ELLIPTIC CURVES AND PRIMALITY PROVING

A. O. L. ATKIN *
F. MORAIN † ‡

December 23, 1992

Abstract. The aim of this paper is to describe the theory and implementation of the Elliptic Curve Primality Proving algorithm.

Problema, numeros primos a compositis dignoscendi, hosque in factores suos primos resolvendi, ad gravissima ac utilissima totius arithmeticae pertinere, et geometrarum tum veterum tum recentiorum industriam ac sagacitatem occupavisse, tam notum est, ut de hac re copiose loqui superfluum foret.

C. F. Gauss [38, Art. 329]

1 Introduction

Primality testing is one of the most flourishing fields in computational number theory. Dating back to Gauss, the interest has recently risen with modern cryptology [16]. For quite a long time, it has been known that one could quickly recognize most composite numbers using Fermat's little theorem. For cryptographical purposes, this idea was extended and it has yielded some fast probabilistic compositeness algorithms (for this, we refer to [52], the introduction of [28] and [9]). On the contrary, testing an *arbitrary* number for primality depended on integer factorization. For this era, see [18, 92, 95]. The reader interested in large or curious primes is referred to [80] as well as [68].

The year 1979 saw the appearance of the first *general purpose* primality testing algorithm, designed by Adleman, Pomerance and Rumely [3]. The running time of the algorithm was proved to be $O((\log N)^{c \log \log \log N})$ for some effective $c > 0$. This algorithm was simplified and made practical by H. W. Lenstra and H. Cohen [28] and then successfully implemented by H. Cohen and A. K. Lenstra [27]. Motivated by our results with elliptic curves (see below), the algorithm was recently optimized by Bosma and Van der Hulst [15] (see also [60]). However, it is not possible to check the results of this algorithm independently without rewriting and re-running the entire program; by contrast our algorithm gives a "certificate" which enables a second programmer to verify our proof in a time much shorter than the original time.

* Department of Mathematics, University of Illinois at Chicago Box 4348, Chicago, IL 60680, USA

† Projet ALGO, Institut National de Recherche en Informatique et en Automatique, Domaine de Voluceau, B. P. 105, 78153 LE CHESNAY CEDEX (France) & Département de Mathématique, Université Claude Bernard, 69622 Villeurbanne CEDEX

‡ On leave from the French Department of Defense, Délégation Générale pour l'Armement.

In 1985, H. W. Lenstra (Jr.) introduced the use of elliptic curves in factorization. There was then hope to find a similar use for primality testing. This was first done by Goldwasser and Kilian [40] using the architecture of the DOWNRUN algorithm of [97] together with a theoretical algorithm due to Schoof [83]. They found that this algorithm recognizes primes in expected random polynomial time, at least assuming some very plausible conjectures in analytic number theory. Almost simultaneously, the first author [4] designed a practical algorithm based on the same ideas, but using results from the theory of elliptic curves over finite fields (see also [25] and [14] for a first insight). From a practical point of view, this algorithm is faster and yields a proof that the computation is correct in the form of a list of numbers by means of which one can easily check the primality properties (see Section 10). In another direction, the theory of *elliptic pseudoprimality* tests and *elliptic pseudoprimes* was introduced [41, 61, 8].

Shortly afterwards, Adleman and Huang announced [2] that they designed a primality testing algorithm using curves of genus two whose expected running time is also polynomial, but without any unproven hypothesis. As for now, it seems that this algorithm has not been implemented.

The purpose of this paper is to describe the test due to the first author (which is known as the Elliptic Curve Primality Proving —ECPP— algorithm), together with the implementations made by the authors (other implementations include that of D. Bernardi for the class number one case and more recently that of Kaltofen and Valente [49] and that of Vardi (personal communication, August 1989) for the MATHEMATICA system).

Since there are considerable differences of detail between the implementations of the two authors, we have decided for the sake of clarity to present the algorithm solely as implemented by the second author. We make a few historical remarks in Section 8.1.

The plan of the paper is as follows. In Section 2, we recall some well known properties of quadratic forms and fields necessary for presenting Section 3, which deals with the theory of Hilbert class fields of imaginary quadratic fields via modular forms. At this point, we introduce Weber's functions as well as Dedekind's η. In Section 4, we present the relevant theory of elliptic curves in a manner similar to that of [57]: This unified approach is well suited for our purpose, which goes from classical elliptic curves over $\mathbf{C}$ to curves over a finite field. Section 5 is concerned with primality testing using elliptic curves as used by Goldwasser and Kilian on the one hand and the first author in his designing the ECPP algorithm on the other. A path towards analyzing ECPP is made in Section 6: We present some heuristic arguments concerning the ability for a number to be good with respect to ECPP as well as the probability of failure of a weak version of ECPP. In Section 7, we develop an efficient algorithm for constructing the Hilbert class field of an imaginary quadratic field by means of the functions introduced in Section 3. At this point, we introduce the concept of *Weber polynomials* and we detail a fast algorithm to compute the factorization of Weber polynomials over their genus field. In Section 8, we detail the computational routines we use in the implementation: Section 9 contains some typical running times for numbers of less than 300 digits and also some running times for larger numbers, most of all taken from [19] or discovered by the authors. Section 10 is briefly concerned with the second problem mentioned above, namely that of the *actual* proof we get by ECPP.

Notation. Throughout the paper, N will denote a probable prime, which means that N was not declared composite by any of the probabilistic primality testing algorithms which were used.

Historical note. The basic algorithm was designed and implemented by the first author in 1986. In 1987, the second author implemented a version of the algorithm based on a paper of Cohen [26]. In May 1989, the two authors met and merged part of their ideas to come up with the present paper, a longer version of which is available as [6].

2 Some properties of quadratic forms and fields

Our aim is to recall basic properties of quadratic forms and fields that are necessary for the following sections. We introduce first quadratic forms that are easy to compute with and then quadratic fields that are well suited for explaining the theory. These are two sides of the same object.

2.1 Quadratic forms

The following results are well known and can be found in [35, 30]. Let $-D$ be a fundamental discriminant, i.e., D is a positive integer which is not divisible by any square of an odd prime and which satisfies $D \equiv 3 \bmod 4$ or $D \equiv 4, 8 \bmod 16$. We can factor $-D$ as $q_1^* \cdots q_t^*$, where $q^* = (-1)^{(q-1)/2} q$ if q is an odd prime and -4 or ± 8 otherwise. In the sequel, the q_i's are supposed to be ordered as follows: if $D \equiv 0 \bmod 4$, then $q_1 = 4$ or 8. Then the q's with $q^* = q$ are listed in increasing order and finally the q's with $q^* = -q$, also in increasing order. We put $l = \#\{i, q_i^* = q_i\}$. It is easy to see that

$$t - l - 1 \equiv 0 \bmod 2. \tag{1}$$

A *quadratic form* of discriminant $-D$ is a 3-tuple of integers (a, b, c) such that $b^2 - 4ac = -D$. There is a correspondence between the set of quadratic forms and the set of 2×2 matrices with half integer coefficients. With $Q = (a, b, c)$, we associate the 2×2 matrix

$$M(Q) = \begin{pmatrix} a & b/2 \\ b/2 & c \end{pmatrix}.$$

Two forms Q and Q' of the same discriminant are said to be *equivalent* (or $Q \sim Q'$) if there exists N in $\mathbf{SL}_2(\mathbf{Z})$ (i.e., a 2×2 integer matrix with determinant 1) such that:

$$M(Q') = N^{-1} M(Q) N.$$

This clearly defines an equivalence relation on quadratic forms. It can be shown that:

Proposition 2.1 *Each equivalence class contains exactly one form (a, b, c) with a, b, c relatively prime and satisfying $|b| \leq a \leq c$ and ($|b| = a$ or $a = c \Rightarrow b > 0$). Such a form is called* reduced.

There is an algorithm that computes a reduced form equivalent to a given form: we refer to the literature for this [85].

The set of primitive reduced quadratic forms of discriminant $-D$, denoted by $\mathcal{H}(-D)$, is finite (for $|b| \leq \sqrt{D/3}$ if (a, b, c) is reduced). Moreover, it is possible to define an operation on classes that gives to $\mathcal{H}(-D)$ the structure of an Abelian group. This operation is called the *composition of classes* and is ordinarily written multiplicatively. For the actual computation, we refer to [85]. The order of $\mathcal{H}(-D)$ is denoted by $h(-D)$. The neutral element F_D is called the *principal form*. It is equal to $(1, 0, D/4)$ or $(1, 1, (D + 1)/4)$ according as $D \equiv 0$ or $3 \pmod 4$.

Let $C = (a, b, c)$ be an element of $\mathcal{H}(-D)$. For (x, y) in $\mathbf{Z}^2$, put $C(x, y) = ax^2 + bxy + cy^2$ and assume that a is prime to D (otherwise consider c instead of a, since a and c cannot both have a common factor with D). Let p be a rational prime. The equation $p = C(x, y)$ has a solution in (x, y) only if the following conditions are satisfied:

$$\left(\frac{-D}{p}\right) = +1, \text{ and } \left(\frac{p}{q_i}\right) = \left(\frac{a}{q_i}\right), 1 < i \leq t. \tag{2}$$

(*Hint:* write $4ap = (2ax + by)^2 + Dy^2$.)

Put $\chi_i(a) = \chi_i(C) = (a/q_i)$ for all i. This defines a map from $\mathcal{H}(-D)$ to $Z_t = \{\pm 1\}^t$ by:

$$\begin{array}{rccl} \Xi: & \mathcal{H}(-D) & \rightarrow & Z_t \\ & C & \mapsto & (\chi_1(C), \ldots, \chi_t(C)). \end{array} \tag{3}$$

The following theorem was proven by Gauss:

Theorem 2.1 *The map Ξ is onto: If we start from $\varepsilon = (\epsilon_1, \ldots, \epsilon_t)$ satisfying $\prod_i \epsilon_i = +1$, we can find a C such that $\Xi(C) = \varepsilon$. Moreover, Ξ is a homomorphism. The associated cosets are called the* genera *and they inherit the group law. Each coset has cardinality $e = h/g$ where $g = 2^{t-1}$.*

We define the *principal genus* as $G_0 = \Xi^{-1}(+1, \ldots, +1)$. For each genus G_i, we can find C_i in $\mathcal{H}(-D)$ such that $G_i = C_i G_0$. Thus the product of the genera $G_i = C_i G_0$ and $G_j = C_j G_0$ is G_k with $C_k = C_i \cdot C_j$.

A prime p which is representable by a form of G_i is said to belong to G_i (this is denoted by $p \in G_i(-D)$).

2.2 Quadratic fields

Consider now $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$. The extension $\mathbf{K}/\mathbf{Q}$ is Abelian of degree 2, of Galois group $\{1, \tau\}$, where τ denotes complex conjugation. The ring of integers of $\mathbf{K}$ is $\mathcal{O}_K = \mathbf{Z}[\omega]$, where

$$\omega = \begin{cases} \sqrt{-D/4} \text{ if } D \equiv 0 \bmod 4 \\ \frac{1+\sqrt{-D}}{2} \text{ otherwise.} \end{cases}$$

The conjugate of an element $\alpha = x + y\omega$ is $\alpha' = \tau(\alpha) = x + y\tau(\omega)$. The trace (resp. the norm) of α is $T_K(\alpha) = \alpha + \tau(\alpha)$ (resp. $N_K(\alpha) = \alpha\tau(\alpha)$). If α is an element of $\mathbf{K}$, its *associates* are the $v\alpha$ where v is any unit of $\mathbf{K}$ (that is $N_K(v) = 1$). The number of units is denoted by $w(-D)$ and is equal to 6, 4 or 2 according to D equal to 3, 4 or > 4.

The decomposition of the ideal (p) in $\mathbf{K}$ is given by the following theorem:

Proposition 2.2 *If $(-D/p) = +1$, the ideal (p) splits as the product of two distinct ideals in $\mathbf{K}$. If $(-D/p) = 0$, (p) ramifies, and if $(-D/p) = -1$, it is inert.*

We conclude this section with:

Proposition 2.3 *The equation $p = N_K(\pi)$ has a solution in $\mathcal{O}_K$ if and only if (p) splits as the product of two* principal *ideals in* $\mathbf{K}$. *This is equivalent to saying that p is represented by the principal form of discriminant $-D$. In other words: $4p = A^2 + DB^2$ with A and B in* $\mathbf{Z}$.

If p is representable by the principal form of discriminant $-D$, we shall say that "p is a norm in $\mathbf{Q}(\sqrt{-D})$" or simply "p is a norm" when the context is clear. Conversely, we shall say that "$-D$ is good for p" if p is a norm. Thus in general $(-D/p) = 1$, that p splits in $\mathbf{Q}(\sqrt{-D})$, and even that p is representable by a form of the principal genus, are all necessary conditions for p to be a norm.

2.3 Genus field

The genus field of $\mathbf{K}$ is $\mathbf{K}_{\mathcal{G}} = \mathbf{Q}(\sqrt{q_1^*}, \ldots, \sqrt{q_t^*})$, the q_i being described above. The field $\mathbf{K}_{\mathcal{G}}$ is the maximal unramified Abelian extension of $\mathbf{Q}$ containing $\mathbf{K}$. The Galois group of $\mathbf{K}_{\mathcal{G}}/\mathbf{Q}$ is isomorphic to $(\mathbf{Z}/2\mathbf{Z})^t$.

We recall that the Artin symbol associated with the quadratic form C (in fact with the genus G containing C) is (see [29]):

$$\mathcal{A}_G = \left(\frac{\mathbf{K}_{\mathcal{G}}/\mathbf{K}}{\mathfrak{p}}\right) \simeq (\chi_1(G), \ldots, \chi_t(G)),$$

with $\chi_i(G) = (q_i^*/p)$, where $(p) = \mathfrak{p}\mathfrak{p}'$ is any prime number represented by a form of G and $\mathfrak{p}$ the ideal above p in $\mathbf{K}$.

