_k = n_1 f^{k-1}$ and that

$$\alpha^{n_k} = -1 + \beta_k f^k \sqrt{d} \tag{A.12}$$

with $(\beta_k, f\sqrt{d}) = 1$. Then

$$\alpha^{an_k} = \left(-1 + \beta_k f^k \sqrt{d}\right)^a \equiv (-1)^a \left(1 + a\beta_k f^k \sqrt{d}\right) (\mathrm{mod}(f^{k+1}d)).$$

This shows (A.9) cannot hold $(\mathrm{mod}(f^{k+1}\sqrt{d}))$ unless $f|a$. Now

$$\alpha^{fn_k} \equiv -1 + \beta_k f^{k+1} \sqrt{d} \ (\mathrm{mod}(f^{k+2}d)) \tag{A.13}$$

since f is odd and $f|\binom{f}{2}$. Hence $n_{k+1} = n_1 f^k$. Finally set

$$\beta_{k+1} = (\alpha^{fn_k} + 1)/f^{k+1}\sqrt{d}$$

and observe from (A.13) that $\beta_{k+1} \equiv \beta_k \ (\mathrm{mod}(f\sqrt{d}))$ so β_{k+1} is invertible $(\mathrm{mod}\, f\sqrt{d})$, completing the induction step. □

This lemma covers the case $d = 5, f = 5$, since

$$\varepsilon_1 = (2 + \sqrt{5})^5 = 682 + 61 \cdot 5\sqrt{5}$$

in that case. This proves (1.4) in the text.

The following result appears in Wright [**39**, p. 34].

LEMMA A-2. *Let $D > 0$ not be a square. Then the partial quotients a_j of the expansion of $\sqrt{D}$ in an ordinary continued fraction expansion satisfy*

$$0 < a_n < 2\sqrt{D}\ . \tag{A.14}$$

PROOF. Set $\alpha_0 = \sqrt{D}$, and $\alpha_n = a_n + 1/\alpha_{n+1}$, where $a_n = [\alpha_n]$. If we write

$$\alpha_n = \left(A_n + \sqrt{D}\right)/B_n \tag{A.15}$$

we have the identities

$$A_{n+1} = a_n B_n - A_n, \qquad B_{n+1} = \left(D - (A_n)^2\right)/B_n - a_n^2 B_n + 2a_n A_n$$

and

$$B_n = \left(D - (A_{n+1})^2\right)/B_{n+1}.$$

Using the induction hypothesis that $(D - A_n^2)/B_n$ is an integer, we establish by induction that A_n, B_n are all integral. Next let $\bar{\alpha}_n = (A_n - \sqrt{D})/B_n$. Now $\bar{\alpha}_0 < 0$, and since $\bar{\alpha}_n = a_n + 1/\bar{\alpha}_{n+1}$ while $a_n \geqslant 1$, we obtain $-1 < \bar{\alpha}_n < 0$ for $n \geqslant 1$ by induction on n. Since $\alpha_n > 1$

$$2A_n/B_n = \alpha_n + \bar{\alpha}_n > 0 \tag{A.16}$$

while

$$-2\sqrt{D}\ /B_n = \bar{\alpha}_n - \alpha_n < -1.$$

This last inequality shows

$$0 < B_n < 2\sqrt{D}\ . \tag{A.17}$$

Then we have

$$0 < A_n < \sqrt{D}\,, \tag{A.18}$$

the left side following from (A.16) and (A.17), the right from (A.15) and (A.17) since $\alpha_n \geqslant 1$. Putting these in (A.15) shows $a_n \leqslant (A_n + \sqrt{D})/B_n < 2\sqrt{D}$ which with $a_n \geqslant 0$ establishes (A.14). □