3 Modular forms

3.1 The modular group and the modular invariant j

We follow [84]. The *modular group* is defined to be $\Gamma = \mathbf{SL}_2(\mathbf{Z})/\{\pm 1\}$. An element $g = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ of Γ acts on $\mathbf{H} = \{z \in \mathbf{C}, \mathrm{Im}(z) > 0\}$ by

$$gz = \frac{az+b}{cz+d}.$$

It is known that Γ is generated by S and T where

$$S = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, T = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

A *modular form* of weight $2k$ (k any integer) is a function meromorphic everywhere on $\mathbf{H}$ and at infinity, satisfying

$$\forall \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathbf{SL}_2(\mathbf{Z}), \forall z \in \mathbf{H}, f(z) = (cz+d)^{-2k} f\left(\frac{az+b}{cz+d}\right).$$

If the form is holomorphic everywhere (which implies $k > 0$ for non-constant forms), we say that the form is *regular*.

Let $L(1, \omega) = \mathbf{Z} + \omega\mathbf{Z}$ be a lattice in $\mathbf{C}$ ($\omega \in \mathbf{H}$). Put

$$G_{2k}(L) = \sum_{(m,n)\neq(0,0)} \frac{1}{(m\omega + n)^{2k}},$$

for $k > 1$. If $k > 1$, then $G_{2k}(L)$ is a regular modular form of weight $2k$. We put $g_2(L) = 60G_4$, $g_3(L) = 140G_6$ and $\Delta = g_2^3 - 27g_3^2$: these are regular modular forms of weight 4, 6 and 12 respectively. The *modular invariant* is then

$$j = 12^3 \, \frac{g_2^3}{\Delta}.$$

We have

Proposition 3.1 *The function j is a modular function (i.e., a modular form of weight 0), is holomorphic in $\mathbf{H}$ and has a simple pole at infinity. The function j is a complex analytic isomorphism from $\mathbf{H}/\Gamma$ to $\mathbf{C}$.*

One can show that the q-expansion of j is (Cf. [84]):

$$j(q) = \frac{1}{q} + 744 + \sum_{n\geq 1} c_n \, q^n, \tag{4}$$

where the c_n are positive integers. For a survey of the arithmetical and numerical properties of the c_n, see for instance [84, 63].

3.2 Complex multiplication for lattices

Let $L = L(1, \omega)$ be a lattice in $\mathbf{C}$. Put $M(L) = \{\alpha \in \mathbf{C}, \alpha L \subset L\}$. It is clear that $\mathbf{Z} \subset M(L)$. When $M(L)$ is greater than $\mathbf{Z}$, we say that L has *complex multiplication*. It can be shown [53, chapter 1] that if L has complex multiplication then ω belongs to a complex quadratic field $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$. Then $M(L)$ is an order of $\mathbf{K}$, that is a ring which is a free submodule of rank 2 over $\mathbf{Z}$ of $\mathcal{O}_K$ the ring of integers of $\mathbf{K}$.

3.3 Class field theory of imaginary quadratic fields

Class field theory is one of the most remarkable achievements of mathematics. One of its motivating problem was the construction of the maximal unramified Abelian extension of an imaginary quadratic field (for a modern presentation of the classical approach, see [13]). An algebraic treatment was given by Deuring [34]. The theory was generalized in [87]. In the present paper, we only need to use a comparatively small part of the theory, which we specify below.

Let $-D$ be a fundamental discriminant and $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$. The *Hilbert Class Field* of $\mathbf{K}$ is the maximal unramified Abelian extension of $\mathbf{K}$ and is denoted by $\mathbf{K}_{\mathrm{H}}$(see [34]). We have (see [13, 91]):

Theorem 3.1 *The field* $\mathbf{K}_{\mathrm{H}}$ *can be obtained by adjoining to* $\mathbf{K}$ *any value* $j_r = j(\omega_r)$, *where* ω_r *is the complex number* associated with C_r, *i.e.,* $\omega_r = \omega(C_r) = (-b_r + i\sqrt{D})/(2a_r)$ *with* $C_r = (a_r,\, b_r,\, c_r)$ *in* $\mathcal{H}(-D)$. *The minimal polynomial of the* j_r*'s is denoted by* $H_D(X)$. *It follows that* $\mathbf{K}_{\mathrm{H}}$ *is precisely the splitting field of* $H_D(X)$.

The Galois group Σ_H *of* $\mathbf{K}_{\mathrm{H}}/\mathbf{K}$ *is isomorphic to* $\mathcal{H}(-D)$. *If* C *is an element of* $\mathcal{H}(-D)$, *the corresponding element* σ_C *of* Σ_H *acts on* $j(C')$ *by:*

$$\sigma_C(j(C')) = j(C^{-1} \cdot C'). \tag{5}$$

We also require the following (see [31, 33]):

Theorem 3.2 *A rational prime* p *is a norm in* $\mathbf{K}$ *if and only if* (p) *splits completely in* $\mathbf{K}_{\mathrm{H}}$. *This is equivalent to saying that* $H_D(X) \pmod p$ *has only simple roots and they are all in* $\mathbf{Z}/p\mathbf{Z}$. *Moreover we have that*

$$4p = A^2 + DB^2$$

has a solution in rational integers (A, B) *if and only if* $H_D(X)$ *splits completely modulo* p.

The last statement follows from Proposition 2.3 .

3.4 Dedekind's and Weber's functions

Let z be any complex number and put $q = \exp(2i\pi z)$. Dedekind's η function is defined by [91, §24 p. 85]

$$\eta(z) = q^{1/24} \prod_{m \geq 1} (1 - q^m). \tag{6}$$

We can expand η as [91, §34 p. 112]

$$\eta(z) = q^{1/24} \left(1 + \sum_{n \geq 1} (-1)^n (q^{n(3n-1)/2} + q^{n(3n+1)/2}) \right). \tag{7}$$

The function η is a modular form of weight 1/2 with a complicated multiplier function.

Letting ζ_n stand for $\exp(2i\pi/n)$, the Weber functions are [91, §34 p. 114]

$$f(z) = \zeta_{48}^{-1} \, \frac{\eta((z+1)/2)}{\eta(z)}, \tag{8}$$

$$f_1(z) = \frac{\eta(z/2)}{\eta(z)}, \tag{9}$$

$$f_2(z) = \sqrt{2}\, \frac{\eta(2z)}{\eta(z)}, \tag{10}$$

and [91, §54 p. 179]

$$\gamma_2 = \frac{f^{24} - 16}{f^8}, \tag{11}$$

$$\gamma_3 = \frac{(f^{24} + 8)(f_1^8 - f_2^8)}{f^8}. \tag{12}$$

We can reconstruct the modular invariant j through [91, §54 p. 179]:

$$j(z) = \frac{(f^{24}-16)^3}{f^{24}} = \frac{(f_1^{24}+16)^3}{f_1^{24}} = \frac{(f_2^{24}+16)^3}{f_2^{24}} = \gamma_2^3 = \gamma_3^2 + 1728. \quad (13)$$

We also note the following transformation formulas [91, §34 p. 113]. First

$$\eta(z+1) = \zeta_{24}\, \eta(z),\ \eta(-1/z) = \sqrt{z/i}\, \eta(z) \quad (14)$$

from which

$$f(z+1) = \zeta_{48}^{-1}\, f_1(z),\ f_1(z+1) = \zeta_{48}^{-1}\, f(z),\ f_2(z+1) = \zeta_{24}\, f_2(z) \quad (15)$$

and

$$f(-1/z) = f(z),\ f_1(-1/z) = f_2(z),\ f_2(-1/z) = f_1(z). \quad (16)$$

4 Elliptic curves

4.1 Definition

We follow Lenstra [57]. Let $\mathbf{k}$ be a field of characteristic 0 or prime to 6. Let $\mathbf{P}^2(\mathbf{k})$ be the projective plane over $\mathbf{k}$. The equivalence class containing (x, y, z) is denoted by $(x : y : z)$.

An *elliptic curve* is a pair $E = (a, b)$ (which we sometimes write as $E(a, b)$) of elements of $\mathbf{k}$ such that $\Delta_E = -16(4a^3 + 27b^2) \neq 0$. This quantity is called the *discriminant* of the curve E. We also define the *invariant* of the curve $j(E) = 2^8 3^3 a^3/(4a^3 + 27b^2)$.

The set of points of E over $\mathbf{k}$ is:

$$E(\mathbf{k}) = \{(x : y : z) \in \mathbf{P}^2(\mathbf{k}), y^2 z = x^3 + axz^2 + bz^3\}.$$

There is exactly one point of $E(\mathbf{k})$ with $z = 0$, namely $(0 : 1 : 0)$ called the *point at infinity*, denoted by O_E. The set $E(\mathbf{k})$ can be made an Abelian group with an operation denoted by $+$ using the *tangent-and-chord* method. Suppose temporarily that $\mathbf{k} = \mathbf{R}$. Then $E(\mathbf{R})$ is a projective curve that we can look at. In order to add two points M_1 and M_2 resulting in M_3, we draw the line $M_1 M_2$ (or the tangent if $M_1 = M_2$). This line intersects E in a third point, P, whose reflexion in the x-axis yields the sum $M_3 = M_1 + M_2$. The symmetric of a point $M = (x : y : z)$ is $-M = (x : -y : z)$ and the neutral element is the point at infinity. From a practical point of view, the coordinates of a non trivial M_3 are

$$\begin{cases} x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= \lambda(x_1 - x_3) - y_1 \end{cases}$$

where

$$\lambda = \begin{cases} (y_2 - y_1)(x_2 - x_1)^{-1} & \text{if } x_2 \neq x_1 \\ \\ (3x_1^2 + a)(2y_1)^{-1} & \text{otherwise.} \end{cases}$$

We can compute kP using the binary method [52] (see also [27]) or addition-subtraction chains [71].

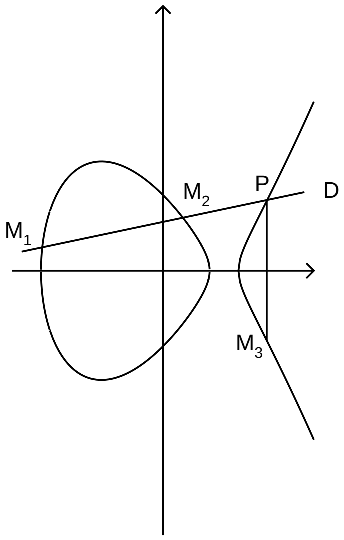

Figure 1: An elliptic curve over $\mathbf{R}$.

The same equations are used to define the group law for arbitrary $\mathbf{k}$.

An isomorphism between $E(a, b)$ and $E(a', b')$ is defined to be an element u in $\mathbf{k}^\times$ such that $a' = u^4 a$ and $b' = u^6 b$. Such an isomorphism induces an isomorphism between $E(a, b)(\mathbf{k})$ and $E(a', b')(\mathbf{k})$ by sending $(x : y : z)$ to $(u^2 x : u^3 y : z)$. An automorphism of E is an isomorphism from E to E. The group of automorphism has at most six elements [57]. For most of the curves, it is of order 2.

4.2 $\mathbf{k} = \mathbf{Z}/p\mathbf{Z}$

Let p be a prime number greater than 3. Let E be an elliptic curve defined over $\mathbf{Z}/p\mathbf{Z}$. We do not intend to explain Deuring's work concerning its reduction modulo p, but the interested reader may consult [53, chapter 13] and the references given there. It can be shown that E can be described as the reduction modulo p of an elliptic curve $E(\mathbf{C})$ with complex multiplication by an order of a quadratic field $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$.

From a practical point of view, one can construct a curve which has complex multiplication by the ring of integers of $\mathbf{Q}(\sqrt{-D})$ in the following way. Suppose that p is a norm in $\mathbf{K}$: $(p) = \mathfrak{p}\mathfrak{p}' = (\pi)(\pi')$ and $\mathfrak{p}$ splits completely in $\mathbf{K}_{\mathrm{H}}$ as $\mathfrak{p} = \mathfrak{P}_1 \cdots \mathfrak{P}_h$ (with $h = h(-D)$). Then the polynomial $H_D(X)$ splits completely modulo p. Let j be any root of H_D modulo p and E an elliptic curve of invariant j. Then $\#E(\mathbf{Z}/p\mathbf{Z}) = p+1-\mathrm{Tr}_K(\pi) = p+1-A$ with $|A| < 2\sqrt{p}$ (this theorem was originally proved by Hasse) and E has complex multiplication by the ring $\mathcal{O}_K$.

Concerning the structure of $E(\mathbf{Z}/p\mathbf{Z})$ as an Abelian group, we have [22]:

Theorem 4.1 *The group $E(\mathbf{Z}/p\mathbf{Z})$ is either cyclic or the product of two cyclic groups of order m_1 and m_2 that satisfy:*

$$m_1 | m_2, \ m_1 | \gcd(m, p-1), \tag{17}$$

where $m = \#E(\mathbf{Z}/p\mathbf{Z})$.

5 Primality testing

5.1 Traditional primality testing and the DOWNRUN process

Before the advent of the Jacobi sums algorithm, the main method for primality testing was to use some known factors of $N^t - 1$, $t = 1, 2, 3, 4, 6$, involving either some converse of Fermat's theorem, or Lucas sequences or a generalization thereof.

The simplest way to prove that an odd number N is prime is to prove that the group $(\mathbf{Z}/N\mathbf{Z})^\times$ is cyclic (and that N is not a prime power). For this, we need only to exhibit a generator of this group. This yields the following theorem. (This is not the optimal theorem, but we cite it for the sake of simplicity.)

Theorem 5.1 *If there exists an a prime to N such that*

$$a^{N-1} \equiv 1 \bmod N$$

but

$$a^{(N-1)/q} \not\equiv 1 \bmod N$$

for every prime divisor q of $N-1$, then N is prime.

Of course, we need to factor $N-1$. Starting with a number N_0, a favorable situation occurs whenever we can completely factor $N_0 - 1$ or we find that $N_0 - 1$ has a large factor N_1 which is probably prime: such a number N_0 we call *probably factored*. The problem is then reduced to proving that N_1 is prime. Also, we can use some factors of $N_i^t - 1$ to help us in our job.

This idea forms the DOWNRUN process of [97]: Build a decreasing sequence of probable primes $N_0 > N_1 > \cdots > N_k$ such that the primality of N_{i+1} implies that of N_i (see [52, pp. 376–377]). Indeed, this is a *factor and conquer* method. The problem is that for each N_i, there is only a limited number of candidates that we can try to factor. We will see that this difficulty is overcome when using elliptic curves.

5.2 The Goldwasser-Kilian algorithm

From [40], we have:

Theorem 5.2 *Let N be an integer prime to 6, E an elliptic curve over $\mathbf{Z}/N\mathbf{Z}$, together with a point P on E and m and s two integers with $s \mid m$. For each prime divisor q of s, we put $(m/q)P = (x_q : y_q : z_q)$. We assume that $mP = O_E$ and $\gcd(z_q, N) = 1$ for all q. Then, if p is a prime divisor of N, one has $\#E(\mathbf{Z}/p\mathbf{Z}) \equiv 0 \bmod s$.*

We have also:

Corollary 5.1 *With the same conditions, if $s > (\sqrt[4]{N} + 1)^2$, then N is prime.*

Combining this theorem with Schoof's algorithm that computes $\#E(\mathbf{Z}/p\mathbf{Z})$ in time $O((\log p)^{8+\varepsilon})$ (see [56]), we obtain the Goldwasser-Kilian algorithm.

procedure GK(N)

1. choose an elliptic curve E over $\mathbf{Z}/N\mathbf{Z}$, for which the number of points m (computed with Schoof's algorithm) satisfies $m = 2\,q$, with q a probable prime;
2. if $(E,\, m)$ satisfies the conditions of the theorem with $s = m$, then N is prime, otherwise it is composite;
3. the primality of q is proved in the same way;
4. **end.**

We see that we have solved one of the problem arising in the ordinary DOWNRUN: this time, we have a lot of numbers which we can try to factor.

The problem with GK is that Schoof's algorithm seems almost impossible to implement (however, see [5]). We will use instead the properties of elliptic curves over finite fields related to complex multiplication.

5.3 The ECPP algorithm

In algorithm GK, we begin by searching for a curve and then compute its number of points. Here, we do exactly the contrary. We get:

procedure ECPP(N);
(* N is a probable prime *)

1. set $i := 0, N_0 := N$;
2. **while** $N_i > N_{small}$
    1. find a fundamental discriminant $-D_i$ which is good for N_i; in other words $N_i = \pi_i \pi_i'$ in $\mathbf{K} = \mathbf{Q}(\sqrt{-D_i})$ (see Section 2);
    2. if one of the $w(-D_i)$ numbers $m_1, \cdots, m_w$ ($m_r = N_K(\upsilon_r \pi_i - 1)$ where υ_r is a unit in $\mathbf{K}$) is probably factored go to step 2.3 else go to 2.1;
    3. store $\{i, N_i, D_i, \upsilon_r \pi_i, m_r, F_i\}$ where $m_r = F_i N_{i+1}$. Here F_i is a completely factored integer and N_{i+1} a probable prime; set $i := i + 1$ and go to step 2.1;
    4. compute a root j of $H_{D_i}(X) \equiv 0 \bmod N_i$;
    5. compute an equation of the curve E_i of invariant j and whose cardinality modulo N_i is m_i;
    6. find a point P_i on the curve E_i;
    7. check the conditions of the theorem with $s = N_{i+1}$ and $m = m_i$: in other words, check that $Q_i = F_i P_i \neq O_{E_i}$ but $sQ_i = O_{E_i}$;
3. **end.**

Finding m_r which is probably factored will be referred to as "finding a suitable m".

6 Analysis

6.1 Theoretical results

The running time of GK is analyzed in the following theorems [40, 56].

Theorem 6.1 *Suppose that there exist two positive constants c_1 and c_2 such that the number of primes in the interval $[x;\, x+\sqrt{2\,x}]$ $(x \geq 2)$ is greater than $c_1\sqrt{x}\,(\log x)^{-c_2}$. Then GK proves the primality of N in expected time $O((\log N)^{10+c_2})$.*

Theorem 6.2 *There exist two positive constants c_3 and c_4 such that for all $k \geq 2$, the proportion of prime numbers N of k bits for which the expected time of GK is bounded by $c_3(\log N)^{11}$ is at least $1 - c_8\, 2^{-k^{\frac{1}{\log\log k}}}$.*

As for ECPP, we only have the heuristic analysis cited in [54]. The authors find that the running time of the algorithm is roughly $O((\log N)^{6+\epsilon})$ for some $\epsilon > 0$.

The remaining of this section is devoted to some practical considerations concerning ECPP.

6.2 What is a good discriminant?

Let p be a prime number. Then p is a norm in $\mathbf{Q}(\sqrt{-D})$ if and only if p is represented by the principal form of $\mathcal{H}(-D)$. As in Section 2, let $-D = q_1^* \cdots q_t^*$, its class number is $h = h(-D)$ and the number of genera is $g = 2^{t-1}$. The prime p is represented by a form of G_0 if and only if $\forall i, \chi_i(p) = +1$ (see Section 2), which occurs with probability $1/2^t$. Given this, p is represented by F_D with conditional probability g/h. We deduce

Proposition 6.1 *A prime p is represented by F_D with probability $1/(2h)$.*

A proof with less handwaving can be found in [33, Chapter 8].

6.3 Practical considerations: good and bad numbers

For practical purposes, we are only interested in fundamental discriminants D $(D < 10^6)$ with $h(-D) \leq 50$ (the parameters 10^6 and 50 are somewhat arbitrary, and represent the extreme limits of what we expect to need). They form a set $\mathcal{D}$. We have (presumably) that $\#\mathcal{D} = 10628$. Let H and G be two integers. We write $ND(H,G)$ for the number of D in $\mathcal{D}$ for which $h(-D) = H$ and $g(-D) = G$. In Table 1, we indicate the values of $ND(H,G)$ for $H \leq 50$ (they agree with that of [20]). From this, we can deduce the number of D such that $h(-D) \leq 50$ and with given value of H/G. This quantity represents the degree of the final polynomial of which we want a root and its inverse (G/H) is just the probability that N is a norm in $\mathbf{K}$ (provided that $(-D/N) = +1$). This yields Table 2.

Let S be a finite set of primes (here 4 and 8 are assumed to be distinct primes). We define $N_p(S)$ to be the number of D in $\mathcal{D}$ which are divisible by at least one prime of S: This quantity is tabulated in Table 3. From the above results, it is quite clear that bad numbers are those which are quadratic nonresidue modulo small primes, such as $N \equiv -1 \bmod 12$, which kill off one third of our discriminants. As an example, it is interesting to compute the smallest prime which does not split in any of the quadratic fields with class number 1. This number is 3167 (the next one is 607823).

H	G	D_{min}	D_{max}	#	H	G	D_{min}	D_{max}	#
1	1	3	163	9	27	1	983	103387	93
2	2	15	427	18	28	2	831	126043	174
3	1	23	907	16	28	4	935	106723	283
4	2	39	1555	30	29	1	887	166147	83
4	4	84	1435	24	30	2	671	134467	255
5	1	47	2683	25	31	1	719	133387	73
6	2	87	3763	51	32	2	791	164803	187
7	1	71	5923	31	32	4	1239	136843	333
8	2	95	5947	62	32	8	3080	89947	173
8	4	260	6307	56	32	16	7140	40755	15
8	8	420	3315	13	33	1	839	222643	101
9	1	199	10627	34	34	2	1079	189883	219
10	2	119	13843	87	35	1	1031	210907	103
11	1	167	15667	41	36	2	959	217627	271
12	2	327	17803	88	36	4	1295	175123	397
12	4	231	15283	118	37	1	1487	158923	85
13	1	191	20563	37	38	2	1199	289963	237
14	2	215	30067	95	39	1	1439	253507	115
15	1	239	34483	68	40	2	1271	260947	251
16	2	407	31243	101	40	4	2255	250387	438
16	4	399	27307	160	40	8	2415	148603	223
16	8	1140	16555	60	41	1	1151	296587	109
16	16	5460	5460	1	42	2	1959	280267	339
17	1	383	37123	45	43	1	1847	300787	106
18	2	335	48427	150	44	2	1391	319867	261
19	1	311	38707	47	44	4	2135	319243	430
20	2	776	58507	150	45	1	1319	308323	154
20	4	455	43747	200	46	2	2615	308947	267
21	1	431	61483	85	47	1	3023	375523	107
22	2	591	85507	139	48	2	1751	333547	343
23	1	647	90787	68	48	4	3615	335203	621
24	2	695	111763	167	48	8	4935	275587	355
24	4	759	62155	240	48	16	11220	94395	46
24	8	2184	42427	104	49	1	1511	393187	132
25	1	479	93307	95	50	2	1799	389467	345
26	2	551	103027	190					

Table 1: Statistics on the discriminants D with $h(-D) \leq 50$

H/G	$\#(H/G)$	H/G	$\#(H/G)$	H/G	$\#(H/G)$	H/G	$\#(H/G)$	H/G	$\#(H/G)$
1	65	6	683	11	610	16	187	21	424
2	161	7	409	12	788	17	264	22	261
3	335	8	434	13	227	18	271	23	335
4	395	9	581	14	174	19	284	24	343
5	535	10	588	15	323	20	251	25	440

Table 2: Number of discriminants D with given H/G for $H \leq 50$.

S	$N_p(S)$	S	$N_p(S)$	S	$N_p(S)$	S	$N_p(S)$
$\{3\}$	2495	$\{3,4\}$	3669	$\{3,5\}$	3825	$\{3,4,5\}$	4803
$\{4\}$	1540	$\{5\}$	1744	$\{4,5\}$	3020	$\{3,4,5,7,8\}$	6382

Table 3: Divisibility of the discriminants

6.4 A theoretical failure case

Corollary (5.1) cannot be applied when $s \leq (\sqrt[4]{p}+1)^2$. In particular, we cannot use it when the number of points, m, is a perfect square *and* $E(\mathbf{Z}/p\mathbf{Z})$ is isomorphic to $(\mathbf{Z}/M\mathbf{Z}) \times (\mathbf{Z}/M\mathbf{Z})$ with $m = M^2$. A necessary condition for that is

$$M \mid p-1. \tag{18}$$

We also have

$$\sqrt{p}-1 \leq M \leq \sqrt{p}+1,$$

by Hasse's theorem. Put $\lfloor \sqrt{p} \rfloor = a$ and $p = a^2 + r$, with

$$0 < r < 2a+1. \tag{19}$$

Then

$$a \leq M \leq a+1. \tag{20}$$

Suppose first that $M = a$. Then (18) implies

$$a \mid a^2 + r - 1,$$

that is $a|r-1$. There are two cases. First when $r = 1$, one has $p = a^2+1$ and E has complex multiplication by $\mathbf{Q}(\sqrt{-D})$, with $-D = (m-p-1)^2 - 4p = -4a^2$. When $r > 1$, (19) implies $r - 1 = a$ and thus $p = a^2 + a + 1$. It is then easy to see that E has complex multiplication by $\mathbf{Q}(\sqrt{-3})$.

7 Precomputations

This rather lengthy section deals with the effective construction of the Hilbert Class Field of $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$. This will be done using j and other modular functions, especially Weber's *class invariants*. For this purpose, we introduce the following notation. Let u be any complex function. We will denote by $H_D[u](X)$ the minimal polynomial of $u(\omega)$ over $\mathbf{Q}$ (remember that $\mathcal{O}_K = \mathbf{Z}[\omega]$ where ω has been defined in Section 2). When $u = j$, we will abbreviate $H_D[u]$ to H_D.

7.1 Hilbert polynomials

The determination of j as an algebraic integer in $\mathbf{Q}(j)$ has been studied by many authors, including Weber [91], Greenhill [42], Watson [90], Berwick [11] and more recently Gross and Zagier [44] (see also [36]).

We first prefer a basic approach. The simplest way to compute j is to compute $H_D(X)$ using floating point numbers (see [50, 31, 51]). In order to recognize that we have the right polynomial, we use an easy corollary of the work of Gross and Zagier, that can be stated as follows.

Proposition 7.1 *The norm of j in $\mathbf{Q}(j)$, which is the same as $H_D(0)$, is the cube of an integer in $\mathbf{Z}$.*

It is worth remarking at this point that we do not need to *prove* that our calculations with j are correct. If in fact they are, they will lead to elliptic curves which have the properties we need for proving primality, but the primality proof depends only on our computations on those curves. Thus we may find it convenient in the algorithm to work to limited floating point accuracy and confirm our j value without formal proof using observations like Proposition (7.1).

We want to evaluate $j(z)$ as fast as possible. For this, we compute in sequence $\eta(z)$, $\eta(2z)$, $f_2(z)$ and $j(z)$. The heart of the computation being the evaluation of $\eta(z)$, we now study the optimal choice of the parameters. Let us define

$$\mathcal{N}(q) = \sum_{n\geq 1}(-1)^n(q^{n(3n-1)/2} + q^{n(3n+1)/2}),$$

and

$$\mathcal{N}_N(q) = \sum_{n=1}^{N}(-1)^n(q^{n(3n-1)/2} + q^{n(3n+1)/2}),$$

where as usual $q = \exp(2i\pi z)$. We want to compute the error made when computing $\mathcal{N}_N(q)$ instead of $\mathcal{N}(q)$. We put $q = \rho\exp(i\theta) = \rho(cos\theta + i\sin\theta)$. The following proposition is easy to establish.