This lemma immediately gives upper bounds for the convergents P_n/Q_n of the continued fraction expansion of $\sqrt{D}$. For

$$p_{n+1} = a_n p_n + p_{n-1} \leqslant (2\sqrt{D} + 1)p_n,$$

$$q_{n+1} = a_n q_n + q_{n-1} \leqslant (2\sqrt{D} + 1)q_n$$

for $n \geqslant 1$. Hence

$$p_n + q_n\sqrt{D} \leqslant \sqrt{D}\,(2\sqrt{D} + 1)^n. \tag{A.19}$$

On the other hand, it is well known that if (x, y) is a positive solution to $X^2 - DY^2 = -1$ then x/y is a convergent of the continued fraction expansion of $\sqrt{D}$. Choosing a $D = df^{2k}$ with $d \geqslant 2$, $f \geqslant 3$ to which Lemma A-1 applies, we obtain

$$(\varepsilon_1)^{f^{k-1}} = \varepsilon_k \leqslant \sqrt{D}\,(2\sqrt{D} + 1)^n$$

from (A.19), which shows

$$n \geqslant \log(\varepsilon_1) f^{k-1}/\log(2\sqrt{D} + 1) - 1.$$

When $k \geqslant 2$ this yields

$$n \geqslant c(d, f) D^{1/2} (\log D)^{-1} \tag{A.20}$$

where

$$c(d, f) = 2 \log \varepsilon_1 / 3f\sqrt{d} \tag{A.21}$$

is a positive constant depending on d and f, but independent of k. In the case $D = 5^{2k+1}$,

$$c(5, 5) = 10 \log(2 + \sqrt{5})/15\sqrt{5} > \tfrac{1}{3}$$

giving the lower bound $\frac{1}{3} D^{1/2} (\log D)^{-1}$ for the period length of such $\sqrt{D}$.

REMARK. There are some weaker lower bounds known for the period length of more general classes of D. A consequence of a result of Weinberger [**38**, Theorem 4] is that, assuming the truth of the extended Riemann hypothesis, for any squarefree d there is an infinite sequence of primes $\{p_i\}$ such that as D runs through the sequence $D_i = dp_i^2$ the period lengths n_i are bounded below by $n_i \geqslant c(d)(D_i)^{1/2}(\log D_i)^{-1}$ where $c(d)$ is a positive constant depending on d only. The best lower bound for squarefree D is due to Yamamoto [**40**], who showed there is a constant $c > 0$ and an infinite sequence of squarefree D for which the period lengths of $\sqrt{D}$ exceed $c(\log D)^3$.