Proposition 7.2

$$|\mathcal{N}(q) - \mathcal{N}_N(q)| \leq 6\rho^{3N^2/2}. \tag{21}$$

We have to evaluate j for values of z of the form $z = (-b + i\sqrt{D})/2a$, where $(a,\ b,\ (b^2 + D)/4a)$ is a primitive reduced form of discriminant $-D$. When there is no ambiguity, we write $j(a, b)$ for $j((-b + i\sqrt{D})/2a)$. We put $q = \rho e^{i\theta}$, with $\rho = e^{-\pi\sqrt{D}/a}$ and $\theta = -\pi\, b/a$. Since this form is reduced: $a \leq \sqrt{D/3}$. We deduce that $\rho \leq e^{-\pi\sqrt{3}} < 4.34 \times 10^{-3}$.

Now, we remark that if (a, b, c) is an ambiguous form, then $j(a, b)$ is a real number. When (a, b, c) is nonambiguous, we get

$$j(a, -b) = \overline{j(a, b)}$$

(conjugation in $\mathbf{C}$) which halves the computation. After we have computed the h values of j, we build $H_D(X)$.

Using the q-expansion of j, it is not hard to see that $\log|j| \approx \pi\sqrt{D}/a$. The number of decimal digits of $j(q)$ is asymptotically $\pi\sqrt{D}/(a\log 10)$. We have to compute the coefficients of $H_D(X)$ to within 0.5. The precision required is thus

$$\mathrm{Prec}(D) = \binom{h}{\lfloor h/2 \rfloor} \frac{\pi\sqrt{D}}{\log 10} \sum \frac{1}{a} + \nu_0, \tag{22}$$

where the sum is taken over all primitive reduced forms of discriminant $-D$, and ν_0 a positive constant that takes care of the rounding error and the error made in our estimation of $\log|j|$ (typically $\nu_0 = 10$).

Suppose we want to compute $j(a,b)$. Then, using (7.2), we compute $\eta(kz)$ to the order

$$\sqrt{S \times \frac{a}{k}}, \tag{23}$$

where

$$S = \frac{2}{3} \frac{\log 6 + \mathrm{Prec}(D)\log 10}{\pi\sqrt{D}}. \tag{24}$$

We then form all products of the form $X - j$, grouping terms of the type $(X - j)$ and $(X - \overline{j})$ to get

$$(X-j)(X-\overline{j}) = X^2 - (j+\overline{j})X + j\overline{j},$$

which reduces computational errors.

We check the result with (7.1). If we find that $H_D(0)$ is the cube of an integer to within 0.5, we are confident that the computed polynomial is indeed the one we were looking for.

The coefficients of these polynomials become very large. For example, $D = 23$ already yields

$$H_{23}(X) = X^3 + 3491750X^2 - 5151296875X + 23375^3.$$

Thus it may be desirable to use subsidiary functions on subgroups of Γ.

7.2 Weber polynomials

Let $u(z)$ denote any modular function: Weber calls $u(\omega)$ a *class invariant* if $u(\omega)$ is in $\mathbf{K}(j(\omega)) = \mathbf{K}_{\mathrm{H}}$ (ω is the generator of $\mathcal{O}_K$). It turns out that there are a lot of alternative choices of class invariants other than j.

The following results can be found in [91, §125-144] or in [12, 82].

Theorem 7.1 ([91, §125, p. 459]) *Let z be a quadratic number defined by $Az^2+Bz+C = 0$. If*

$$3 \mid B, 3 \nmid A, 3 \nmid B^2 - 4AC, \tag{25}$$

we have

$$\mathbf{Q}(\gamma_2(z)) = \mathbf{Q}(j(z)).$$

Remark that the conditions are redundant, since A and B cannot be both divisible by 3 (else $D \equiv 0 \bmod 3$). Moreover, a careful look at the proof of this in [91, §125] shows that we can replace the above conditions by $A \equiv C \equiv 0 \bmod 3$ and $B \not\equiv 0 \bmod 3$. From this, we deduce a very simple algorithm to compute the correct value of the conjugates of $\gamma_2(\omega)$. We start from a form (a, b, c) associated with z_0 and we compute an equivalent form satisfying the above conditions, say (A, B, C) associated with z. We use the following procedure.

procedure GAMMA2(a, b, c)

1. if $a \not\equiv 0 \bmod 3$, then choose k such that $B \equiv b+2ak \equiv 0 \bmod 3$; take $k \equiv -b/(2a) \bmod 3$ and $(a, b+2ak, c+bk+ak^2)$ satisfies one of the above conditions;
2. if $a \equiv 0 \bmod 3$, but $b \not\equiv 0 \bmod 3$, then find k such that $C \equiv c + bk + ak^2 \equiv 0 \bmod 3$; a solution is given by $k \equiv -c/b \bmod 3$;
3. compute $\gamma_2(z) = \exp(2i\pi k/3)\gamma_2(z_0)$. This is valid because of (14) and (11).

From a practical point of view, the computation of $\gamma_2(z)$ is thus quite fast. It turns out that its coefficients are smaller than those of the original $H_D(X)$. For example, for $D = 23$, we find

$$H_{23}[\gamma_2](X) = X^3 + 155X^2 + 650X + 23375.$$

When $D \equiv 3 \bmod 6$, we have the following result.

Theorem 7.2 ([91, §134, p. 502]) *If $Az^2+Bz+C = 0$ with $2 \nmid A$, then $\mathbf{Q}(\sqrt{-D}\gamma_3(z)) = \mathbf{Q}(j(z))$.*

For instance, if $D = 15$, then

$$H_D[\sqrt{-D}\gamma_3](X) = X^2 - 1575X - 218295.$$

We can also use some power of the functions f, f_1 or f_2. We extract the following results from [51] (alternatively, see the references above). It is assumed from now on that $D \not\equiv 0 \bmod 3$. With each value of $D \bmod 32$, we have a canonical choice for u. Hence, we write $W_D(X)$ for the corresponding minimal polynomial.

D	u	$W_D(0)$	$\deg(W_D)$
$7 \bmod 8$	$f(\sqrt{-D})/\sqrt{2}$	-1	h
$3 \bmod 8$	$f(\sqrt{-D})$	$(-2)^h$	$3h$
$0 \bmod 4$			
$D/4 \equiv \pm 2 \bmod 8$	$f_1(\sqrt{-D})/\sqrt{2}$	± 1	h
$5 \bmod 8$	$f(\sqrt{-D})^4$	$\pm 2^h$	h
$1 \bmod 8$	$f(\sqrt{-D})^2/\sqrt{2}$	$(-1)^h$	h

Weber also gives conditions for more general z to satisfy the same properties. (One should also consult [88].) By extension, we will call *class invariant* any conjugate of $u(\omega)$ for a suitable u.

Theorem 7.3 *Suppose $Az^2 + 2Bz + C = 0$ with $4B^2 - 4AC = -4D$, A and C odd, $3 \mid B$, or equivalently, $A \equiv C \equiv 0 \bmod 3$ and $B \not\equiv 0 \bmod 3$. Then*

1. *in the case where* $D \equiv 1, 5, \pm 2 \bmod 8$*: if* $B \equiv 2((2/A) - 1) \bmod 8$*, then* $f(z)^2/\sqrt{2}$ *(resp.* $f(z)^4$*,* $f_1(z)/\sqrt{2}$*) is a class invariant;*

2. *in the case where* $D \equiv 3, 7 \bmod 8$*: if* $B \equiv 4((2/A) - 1) \bmod 16$*, then* $f(z)$ *(resp.* $f(z)/\sqrt{2}$*) is a class invariant.*

We only sketch the proof in the case $D \equiv 7 \bmod 8$. We combine the following results.

Proposition 7.3 ([91, §127, pp. 467]) *Let* z *be a root of* $Az^2 + 2Bz + C = 0$*, with* $-4D = 4(B^2 - AC)$*. Assume that* $3 \mid B$ *and that* A *and* C *are both odd and nondivisible by 3 (or* $A \equiv C \equiv 0 \bmod 3$ *and* $B \not\equiv 0$*). Then* $f^8(z)$ *is a class invariant.*

Proposition 7.4 ([91, §127, pp. 472]) *Assume the same conditions as above and also that*

$$C^2 + CB - 1 \equiv 0 \text{ or } 8 \bmod 16. \tag{26}$$

Then $\sqrt{2}f^3(z)$ *is a class invariant.*

Note that (26) implies that B is divisible by 8. Suppose now that all the preceding conditions are satisfied for $(A, 2B, C)$. Since $AC \equiv D \equiv -1 \bmod 8$, we see that $A \equiv -C \bmod 8$ and therefore $(2/A) = (2/C)$. Assume first that $(2/C) = 1$. Then $C \equiv \pm 1 \bmod 8$ and if $B \equiv 0 \bmod 16$ then

$$C^2 + CB - 1 \equiv 0 \bmod 16.$$

The case $(2/C) = -1$ is treated in the same way. We then write

$$\frac{f(z)}{\sqrt{2}} = \frac{1}{4}\frac{(\sqrt{2}f^3)^3}{f^8}.$$

The other cases are dealt with using results from the same section of Weber's book. □

We now briefly describe the algorithm needed to compute $H_{4D}[f/\sqrt{2}](X)$ for $D \equiv 7 \bmod 8$.

procedure WEBER7$(a, 2b, c)$

1. if a is even replace $(a, 2b, c)$ with $(c, -2b, a)$;
2. put $\xi(a) = 4((2/a) - 1) \bmod 16$;
3. if $a \not\equiv 0 \bmod 3$ choose k such that $B \equiv b + ak \equiv 0 \bmod 3$ and $B \equiv \xi(a) \bmod 16$; then $(a, 2b + 2ak, c + bk + ak^2)$ satisfies one of the conditions of Theorem 7.3;
4. if $a \equiv 0 \bmod 3$, but $b \not\equiv 0 \bmod 3$, then find k such that $C \equiv c + 2bk + ak^2 \equiv 0 \bmod 3$ and $b + ak \equiv \xi(a) \bmod 16$;
5. if z_0 (resp. z) is associated with $(a, 2b, c)$ (resp. $(A, 2B, C)$), then $z = z_0 - k$ and $f(z) = \zeta_{48}^k f(z_0)$ (resp. $f(z) = \zeta_{48}^k f_1(z_0)$) if k is even (resp. odd), using (15).

As an example, we find

$$H_{4\times 23}[f/\sqrt{2}](X) = X^3 - X - 1.$$

From this, it is easy to compute $j(\omega)$ for $\omega = (-1 + \sqrt{-23})/2$ via $f_2(\omega) = \sqrt{2}\zeta_{48}/f(\sqrt{-23})$ (see [91, §34, (19)]).

Other cases yield the same kind of algorithms.

7.2.1 Alternative class invariants

The second author is indebted to J.-F. Mestre who explained the following [59]. Let s be a prime positive integer and $X_0(s)$ be the modular curve [73]. It can be shown that (see, for example [37] or [58]), when $X_0(s)$ is of genus 0 (i.e., $s = 2, 3, 5, 7, 13$), it can be parametrized by:

$$x_s(z) = \left(\frac{\eta(z)}{\eta(sz)}\right)^{24/(s-1)}. \tag{27}$$

The modular invariant j is related to x_s via the following formulae:

$$\begin{aligned} j &= \frac{(x_2+16)^3}{x_2} = \frac{(x_3+27)(x_3+3)^3}{x_3} = \frac{(x_5^2+10x_5+5)^3}{x_5} \\ &= \frac{(x_7^2+13x_7+49)(x_7^2+5x_7+1)^3}{x_7} = \frac{(x_{13}^2+5x_{13}+13)(x_{13}^4+7x_{13}^3+20x_{13}^2+19x_{13}+1)^3}{x_{13}}. \end{aligned}$$

Theorem 7.4 *Let $-D$ be a fundamental discriminant and $s \in \{3, 5, 7, 13\}$ such that $(-D/s) = 1$. Let $(s) = \mathfrak{s}\mathfrak{s}'$ in $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$. Let i be a reduced ideal. Suppose we have found a basis (e_1, e_2) of i such that $\mathfrak{s} \times \mathfrak{i} = (e_1, se_2)$. Putting $\tau = -e_2/e_1$, the number $u_s(\mathfrak{i}) = x_s(\tau) + s^{12/(s-1)}/x_s(\tau)$ is a class invariant.*

For example, let $D = 23$ and $s = 13$. We have $\mathfrak{s} = (13, 8 + \omega)$. There are three reduced ideals. In the following table, we give the values of these ideals, the values of the reduced ideals $\mathfrak{s} \times \mathfrak{i}$ and the values of u_{13}.

$\mathfrak{i}$	$\mathfrak{s} \times \mathfrak{i}$	(e_1, e_2)	$x_{13}(-e_2/e_1)$	$u_{13}(\mathfrak{i})$
$(1, \omega)$	$(13, 8 + \omega)$	$(8 + \omega, 1)$	$-2.09988277 - 1.73159352i$	$-5.78492014 + 1.30714128i$
$(2, 1 + \omega)$	$(26, 21 + \omega)$	$(21 + \omega, 2)$	$-3.68503738 - 3.03873481i$	$-5.78492014 - 1.30714129i$
$(2, \omega)$	$(26, 8 + \omega)$	$(8 + \omega, 2)$	$-2.71507985 + 2.37241257i$	-5.43015970

We remark that, as soon as $\mathfrak{i} = (a, b + \omega)$ and $\mathfrak{s} \times \mathfrak{i} = (u, v + \omega)$, then (e_1, e_2) is precisely $(v + \omega, a)$, since $u = as$.

Finally, we get

$$\prod_{\mathfrak{i}}(X - u_{13}(\mathfrak{i})) = X^3 + 16.99999999X^2 + 97.99999994X + 190.9999999$$

and the minimal poynomial of u_{13} is $H_{23}[u_{13}](X) = X^3 + 17X^2 + 98X + 191$.

It is easily seen that in this case the minimal polynomial of x_{13} is

$$H_{23}[x_{13}](X) = (X^3 + \frac{17}{2}X^2 + \frac{59}{2}X + 37)^2 + 23(\frac{X^2}{2} + \frac{5X}{2} + 6)^2$$

so that x_{13} and hence j could be found by solving a cubic equation modulo p (recall that $\sqrt{-23} \bmod p$ will already have been found). The same situation arises in all cases given by Theorem 7.4.

With more effort one can also use values of s for which $\Gamma_0(s)$ does not have genus 0.

7.2.2 Remarks

A naive approach to the computation of W_D is to use polynomial factorization, or the LLL algorithm [51].

One of the phases of ECPP is to factor the polynomials H_D over $\mathbf{Z}/p\mathbf{Z}$. This can be expensive, since for a fixed large p the complexity of such computations is basically proportional to the square of the degree of the polynomial (see Section 8.6.1): This explains why we discard the case $D \equiv 3 \bmod 8$, since in this case, we might work on polynomials of degree $3h$.

We shall see in the following section how this computation can be simplified by factoring these equations over the genus field of $\mathbf{K}$. In order to simplify the notation, we will refer to $\mathcal{W}_D(X)$ as the defining polynomial of $\mathbf{K}_{\mathrm{H}}$ corresponding to whichever $H_D[]$ we can use. We call $\mathcal{W}_D$ a *Weber polynomial* associated with $-D$.

Let us end this subsection by summarizing the strategy for computing $\mathcal{W}_D$ given D.

procedure Weber(D)

1. **if** $D \not\equiv 0 \bmod 3$ and $D \not\equiv 3 \bmod 8$ **then**
   1. **if** $D \equiv 7 \bmod 8$ **then** $\mathcal{W}_D = H_{4D}[f/\sqrt{2}]$;
   2. **if** $D/4 \equiv \pm 2 \bmod 8$ **then** $\mathcal{W}_D = H_D[f_1/\sqrt{2}]$;
   3. **if** $D/4 \equiv 5 \bmod 8$ **then** $\mathcal{W}_D = H_D[f^4]$;
   4. **if** $D/4 \equiv 1 \bmod 8$ **then** $\mathcal{W}_D = H_D[f^2/\sqrt{2}]$;
2. **if** there exists s in $\{3, 5, 7, 13\}$ such that $(-D/s) = 1$ **then** $\mathcal{W}_D = H_D[x_s]$;
3. **if** $D \equiv 3 \bmod 6$ **then** $\mathcal{W}_D = H_D[\sqrt{-D}\gamma_3]$;
4. **otherwise** take $\mathcal{W}_D = H_D$.

7.3 Factoring the equations over the genus field

The aim of this section is to explain how it is possible to factor our $\mathcal{W}_D$'s over $\mathbf{K}_{\mathcal{G}}$. We will show that $\mathcal{W}_D$ has exactly g factors each of degree $e = h/g$ with coefficients in $\mathbf{K}_{\mathcal{G}}$. This reduces the time needed to compute a root of $\mathcal{W}_D \bmod p$ for large p, since we have to find a root of degree e instead of h.

$$\begin{array}{cl} \mathbf{K}_{\mathrm{H}} & \\ | & e = h/g \\ \mathbf{K}_{\mathcal{G}} & \\ | & g \\ \mathbf{K} & \end{array}$$

We first give some properties of composite quadratic fields, including the computation of an integral basis. Then, we set up an ordering on the genera of $\mathcal{H}(-D)$ through the action of the Galois group of $\mathbf{K}_{\mathcal{G}}/\mathbf{K}$. After proving the preceding results, we detail our algorithm and give some examples.

7.3.1 Some properties of composite quadratic fields

Let $u_1, \ldots, u_n$ be n squarefree multiplicatively independent elements of $\mathbf{Z}$. Suppose moreover that they are multiplicatively independent (i.e., $u_1^{a_1} \times \cdots \times u_n^{a_n} = 1$ is possible for some integers a_i if and only if the a_i's are all zero). We put $k_n = \mathbf{Q}(\sqrt{u_1}, \ldots, \sqrt{u_n})$ and $g = 2^n$. Following [23], we introduce the sequence $\{A_i\}_{0 \le i < g}$ defined by

$$A_0 = 1,$$
$$A_j = \begin{cases} u_{k+1} \text{ if } j = 2^k, \\ A_{2^{k-1}} A_i / \gcd(A_{2^{k-1}}, A_i)^2 \text{ if } j = 2^{k-1} + i \text{ and } 0 < i < 2^{k-1}. \end{cases}$$

We also define $\alpha_i = \sqrt{A_i}$. Then $\{1, \alpha_1, \ldots, \alpha_{g-1}\}$ is a basis for $k_n/\mathbf{Q}$.

Proposition 7.5 ([23]) *The integers of k_n are necessarily of the form*

$$x = \frac{1}{2^n} \sum_{i=0}^{g-1} P_i \alpha_i, \qquad (28)$$

where the P_i's are rational integers of the same parity, and all even if there is an i in $\{0, \ldots, g-1\}$ such that $A_i \not\equiv 1 \bmod 4$.

7.3.2 Computations in $\mathbf{K}_{\mathcal{G}}/\mathbf{K}$

As in Section 2, we write $-D = q_1^* \cdots q_t^*$, where $q^* = (-1)^{(q-1)/2} q$ if q is an odd prime and -4 or ± 8 otherwise. The q_i's are supposed to be ordered as follows: if $D \equiv 0 \bmod 4$, then $q_1 = 4$ or 8. Then the q's with $q^* = q$ are listed in increasing order and finally the q's with $q^* = -q$, also in increasing order. Then l is the number of positive q^*'s:

$$-D = q_1 \cdots q_l (-q_{l+1}) \cdots (-q_t).$$

The genus field $\mathbf{K}_{\mathcal{G}} = \mathbf{Q}(\sqrt{q_1^*}, \ldots, \sqrt{q_t^*})$ can be described as

$$\mathbf{K}_{\mathcal{G}} = \mathbf{K}(\sqrt{u_1}, \ldots, \sqrt{u_{t-1}}), \qquad (29)$$

where

$$u_i = \begin{cases} q_i & \text{for } 1 \le i \le l \\ q_t q_i & \text{for } l < i < t. \end{cases}$$

The Galois group of $\mathbf{K}_{\mathcal{G}}/\mathbf{K}$ is $\Sigma_G = < \varphi_1, \ldots, \varphi_{t-1} >$ where

$$\varphi_i(\sqrt{u_j}) = \begin{cases} -\sqrt{u_i} & \text{if } j = i, \\ \sqrt{u_j} & \text{if } j \ne i. \end{cases}$$

Hence, Σ_G is isomorphic to $(\mathbf{Z}/2\mathbf{Z})^{t-1}$, and we can represent an element ϕ of Σ_G by a $(t-1)-$ *tuple* of *signs* (i.e., elements of $\{\pm 1\}$). We decide to use the following ordering of the ϕ_i. If i is an integer between 0 and $2^{t-1} - 1$, we can write: $i = \sum_{s=0}^{t-1} \nu_{s+1} 2^s$ ($\nu_i \in \{0, 1\}$) and we take

$$\phi_i = \varphi_1^{\nu_1} \circ \cdots \circ \varphi_{t-1}^{\nu_{t-1}}.$$

We represent ϕ_i by $(e_1, \ldots, e_{t-1})$ where $e_s = 2\nu_s - 1$.

With this ordering, the i-th conjugate of an integer θ of $\mathbf{K}_{\mathcal{G}}$ is $\theta^{(i)} = \phi_i(\theta)$.

7.3.3 Ordering the genera

We show how to express the ϕ_i's in terms of $\mathcal{A}_G$, as described in Section 2. Let us write $\phi_i = (e_1, \ldots, e_{t-1})$ and $\mathcal{A}_G = (\epsilon_1, \ldots, \epsilon_t)$. What we have to solve is the system

$$\begin{cases} \epsilon_1 & = & e_1 \\ & \cdots & \\ \epsilon_l & = & e_l \\ \epsilon_{l+1}\epsilon_t & = & e_{l+1} \\ & \cdots & \\ \epsilon_{t-1}\epsilon_t & = & e_{t-1}. \end{cases} \tag{30}$$

We compute

$$\prod_{i=1}^{t-1} e_i = \left(\prod_{i=1}^{t-1} \epsilon_i\right) \epsilon_t^{t-l-1}.$$

With (1), we can simplify

$$\prod_{i=1}^{t-1} e_i = \prod_{i=1}^{t-1} \epsilon_i = \epsilon_t.$$

The solution of the system (30) is thus

$$\epsilon_i = \begin{cases} e_i & \text{for } 1 \leq i \leq l, \\ \prod_{i=1}^{t-1} e_i & \text{if } i = t, \\ \epsilon_t e_i & \text{for } l < i < t. \end{cases} \tag{31}$$

We take the ordering on the genera to be that induced by the preceding process. Let us give an example. Suppose that $-D = -308 = (-7) \times (-11) \times (-4)$. We take $u_1 = (-1) \times (-7)$ and $u_2 = (-1) \times (-11)$. The ϕ_i's and the associated genera are given below.

i	ϕ_i	G_i
0	$(+,+)$	$(+,+,+)$
1	$(-,+)$	$(+,-,-)$
2	$(+,-)$	$(-,+,-)$
3	$(-,-)$	$(-,-,+)$

(32)

It should be noted that the genus associated with ϕ_0 is always G_0, the principal genus. Moreover the ordering on the ϕ_i's depends only on g and not on D, whereas the correspondence with the genera depends on D and l. With each pair (t, l) satisfying $l \equiv t - 1 \bmod 2$, we associate the *generic ordering* defined by the above process. The example given above is the generic ordering $(3, 0)$.

We end this subsection by introducing

$$J_i = J(G_i) = \{j(C), C \in G_i\} = \{j_{i1}, \ldots, j_{ie}\}, 0 \leq i < g$$

and

$$\mathcal{W}_D{}^{(i)}(X) = P(J_i) = P(G_i) = \prod_{r=1}^{e} (X - j_{ir}).$$

We remark that $\mathcal{W}_D = \prod \mathcal{W}_D{}^{(i)}$ and that each $\mathcal{W}_D{}^{(i)}$ has only real coefficients, since two conjugate j's are in the same J.

The following fundamental theorem is now an easy consequence of Galois theory.

Theorem 7.5 *For all i, $\mathcal{W}_D{}^{(i)}(X)$ is in $\mathbf{K}_{\mathcal{G}}[X]$.*

We also have

Corollary 7.1 *For all i, $\phi_i(\mathcal{W}_D^{(0)}) = \mathcal{W}_D^{(i)}$.*

This motivates our choice of the ordering on the G's, since otherwise we would have to justify that the ϕ's permute the $\mathcal{W}_i$'s.

This result yields an algorithm for computing the expression of $\mathcal{W}_D^{(0)}$ over $\mathbf{K}_{\mathcal{G}}$. We describe this algorithm in the next section.

7.3.4 Description of the algorithm

The preceding results make it clear that the critical parameters are h and g; the algorithm does not depend explicitly on $-D$. Our purpose is now to explain how we can compute the coefficients of $\mathcal{W}_D^{(0)}$ and to exemplify the use of symbolic manipulation in the process.

We are looking for the coefficients of the polynomial $\mathcal{W}_D^{(0)}(X)$, which is a factor of $\mathcal{W}_D$ over $\mathbf{K}_{\mathcal{G}}$. We shall write $\mathcal{W}_i$ for $\mathcal{W}_D^{(i)}$ since there is no ambiguity. In fact, since the coefficients of $\mathcal{W}_0$ are real, we can work over k_{t-1} as defined above. The results are still valid by using the canonical isomorphism between the Galois groups of $\mathbf{K}_{\mathcal{G}}/\mathbf{K}$ and $k_{t-1}/\mathbf{Q}$.

We write

$$\mathcal{W}_0(X) = X^e + \sum_{r=0}^{e-1} \left(\sum_{s=0}^{g-1} a_{sr} \alpha_s \right) X^r, \tag{33}$$

where all the a_{sr} are in $(1/g)\mathbf{Z}$ and the α_s's as in Section 7.3.1. We will find these coefficients by means of the resolution of a linear system. Let $\alpha_s^{(i)} = \phi_i(\alpha_s)$.

For any polynomial $Q(X)$, let $[X^r]Q$ denote the coefficient of degree r of Q. Then

$$\sum_{s=0}^{g-1} a_{sr} \alpha_s = [X^r]\mathcal{W}_0. \tag{34}$$

Suppose now that r is fixed, $0 \le r \le e-1$. If we apply ϕ_i to (34), we find

$$\sum_{s=0}^{g-1} a_{sr} \alpha_s^{(i)} = [X^r]\mathcal{W}_i. \tag{35}$$

We do the same thing for $i = 0..g-1$ and we see that $(a_{sr})_{0 \le s < g}$ is the solution of the linear system

$$\left\{ \begin{array}{lcl} x_0 + x_1 \alpha_1^{(0)} + \cdots + x_{g-1} \alpha_{g-1}^{(0)} & = & Y_0 \\ x_0 + x_1 \alpha_1^{(1)} + \cdots + x_{g-1} \alpha_{g-1}^{(1)} & = & Y_1 \\ & \cdots & \\ x_0 + x_1 \alpha_1^{(g-1)} + \cdots + x_{g-1} \alpha_{g-1}^{(g-1)} & = & Y_{g-1}, \end{array} \right. \tag{36}$$

where we replace Y_i by $[X^r]\mathcal{W}_i$. We call the preceding system the *generic system of order* g, since it depends only on g. We see that we have just to solve this system once for each different value of g, computing all the a_{sr}'s by replacing the values of the α's by their corresponding floating point approximations.