REFERENCES

1. L. Adleman and K. Manders, *Diophantine complexity*, (Proc. 17th IEEE Symposium on Foundations of Computer Science), Proc. IEEE (1976), 81–88.
2. _____, *Reducibility, randomness and intractability*, (Proc. 9th Annual ACM Symposium on Theory of Computing), J. Assoc. Comput. Mach. (1977), 151–163.
3. N. Ankeny, *The least quadratic nonresidue*, Ann. of Math. (2) **55** (1952), 65–72.
4. H. Bauer, *Zur Berechnung der 2-Klassenzahl der quadratischer Zahlkorper mit genau zwei verscheidenen Discriminantenprimzahlen*, J. Reine Angew. Math. **248** (1971), 42–46.
5. D. A. Burgess, *The distribution of quadratic residues and non-residues*, Mathematika **4** (1957), 106–112.
6. H. Cohn, *A second course in number theory*, Wiley, New York, 1962.
7. L. E. Dickson, *Introduction to the theory of numbers*, Univ. of Chicago Press, Chicago, 1929.
8. P. G. L. Dirichlet, *Werke*, Vol. I, 219–236.
9. _____, *De formarum binarium secundi gradus compositionae*, J. Reine Angew. Math. **47** (1854), 155–160.
10. _____, *Une propriété des formes quadratiques a determinant positif*, J. Math. Pures Appl. (3) **1** (1856), 76–79.
11. C. F. Gauss, *Disquisitiones arithmeticae*, 1801; English translation, Yale Univ. Press, New Haven, Conn., 1966.
12. M. Hall, Jr., *The theory of groups*, Macmillan, New York, 1959.
13. H. Hasse, *An algorithm for determining the structure of the 2-Sylow-subgroup of the divisor class group of a quadratic number field*, Symposia Mathematica **15** (1975), 341–352.
14. M. D. Hendy, *The distribution of ideal class numbers of quadratic fields*, Math. Comp. **29** (1975), 1129–1134.
15. L. Holzer, *Minimal solutions of Diophantine equations*, Canad. J. Math. **2** (1950), 238–244.
16. L. K. Hua, *On the least solution to Pell's equation*, Bull. Amer. Math. Soc. **48** (1942), 731–735.
17. P. Kaplan, *Sur le 2-groupe des classes d'ideaux des corps quadratiques*, J. Reine Angew. Math. **283/284** (1976), 313–363.
18. P. Kaplan and C. Sanchez, *Table des 2-groupe des classes d'ideaux de* $\mathbf{Q}(\sqrt{2p})$ *pour* $p < 2 \cdot 10^6$, U.E.R. de Mathematique, Université de Nancy I, Nancy, France, 1974.
19. D. Knuth, *Seminumerical algorithms*, Addison-Wesley, Reading, Mass., 1969.
20. J. C. Lagarias, *On the 4-rank of the class group of a quadratic field*, J. Number Theory (to appear).
21. _____, *Worst-case complexity bounds in the theory of integral quadratic forms*, J. Algorithms **1** (1980).
22. _____, *Succinct certificates for the solvability of binary quadratic Diophantine equations*, (Proc. 20th IEEE Symposium on Foundations of Computer Science), Proc. IEEE (1979), 47–54.
23. S. Lang, *Algebraic number theory*, Addison-Wesley, Reading, Mass., 1968.
24. K. Manders and L. Adleman, *NP-complete decision problems for quadratic polynomials*, (Proc. 8th ACM Conference on Theory of Computing), J. Assoc. Comput. Mach. (1976), 23–29.
25. G. B. Mathews, *Theory of numbers*, 2nd ed., Chelsea, New York, (Reprint 1961).
26. P. Morton, *On Redei's theory of the Pell equation*, J. Reine Angew Math. **307/8** (1979), 373–398.
27. W. Narkiewicz, *Elementary and analytic theory of algebraic numbers*, PWN, Warsaw, 1974.
28. J. Pollard, *Theorems on factorization and primality testing*, Proc. Cambridge Philos. Soc. **76** (1974), 521–528.
29. D. Pumplün, *Über die Klassenzahl und die Grundeinheit des reelquadratischen Zahlkorper*, J. Reine Angew. Math. **230** (1968), 167–210.
30. L. Redei, *Bedingtes Artinsches symbol mit Andwendung in der Klassenkörpertheorie*, Acta Math. Acad. Sci. Hungar. **4** (1953), 1–29.
31. _____, *Die 2-Ringklassengruppe des Quadratischen Zahlkörpers und die theorie der Pellschen Gleichung*, Acta Math. Acad. Sci. Hungar. **4** (1953), 31–87.
32. A. Scholz, *Über die Losbarkeit der Gleichung* $t^2 - Du^2 = -4$, Math. Z. **39** (1934), 93–111.
33. D. Shanks, *Class number, a theory of factorization, and genera*, (1969 Number Theory Institute), Proc. Sympos. Pure Math., vol. 20, Amer. Math. Soc., Providence, R. I., 1971, pp. 415–440.
34. _____, *Gauss's ternary form reduction and the 2-Sylow subgroup*, Math. Comput. **25** (1971), 837–853.
35. _____, *Five number-theoretic algorithms*, (Proc. 2nd Manitoba Conf. on Numerical Mathematics), 1972, 51–70.

36. H. J. S. Smith, *Report on the theory of numbers*, Chelsea, New York.

37. F. Tano, *Sur quelques theorems de Dirichlet*, J. Reine Angew. Math. **105** (1889), 160–169.

38. P. J. Weinberger, *Euclidean rings of algebraic integers*, Analytic Number Theory, Proc. Sympos. Pure Math., vol. 24, Amer. Math. Soc., Providence, R. I., 1973, pp. 321–332.

39. H. N. Wright, *First course in theory of numbers*, Wiley, New York, 1939.

40. Y. Yamamoto, *Real quadratic fields with large fundamental units*, Osaka J. Math. **8** (1971), 261–270.

41. H. Yokoi, *Units and class numbers of real quadratic fields*, Nagoya Math. J. **37** (1970), 61–65.

Department of Mathematics, University of Maryland, College Park, Maryland 20742

Current address: Room 5F-124, Bell Laboratories, Murray Hill, New Jersey 07974