From a practical point of view, we compute an approximation to ga_{sr}, take the nearest integer and then divide out by the same g. When we have computed our $\mathcal{W}_0$, we compute L, the lcm of the denominators of the coefficients and we store the coefficients of $L\mathcal{W}_0$.

As an example, let us treat the case of $-D = -308 = (-7)\times(-11)\times(-4)$. The generic system of order 4 is

$$\left\{\begin{array}{lcl} x_0 + x_1\alpha_1^{(0)} + x_2\alpha_2^{(0)} + x_3\alpha_3^{(0)} & = & Y_0 \\ x_0 + x_1\alpha_1^{(1)} + x_2\alpha_2^{(1)} + x_3\alpha_3^{(1)} & = & Y_1 \\ x_0 + x_1\alpha_1^{(2)} + x_2\alpha_2^{(2)} + x_3\alpha_3^{(2)} & = & Y_2 \\ x_0 + x_1\alpha_1^{(3)} + x_2\alpha_2^{(3)} + x_3\alpha_3^{(3)} & = & Y_3 \end{array}\right. \tag{37}$$

where

$$\left\{\begin{array}{lcl} \alpha_1^{(0)} & = & \sqrt{u_1} \\ \alpha_2^{(0)} & = & \sqrt{u_2} \\ \alpha_3^{(0)} & = & \sqrt{u_1u_2} \end{array}\right.$$

The generic ordering for $D = 308$ is $(3,0)$ and was given in section (7.3.3).

We want to get the expression of $H_{308}[\gamma_2]^{(0)}(X)$ over $\mathbf{K}_{\mathcal{G}}$. We have

$$\begin{array}{rcl} H_{308}[\gamma_2](X) & = & X^8 - 95835320X^7 - 923879753200X^6 + 121516780240000X^5 \\ & & -19528764670656000 0X^4 - 1627416205536000000X^3 \\ & & +35433687468608000000X^2 + 136128371025152000000 0X \\ & & -1293704102704640000000 0 \end{array}$$

Suppose that we have built the sets of roots of $H_{308}[\gamma_2]$ according to the genera. We have in this case

$$\begin{array}{ll} J_1 = J(+,+,+) & = \{88045635388240795530 5050.260304, 797.592915355\} \\ J_2 = J(+,-,-) & = \{5648.96421088 \pm 8460.8161800511\, i\} \\ J_3 = J(-,+,-) & = \{3456.226641, -938326357130.70446379\} \\ J_4 = J(-,-,+) & = \{-47921735.6519096497 \pm 83004169.578235232\, i\}. \end{array}$$

Finally, we obtain

$$\begin{array}{rl} H_{308}[\gamma_2]^{(0)}(X) = & X^2 + (-23958830 - 9057440\alpha_1 - 7223840\alpha_2 - 2730910\alpha_3)X \\ & +222228600 + 84022400\alpha_1 + 66972800\alpha_2 + 25321800\alpha_3. \end{array}$$

Numerous additional checks on the accuracy of our calculations are available using the supersingular equation. For example, $j = 0$ is the only supersingular value modulo 5, so that for $(-D/5) = -1$ all the roots of H_D must be zero modulo 5, as exemplified above.

8 Implementation details

8.1 Machines and Languages

The algorithm as described in detail in this paper has been implemented by the second author on a SUN 3/60 using `Le_Lisp` and the arithmetic described in [45].

The first author implemented the main ideas in the spring of 1986 using an IBM 3081 and his procedure LMA4064V. Most of the general purpose number theoretic routines were already available and 95% efficient using a combination of FORTRAN and ASSEMBLER. However he did not at that time have his (subsequently written) arbitrary precision complex floating point routines, and was thus confined in the computation of the H_D to IBM quadruple precision and some casual ingenuity. With a list of only 119 discriminants he was compelled to factorize the numbers of points excessively at great cost for large inputs. However the largest remaining prp343 in the Cunningham tables was done in 2.5 hours, and 250-digit numbers routinely in 3 to 8 minutes.

8.2 Strategies

8.2.1 Architecture of the program

The first basic approach is the *Factor and Prove Strategy* (FPS), following the direct application of the procedure ECPP. In other words, as soon as we have found a probably factored number, we immediately verify the conditions of the corresponding theorem. This idea works fine with small numbers (less than 10^{300}, say) since we are almost sure to find a good candidate among our list of D's. However, for large N's, our finite lists of D's can be too short and sometimes, we are forced to backtrack in our sequence of intermediate primes.

The preferred one is the *Factor All Strategy* (FAS) which first builds the sequence of intermediate primes and then proves all the theorems. This enables backtracking, as well as a more rational distributed algorithm (see [67]).

8.2.2 Philosophy

We constantly use some principles:

1. the tests with $N \pm 1$ are treated as a particular case of the elliptic curve test;
2. it is understood that, if a probable prime is later proved composite, then the program immediately returns to the preceding place in the DOWNRUN or exits if we were at the top. This of course involves the possibility of backtracking inside the program.

8.2.3 Computing $\mathcal{W}_D(X)$

In the proving part of our algorithm, we must compute $\mathcal{W}_D(X)$ in order to find a zero modulo p. There are two strategies. The first one is to precompute a list of $\mathcal{W}_D(X)$ for a subset of $\mathcal{D}$ and store them in a file. The other one is to compute $\mathcal{W}_D(X)$ on the fly, as required by the factoring part of the algorithm. It is clearly impossible to store all the $\mathcal{W}_D$ for all D's and thus we mix the two ideas. We have computed $\mathcal{W}_D$ for all D with $h(-D) \leq 20$ and stored them. This makes about 1.5 Mbytes (on a SUN 3/60). If necessary, other polynomials can be computed and introduced in the program. The actual computation of $\mathcal{W}_D$ is done by means of a MAPLE program. If one has the desired complex multiprecision arithmetic, one can of course merge the two programs.

We have computed all $\mathcal{W}_D(X)$ for all (known) D such that $h \leq 20$ and for $(h, g) \in \{(32,16), (24,8), (48,16), (32,8), (64,16)\}$. This yields 4500 potential numbers of points for each probable prime in our DOWNRUN. These are made up of 2 for ± 1, 6 for -3, 4 for -4 and 2 each for the remaining 2244 discriminants.

8.2.4 Ordering the data

We decide to use only the D less than 10^6 with $h(-D) \leq 50$. We remark that there are two parts in deciding whether p is a norm in $\mathbf{Q}(\sqrt{-D})$ or not. The first one is checking that $p \in G_0(-D)$: this is easy because we have only Jacobi symbols to compute. At this point, p is represented by F_D with probability $g(-D)/h(-D)$, but to be certain we must find a square root of $-D \bmod p$ and in effect reduce a quadratic form. So, we store our D's in increasing order with respect to $(h/g, h, D)$. The most interesting discriminants are those with $h = g$, which are called *idoneal numbers*: Assuming the Extended Riemann Hypothesis, there are 65 of them (see [35, 24]).

8.3 Logistics and Tactics

Many of the routines we use are explained and codified in [27]. We mention here one or two additional points.

8.3.1 Multiprecision

It is obvious that we need the fastest algorithms possible, especially a good routine for finding gcd's and multiplicative inversions. Also, the size of numbers we are currently tackling (more than 20 32-bit words) makes it worth using Karatsuba's algorithm. We refer to [52] for all this. We add below some remarks which may be well known, but not easily found in the literature.

We can use a special routine for squaring based on the following (trivial) observation. Let $m = \sum_{i=0}^{l-1} m_i B^i$ be an integer written in base B. Then:

$$m^2 = \sum_{i=0}^{l-1} m_i^2 B^{2i} + 2 \sum_{i=1}^{l-2} m_i B^i \sum_{j=i+1}^{l-1} m_j B^j.$$

This yields an algorithm for squaring that is asymptotically twice as fast as (ordinary standard) multiplication. In order to speed up things, it is necessary to program it directly in assembly in order to minimize overhead.

With this idea, we can replace multiplication by:

$$ab = \frac{(a+b)^2 - (a-b)^2}{4} = \frac{(a+b)^2 - a^2 - b^2}{2},$$

both formulae being useful, the latter one in the case where we must multiply many a's by the same b. Another application is given below.

8.3.2 Exponentiation over various rings

We use the exponentiation by blocks method as described in [27] for $\mathbf{Z}/p\mathbf{Z}$ $(= \mathrm{GF}(p))$, $\mathrm{GF}(p^2)$ ($N+1$ primality test) and for elliptic curves. The optimal value for the size of the block was determined empirically. The value of 2^6 seems to be the right one for almost all values of N.

When using Berlekamp's algorithm (as well as Girstmair's ideas, see below), we have to compute $P(z)^e \bmod (p, f(z))$ for a fixed monic $f(z)$. Write $f(z) = z^d + f_{d-1}z^{d-1} + \cdots + f_0$. We precompute:

$$F^{(i)} = z^i \bmod (p, f(z)) = F^{(i)}_{d-1} z^{d-1} + \cdots + F^{(i)}_0, 0 \leq i \leq 2d-2.$$

The basic operation we have to perform is the multiplication of $P(z) = p_{d-1}z^{d-1} + p_{d-2}z^{d-2} + \cdots + p_0$ by $Q(z) = q_{d-1}z^{d-1} + q_{d-2}z^{d-2} + \cdots + q_0$. We have:

$$P(z) \times Q(z) \bmod (p, f(z)) = \sum_{i=0}^{d-1}\sum_{j=0}^{d-1} p_i q_j z^{i+j} = \sum_i \sum_j p_i q_j F^{(i+j)}.$$

The evaluation of $p_i q_j$ is then done using the multiply-by-squaring method described above. If we precompute the p_i^2 and the q_j^2 (modulo p), we reduce the cost of computing $P \times Q$ to

$$\sum_i T_{sq}(p_i) + \sum_j T_{sq}(q_j) + \sum_{i,j} T_{sq}(p_i + q_j),$$

which is basically $(d^2 + 2d)T_{sq}$ compared to $d^2 T_\times$. The gain is thus:

$$\frac{d^2 + 2d}{d^2}\frac{T_{sq}}{T_\times} \approx \frac{1}{2}\left(1 + \frac{2}{d}\right).$$

The most obvious gain is when we have to compute the square of a polynomial. The cost of it is now $(d^2 + d)T_{sq}$.

8.4 Finding a good D

We have decided to consider the $N \pm 1$ test as a special case corresponding to a fictitious $D = \pm 1$. These tests have been well studied and many tricks are known to speed them up. In particular, we prefer the description of [27] since we can apply very easily the exponentiation by blocks method when working directly over $\mathrm{GF}(p^2)$ but not with Lucas sequences. We make here the remark that we use a trick of [21] to reduce the number of computations needed when one of our $N \pm 1$ has many factors (this is also valid for the elliptic case).

8.4.1 Looking for a splitting D

In the general case, we are looking for a fundamental discriminant $-D$ for which our probable prime N is a norm. The first thing we do is to check that $N \in G_0(-D)$. This is done by computing the Jacobi symbols $\chi_i(N) = (q_i^*/N)$ in our notation. If all these symbols are equal to $+1$, we proceed to the second phase, that is computing a representation of N by F_D.

Though the computation of Jacobi symbols is very cheap, one can arrange the D's in such a way that if N is a nonresidue modulo 3 (say), then we only look at those D which are not divisible by 3. In the same way, we can store the values of (N/q) for some small primes q (typically $q < 100$) so as not to recompute the same objects.

8.4.2 Solving $p = N_K(\pi)$

We want to get the representation of a prime p as a norm in $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$. Equivalently, we must solve:

$$4p = A^2 + DB^2 \tag{38}$$

with A and B in $\mathbf{Z}$. We can solve this problem using Shanks' algorithm [86] or lattice reduction [89]. These two algorithms are basically the same and solve the general case of representation of a prime number by a given quadratic form. In the case where we want to represent p by the principal form only, one can do slightly better, using the work of Cornacchia.

We first make some remarks. If $D \equiv 0 \bmod 4$, one puts $D = 4d$ and we have to solve $p = A^2 + dB^2$. If $D \equiv -1 \bmod 8$, then

$$4p = A^2 + (8d-1)B^2 \Rightarrow A^2 - B^2 \equiv 4 \bmod 8,$$

which is possible if $A^2 \equiv 0 \bmod 8$ and $B^2 \equiv 4 \bmod 8$ and in particular A and B even. So we actually solve $p = A'^2 + (8d-1)B'^2$. We can say nothing when $D \equiv 3 \bmod 8$.

Cornacchia's algorithm [32] finds the representation of $p = u^2 + dv^2$ whenever such one exists, with $(p, u, v) = 1$. A proof of its validity can be found in [70]. It runs like this:

procedure CORNACCHIA(u, v, d, p);

(* solution of $u^2 + dv^2 = p$ *)

1. let x_0 be a solution of $x^2 \equiv -d \bmod p$ that satisfies $p > x_0 > p/2$;
2. develop p/x_0 as a continued fraction:

$$\begin{aligned} p &= q_0 x_0 + x_1 \\ x_0 &= q_1 x_1 + x_2 \\ &\cdots \\ x_r &= q_{r+1} x_{r+1} + x_{r+2} \end{aligned}$$

   and stop when $x_r^2 < p \le x_{r-1}^2$;

3. put:

$$u = x_r \text{ and } v = \sqrt{\frac{p - x_r^2}{d}}.$$

4. if v is not an integer, p is not representable as $u^2 + dv^2$.

In the case $D \equiv 3 \bmod 8$, we can use the same algorithm using for x_0 a solution of $x^2 + x + \frac{D+1}{4} \bmod p$.

8.4.3 Extracting a square root modulo N

In using the above procedure, we have to compute square roots modulo N. We can benefit from some previous computations as follows.

If we use Shanks's algorithm, we need a z such that $z^{2^{k-1}} \equiv -1 \bmod N$, where $N = 1 + 2^k \times b$, b odd. We can obtain z as a byproduct of the pseudoprimality test for N as follows: first, find a such that $(a/N) = -1$ (if after 50 trials, we have not found one, maybe N is a square). Then, compute $z = a^b$: If $z^{2^{k-1}} \not\equiv -1 \bmod N$ then N is composite. Otherwise, N is a probable prime and we can use z in Shanks' algorithm. It should be noted that succeeding in computing a square root with this algorithm is almost a guarantee that we have a prime. In other words, if a composite number passed the pseudoprimality test, it is very unlikely to pass this step (see [94] for the combination of square roots with primality tests).

In many cases, a number of square roots can be found very cheaply at this first stage. If $N \equiv 1 \bmod q$ for small odd q, we can usually find the square root of $(-1/q)q$; while if $N \equiv 1 \bmod 8$ we find both $\sqrt{-1}$ and $\sqrt{-2}$.

Also, it is possible to accumulate the square roots that we have to compute until we find a suitable D. Hence, we can store $\sqrt{q^*} \bmod N$ for some small primes q. After that, we can use these values in the computation of $j \bmod N$ when $\mathcal{W}_D$ splits over an intermediate quadratic field of discriminant q^*. We also order our D's in such a way that we compute less square roots modulo p. For instance, we try $D = 3$, $D = 4$ then $D = 15$ by computing only $r_5 = \sqrt{5} \bmod p$, $D = 20$ by combining $D = 4$ and r_5, etc.

8.5 Factoring the number of points

8.5.1 Finding all factors of an integer m which are less than B

It is well known that the general problem of getting *all* factors of an arbitrarily large number is very difficult (see [55]). However the problem of getting small factors of a number m is a little better understood.

What we want is an algorithm that can find small factors of a number in a reasonable amount of time. Apart from trial division that is routinely used to find all factors less than 10^6, the two best candidates are Pollard's ρ method [62] and the ECM method of Lenstra [57]. Following [17], it seems that the first one is worth using for finding factors less than 10^8 and the second for factors from 10^{10} to 10^{15} using various speedups [62, 7].

However, the very best value among these probabilistic factoring methods is given by Pollard's $p - 1$, even though this can only be used once.

It should be noted that we do not store the intermediate factors found, only their product. This is motivated by the fact that we do not need to have the exact factorization of m (unless m is small). It can happen that a 20-digit factor of a 1000-digit number is not prime, but we are only interested in having a 980-digit probable prime.

We detail the choices we made in the following section.

8.5.2 Sieving with small primes

We begin by looking for small prime factors of m. Let $p_1, \ldots, p_k$ be all the prime numbers less than a given B. We suppose they are stored in a file. We extend a method already used in [18, Section 7, Rem. 1] and [27]. We first compute the quantities

$$r_i = (N+1) \bmod p_i, \text{for } i = 1..k.$$

Divisibility of $N \pm 1$ is then tested as follows. If $r_i = 0$, then $N+1$ is divisible by p_i and if $r_i = 2$ then p_i divides $N-1$. In the case of testing the divisibility of $m_\pm = N + 1 \pm t$, with $|t| \leq 2\sqrt{N}$, we first compute $r = t \bmod p_i$ and check whether $r \equiv \pm r_i \bmod p_i$. We replaced $2k$ divisions of numbers of size L with k divisions of integers of size $L/2$.

However there is yet a further factor of 2 to be gained. With $4p = A^2 + DB^2$ we have $4m = (A \pm 2)^2 + DB^2$. For any sieving prime $p_i > 2$ with $(-D/p_i) = -1$, we have $p_i \mid m \Rightarrow p_i \mid B^2$ and $p_i \mid (A \pm 2)^2$ respectively. Thus we first form $\gcd(A+2, B)$ and $\gcd(A-2, B)$ an remove the common factors from m; subsequently only primes p_i with $(-D/p_i) = 1$ are used in the sieve (and recognized from a lookup table modulo D or $4D$).

8.5.3 Pollard's ρ

From [17], it is reasonable to find all factors less than 10^8 with this method. Using the ideas of [62], we decide to make 10^5 iterations of this method. We accumulate the iterates of the function and do only two gcd's.

8.5.4 ECM

We use the algorithm as described in [17] with the parametrizations of [62, 7] for having curves with some prescribed small divisors. One of the major problems is the storage that is prohibitive when dealing with 1000-digit numbers. This explains why the second stage is not performed on numbers of size greater than 10^{700}.

8.5.5 Pollard's $p-1$

We note that this is reasonable when testing the Cunningham numbers which have often the property of being congruent to ± 1 modulo some large known prime integer. So we can spend a little time to see if we can get a factor (possibly large) of m this way.

8.5.6 Further improvements

The best discriminants are those for which $h = 1$, because j is easy to compute and E is easy to find (see next section). Hence, we decide to use more factoring power on them. Say we multiply all factorization parameters by 1.2, maybe with all possible methods as well. However, in order for them to appear in the DOWNRUN, we must find a suitable number of points.

Suppose we test N_i and we get a candidate N' for N_{i+1}. First of all, we can impose an upper bound on N'. More precisely, we want to go down in our sequence of primes as fast as possible. Therefore, we decide to reject all possible N' such that $N_i/N' < 10^{minx}$, say. The exact value of $minx$ is best found by experiments. This results in many different strategies. which we do not discuss here.

We also try to have a next candidate that is as *promising* as possible. If we find a N' which is congruent to 1 modulo 3 or 4, we take it. On the contrary, when $N' \equiv -1 \bmod 24$, we prefer to try another one; the two strategies can be combined.

8.6 Finding $j(E) \bmod p$ and a point on $E(\mathbf{Z}/p\mathbf{Z})$

The process is the following: first compute $j(E)$ a root of $H_D(X) \equiv 0 \bmod p$, then find the equation of E and a point on E. In fact, we compute a root of $\mathcal{W}_D(X) \equiv 0 \bmod p$ and we compute j.

8.6.1 Solving $\mathcal{W}_D(X) \equiv 0 \bmod p$

The obvious approach to solving $\mathcal{W}_D(X) \equiv 0 \bmod p$ is to use Berlekamp's algorithm [10, 52]. The complexity of this algorithm is roughly:

$$O((d^2(\log p) + d^3)(\log d)(\log p)^2),$$

if we use standard algorithms (with $d = h(-D)/g(-D)$). For small d, it is possible to mimic the standard resolution over $\mathbf{C}$ (see [96, 65]).

Alternatively, one can use the fact that the Galois group of $\mathcal{W}_D(X)$ is very often a dihedral group. Then using Girstmair's ideas [39], it is possible to solve the equation $\mathcal{W}_D(X) \equiv 0$ by radicals and use the resulting expressions modulo $(p, f(z))$ where $f(z)$ is any factor of the h-th cyclotomic polynomial modulo p. For example, take $p = 439 = 1^2 + 1\times 6 + 6^2\times(47+1)/4$, whose order modulo 5 is 2. A root of $W_{47}(X) = X^5 - X^3 - 2X^2 - 2X - 1$ over $\mathbf{C}$ is given by:

$$5x_5 = \sum_{k=1}^{4} \frac{z^{(k)}}{z^{(1)}} y^k,$$

with:

i	$2z^{(i)}$
1	$(80\sqrt{-47} - 650)\zeta^4 + (15\sqrt{-47} - 975)\zeta^3 + (-975 - 15\sqrt{-47})\zeta^2 + (-650 - 80\sqrt{-47})\zeta$
2	$(15\sqrt{-47} - 105)\zeta^4 + (5\sqrt{-47} - 185)\zeta^3 + (-185 - 5\sqrt{-47})\zeta^2 + (-105 - 15\sqrt{-47})\zeta$
3	$(\sqrt{-47} - 35)\zeta^4 + (-15 - 3\sqrt{-47})\zeta^3 + (3\sqrt{-47} - 15)\zeta^2 + (-35 - \sqrt{-47})\zeta$
4	$-2\zeta^4 - 8\zeta^3 - 8\zeta^2 - 2\zeta,$

where ζ is a primitive 5-th root of unity and $y^5 = z_1$. We work over $(\mathbf{Z}/439\mathbf{Z})[z]/(z^2 + 70z + 1)$: The corresponding value of ζ is simply z. A squareroot of $-47 \bmod 439$ is 294. We extract a fifth root of y using an extension of the algorithm of [1] as described in [47] (see also [46]). We find

$$y^5 = 269z + 64 = (383z + 244)^5 \bmod (439, z^2 + 70z + 1).$$

and $x = 15$ is a root of $W_{47} \bmod 439$. The ideas are detailed at some length in [64].

In general, the Abelian Galois group $\mathbf{K}_{\mathrm{H}}/\mathbf{K}_{\mathcal{G}}$ is cyclic; when this is so and the order is composite, the usual resolution into a sequence of equations of prime degree (each with coefficients in the field defined by the previous equation) is highly effective in solving for the (known) root modulo p. For example, with $D = -199$ we find

$$H_{4\times 199}[f/\sqrt{2}](X) = X^9 - 5X^8 + 3X^7 - 3X^6 - 3X^3 - X - 1,$$

whose roots are solved via

$$Y^3 - 4Y^2 + Y - 1,$$

$$X^3 - (Y^2 - 3Y + 1)X^2 - X - Y.$$

Further examples can be found in [66].

8.6.2 Finding the right equation for E

We have to find an equation of the curve $E(\mathbf{Z}/p\mathbf{Z})$ whose invariant is j (computed above) and whose Frobenius is π with p a norm in $\mathbf{Q}(\sqrt{-D})$: $p = \pi\pi'$. In the general case $D > 4$, the equation of E is of the form

$$y^2 = x^3 + 3kc^2x + 2kc^3, \tag{39}$$

where $k = j/(1728 - j)$ with c any element of $\mathbf{Z}/p\mathbf{Z}$.

We can restate the problem as follows. By Deuring's work, we have

$$\Sigma_E(p) = \sum_{x=0}^{p-1} \left(\frac{x^3 + 3kc^2x + 2kc^3}{p}\right) = -\mathrm{Tr}_K(\pi_c), \tag{40}$$

where π_c is the actual Frobenius of E as parametrized by c. As a matter of fact, it is always possible to write

$$\mathrm{Tr}_K(\pi_c) = \varepsilon(D, \pi) \left(\frac{c}{p}\right) \mathrm{Tr}_K(\pi) \tag{41}$$

where $\varepsilon(D, \pi)$ ($\varepsilon \in \{\pm 1\}$) is a function of π and D. The equation we are looking for is thus characterized by c such that

$$\left(\frac{c}{p}\right) = \varepsilon(D, \pi).$$

The aim of this section is to explain how we find the value of $\varepsilon(D, \pi)$ in some cases. Before that, we treat two special cases.

The case $h(-D) = 1$. The first two cases are $D = 3, 4$. They are treated at length in [48, Chapter 18, §3–4] and involve quartic and sextic symbols. For the sake of self-containedness, we just give the algorithms used in each case. The validity of these come from [63]. Let us first consider $D = 3$.

procedure FINDE3(p)

(* $p = \pi\pi'$ with $\pi = A + B\rho$, A, B in $\mathbf{Z}$ and $\rho = (1 + \sqrt{-3})/2$ *)
(* the equation of E is $y^2 = x^3 + b$ *)

1. let $\zeta = r + s\rho$ with r, s in $\{\pm 1, 0\}$, $r \equiv 2(A - B) \bmod 3$ and $s \equiv B \bmod 3$; then $\zeta^6 = 1$ and $\zeta\pi \equiv 2 \bmod 3$;
2. determine $\mathcal{B}$ in $\mathbf{Z}/p\mathbf{Z}$ such that $\mathcal{B} \equiv -\zeta \bmod \pi$:
    1. solve $(A - B)v + Bu = s$ in rational integers (u, v);
    2. put $\mathcal{B} = -r + Au - Bv$;
3. any b such that $(4b)^{(p-1)/6} \equiv \mathcal{B} \bmod p$ yields a curve $E : y^2 = x^3 + b$ such that $\#E = N_K(\pi - 1)$.

For $D = 4$, we have

procedure FINDE4(p, π)

(* $p = \pi\pi'$ with $\pi = A + Bi$, A, B in $\mathbf{Z}$ and $i^2 = -1$ *)
(* exactly one of A or B is even *)
(* the equation of E is $y^2 = x^3 + ax$ *)

1. let r and s be two integers in $\{\pm 1, 0\}$ such that $rs = 0$ and $(r, s) = (0, (B - A) \bmod 4)$ if A is even and $(r, s) = ((A - B) \bmod 4, 0)$ if B is even; then $\zeta^4 = 1$ and $\pi \equiv \zeta \bmod (2 + 2i)$;
2. determine $\mathcal{A}$ in $\mathbf{Z}/p\mathbf{Z}$ such that $\mathcal{A} \equiv \zeta^{-1} \bmod \pi$:
   1. solve $Av + Bu = s$ in rational integers (u, v);
   2. then $\mathcal{A} = r + Au - Bv$;
3. any a with $(-a)^{(p-1)/4} \equiv \mathcal{A} \bmod p$ gives a curve $E : y^2 = x^3 + ax$ with $\#E = N_K(\pi - 1)$.

When $D = 8$, we use a result from [78]. Write

$$E_\vartheta : y^2 = x(x^2 - 4\vartheta x + 2\vartheta^2) \tag{42}$$

with ϑ in $\mathbf{Z}$.

Theorem 8.1 *Let $p = \pi\pi' = A^2 + 2B^2 = 8\kappa + l$ ($l \in \{1, 3\}$) with A and B in $\mathbf{Z}$, A odd. Then*

$$\Sigma_{E_\vartheta}(p) = -\left(\frac{\vartheta}{p}\right)(-1)^\kappa \left(\frac{-1}{A}\right) \mathrm{Tr}_K(\pi).$$

We can restate this in the form of (39). We have:

$$j(\sqrt{-2}) = 20^3 \text{ and } k = -\frac{5^3}{2.7^2}.$$

Letting $c = 14\vartheta/15$, we find that E_ϑ is isomorphic to (39). Hence, we deduce that

$$\Sigma_E(p) = \Sigma_{E_\vartheta}(p) = -\left(\frac{15 \times 14 \times c}{p}\right)(-1)^\kappa \left(\frac{-1}{A}\right) \mathrm{Tr}_K(\pi). \tag{43}$$

We conclude that

$$\varepsilon(8, \pi) = \left(\frac{2.3.5.7}{p}\right)\left(\frac{-1}{A}\right)(-1)^\kappa.$$

For the remaining cases where $h(-D) = 1$ and D is odd, we refer to some work of Rajwade. In [79], he has designed a method to solve the problem in the case where $D = 7$ and later extended it to the cases $D \in \{11, 19\}$ (see also the bibliography in [76]). He uses the action of the Frobenius of the curve on the $\sqrt{-D}$-division points to deduce the actual value of Σ_E. For D in $\{43, 67, 163\}$, he quotes some unpublished results from Stark (see [74]).

All these results can be summarized by the following theorem:

Theorem 8.2 *Suppose that D is odd and $h(-D) = 1$. Let $j = j((-1+\sqrt{-D})/2)$ be the invariant of the curve having complex multiplication by the maximal order of the quadratic field $\mathbf{K} = \mathbf{Q}(\sqrt{-D})$, defined over $\mathbf{Q}$. Let u and v be defined by*

$$u^3 = j, -Dv_0^2 = j - 1728, v = \left(\frac{2}{D}\right) v_0, v_0 > 0.$$

Let p be a norm for $-D$: $p = \pi\pi'$. (In this case, this is merely the same as $(-D/p) = +1$.) Then

$$\varepsilon(D, \pi) = \left(\frac{3uv}{p}\right)\left(\frac{2\mathrm{Tr}_{\mathrm{K}}(\pi)}{D}\right). \square \tag{44}$$

We now give the numerical values of u and v_0 for all D.

D	u	v_0	$(2/D)$
7	-3.5	3^3	$+$
11	-2^5	$2^3.7$	$-$
19	$-2^5.3$	$2^3.3^3$	$-$
43	$-2^6.3.5$	$2^3.3^4.7$	$-$
67	$-2^5.3.5.11$	$2^3.3^3.7.31$	$-$
163	$-2^6.3.5.23.29$	$2^3.3^3.7.11.19.127$	$-$

Examples. Let $p = 17401 = 101^2+2\times60^2$. We choose $\pi = 101+60\sqrt{-2}$ ($m = p+1-2\times101 = 17200$). We find that $\kappa = 2175$ and:

$$\left(\frac{c}{p}\right) = \left(\frac{2.3.5.7}{p}\right)(-1)^\kappa\left(\frac{-1}{101}\right) = (+1)(+1)(+1)(-1)(-1)(+1) = +1. \tag{45}$$

Therefore, the curve $E : y^2 \equiv x^3 + 15444x + 10296 \bmod 17401$ has 17200 points.

Let $p = 107$ and $D = 7$. We find:

$$107 = 10^2 + 7\times1^2$$

so that $(107) = (\pi)(\pi')$ with $\pi = 10+\sqrt{-7}$. We choose $m = N_K(\pi-1) = 88$. We compute:

$$\left(\frac{c}{107}\right) = \left(\frac{-3(3\times5)(3^3)}{107}\right)\left(\frac{20}{7}\right) = (-1)\left(\frac{15}{107}\right)\left(\frac{20}{7}\right) = (-1)(-1)(-1) = -1,$$

and the curve $y^2 \equiv x^3 + 60x + 80 \bmod 107$ has 88 points.

The case $h = 2$: Let $D \equiv 3 \bmod 4$ such that $h(-D) = 2$. Write $-D = (q_1)(-q_2)$. Then j lies in $\mathbf{Q}(\sqrt{q_1})$. Let $j = a + b\sqrt{q_1}$ and $v = \sqrt{(j-1728)/(-D)}$. Then v is also in $\mathbf{Q}(\sqrt{q_1})$ (see [44]). We conjecture the following:

Conjecture 8.1 *Suppose that $-D = q_1(-q_2)$ as above and $v = A + B\sqrt{q_1}$ in such a way that* $\mathrm{sign}(B) = -\mathrm{sign}(b)$. *Then:*

$$\varepsilon(D, \pi) = \left(\frac{3jv\left(\frac{2}{D}\right)}{p}\right) \times \left(\frac{\mathrm{Tr}_{\mathrm{K}}(\pi)}{D}\right).$$

Example. Let $-D = -403 = (13)(-31)$. We have

$$j = -122640569461466569598976000 + 34014373972724674193817600\sqrt{13},$$

$$v = 1233529551576 - \frac{4447554048000}{13}\sqrt{13}.$$

Remark. The above results are also related to the concept of $\mathbf{Q}$-curve as introduced by Gross in [43]. Some of the methods used by him would yield the same results, but using deep methods from algebraic geometry.

8.6.3 Finding P on E

Let (a, b) be two elements of $\mathbf{Z}/p\mathbf{Z}$. If x_0 is any element of $\mathbf{Z}/p\mathbf{Z}$, put

$$\lambda = x_0^3 + ax_0 + b.$$

Then $P = (\lambda x_0, \lambda^2)$ is on the curve:

$$Y^2 = X^3 + a\lambda^2 X + b\lambda^3.$$

We suppose that $E : y^2 = x^3 + 3kc^2x + 2kc^3$. If we know something about (c/p) (typically when $h(-D) = 1$), then we choose x_0 such that (λ/p) agrees with (c/p). Then, we have simultaneously E, and P on E. Otherwise, we choose x_0 at random and test whether mP is on E. If it is not, then we try the twist of E. In the general case, the time needed to find the right curve is thus 1.5 times the time needed fot the $h = 1$ case. In all cases, we have no extraction of square roots modulo p.

9 Numerical results

9.1 Timings on random input numbers

We follow the protocol given in [27]. That is, we obtain certain statistics on the behavior of our program for 20 numbers of w 32-bit words. The program is the DecStation 5000 version with the FAS strategy. We list the time for the first phase (building the sequence) on the first line, the second one (proving) on the second line and the total time on the third. Times are in seconds. The set $\mathcal{D}$ consists of all D's with $h(-D) \leq 20$.

For larger numbers, we use a distributed process with all D with h less than 51 and some others (see Section 8.2.3). The order of magnitude of the time needed is given in equivalent time for a DecStation 5000.

9.2 Some large primes

Both authors used their implementations to give primality proofs for the probable primes of the Cunningham Tables [19]. The first author did some with 212 to 343 digits (namely the cofactor of $2, 1171+$) [19, Update# 5] and the second author completed the long standing list (about 50 numbers with more than 200 digits). The second author verified the primality of the cofactor of F_{11} (564 digits) [19, Update 2.2], and also $(2^{3539} + 1)/3$ (1065 digits) with a distributed version of ECPP [67].

w	min	max	mean	st. dev.
2	0.1	0.4	0.2	0.1
	0.0	0.3	0.1	0.1
	0.1	0.6	0.3	0.1
4	0.4	2.6	1.2	0.5
	0.2	2.7	1.5	0.7
	0.7	5.3	2.7	1.1
6	1.7	55.2	6.1	11.4
	3.5	8.9	5.8	1.5
	5.4	61.6	12.0	11.7
8	2.5	13.7	7.7	3.0
	7.3	24.4	14.9	4.8
	10.2	34.8	22.6	7.5
10	7.6	31.3	19.9	6.2
	20.6	54.6	35.3	9.2
	28.2	85.0	55.2	14.2

w	min	max	mean	st. dev.
12	14.9	238.9	45.6	48.1
	31.3	88.3	61.1	14.2
	50.4	301.9	106.6	49.4
14	41.8	181.1	75.5	36.8
	75.8	150.7	107.3	21.4
	122.9	259.7	182.8	49.3
16	120.2	763.8	261.1	138.2
	129.5	226.8	174.9	26.6
	249.7	990.7	436.0	154.1
18	118.8	1699.1	657.3	420.2
	123.4	339.5	214.7	53.6
	242.1	1973.8	872.1	447.5
20	344.0	27561.2	2711.3	5805.6
	174.8	360.9	254.2	52.2
	557.5	27776.8	2965.5	5801.1

Table 4: Time for testing a number of w words for primality.

d	DOWNRUN	proving
400	1 day	0.1 days
600	6 days	0.5 days
800	18 days	5 days

Table 5: Time for testing a d-digit number for primality.

Aside from the Cunningham project, the second author found all primes of the form $N_2(n,r) = r\,\underbrace{1\cdots1}_{n\ times}$ (introduced in [93]) for all r greater than 1 and all n between 100 and 1000. We indicate below these values (note that all numbers with $n \leq 99$ were found by Williams).

r	n	r	n
2	$12, 18, 23, 57, 128, 543, 584, 833$	6	$5, 7, 25, 31, 112, 199$
3	$5, 10, 11, 13, 34, 47, 52, 77, 88, 554, 580$	7	$7, 55$
4	$13, 25, 72, 108, 375, 393, 589, 973$	8	$26, 110, 141, 474$
5	$5, 12, 15, 84, 144, 150$	9	$5, 20, 41, 47, 92, 161, 401, 455$

In addition, some large probable primes were successfully tested. Among these were S_{1493} (572 digits, three weeks on a SUN 3/60) and S_{1901} (728 digits, one month) thus solving the problem mentioned at the end of [72].

Apart from these numbers with quite a lot of arithmetical properties, the second author is currently looking for large primes coming from the factorization of the numbers constructed from well-known constants such as π, e and γ. To this date, the three largest proven primes found are the cofactor of $\gamma_{1137} = \lfloor 10^{1137}\gamma \rfloor = 2 \times 47 \times 4231 \times 7789 \times p_{1128}$ (with the distributed implementation in equivalently about 1.75 years of CPU of a SUN 3/60); the cofactor of $e_{1230} = \lfloor 10^{1230}e \rfloor = 36037 \times P_{1226}$ (1.83 years of CPU); the partition number $p(1840926)$ with 1505 digits [69]. Together with the 1008 digit cofactor of M_{3359}, these are five *Titanic* primes successfully tested by ECPP.

10 What proof do we get?

We now turn our attention to the following problem: How can we be sure that our program did not make any error during one month of CPU time? We cannot be certain that there was no bit-loss during this period. However, when the program finishes, we have built a sequence of intermediate primes and found an elliptic curve and its number of points and a point on it satisfying the requirements of a theorem. This we call a *certificate of primality*. We thus generalize previous work of Pratt [77] and Pomerance [75]. We arrange such a certificate in blocks of integers. Each block has the following structure :

$$\begin{array}{c} N_i \\ \textit{type} \\ \boxed{\begin{array}{c} \mathrm{P} \\ \mathrm{R} \\ \mathrm{O} \\ \mathrm{O} \\ \mathrm{F} \end{array}} \\ 0 \end{array}$$

where N_i is the number to be tested, *type* giving the type of theorem used to show the primality of N_i: it is -1 (resp. $+1$) if the $N-1$ test (resp. $N+1$ test) was used, and the absolute value of the fundamental discriminant used in the cases of elliptic curves. The primality proof of N_0 ends with a 0. To each of the types corresponds a list of numbers used to complete the proof of N_i being prime, whenever the following block is valid. We now describe the four possible lists:

type -1	type $+1$	type D
		$m = \#E$
$\left.\begin{array}{c} p_0 \\ \cdots \\ p_k \end{array}\right\}$ factors of $N-1$	$\left.\begin{array}{c} q_0 \\ \cdots \\ q_l \end{array}\right\}$ factors of $N+1$	$\left.\begin{array}{c} r_0 \\ \cdots \\ r_u \end{array}\right\}$ factors of m
0	0	0
		a, b: coefficients of E
		x, y: coordinates of P on E
$\left.\begin{array}{c} b_0 \\ \cdots \\ b_l \end{array}\right\}$ (Cf. Theorem 1 in [97])	$\left.\begin{array}{c} P_0, Q_0 \\ \cdots \\ P_k, Q_k \end{array}\right\}$ id.	$\left.\begin{array}{c} f_1 \\ \cdots \\ f_u \end{array}\right\}$ factors of the order of P

Figure 2: Format of the primality proof

In this way, an independent verifier can check the results. A cross verification of certificates was carried out between the second author and Kaltofen and Valente (personal communications via email, October 1989). After some adjustments of format, they both agreed on the certification of a 222-digit prime, namely $2, 1958M$ (in the notations of [19]). They also checked the 1226-digit record.

11 Conclusion

We have described a primality proving algorithm using the theory of elliptic curves with complex multiplication over finite fields. This algorithm is supposed to have poynomial complexity and performs well in practice, since it is powerful enough to prove the primality of numbers from 100 to 1500 digits. It is now possible to test arbitrary integers up to 400 digits in a few days on a single SUN 3/60 workstation. Numbers with less than 800 digits can be done in about one week of real time using a distributed process [67] on about 10 workstations.

There remains much uncertainty as to the best strategy for applying the method to large probable prime inputs. We first eliminate some minor points which are not germane to the general problem.

The situation for 100 digits and less is quite atypical. There the downrun is dominated by $D = \pm 1$, $D = -3$, and $D = -4$; in particular once $D = -3$ is reached one can usually stay with it to the end. Square roots are much cheaper relative to sieving than they are for large inputs, and optimisation is desirable at all stages of the program.

Also (for all sizes of input) the reduction of quadratic forms takes negligible time, and the polynomials H_D can be computed very quickly at the time when they are needed.

Thus the general operations which should be programmed optimally, and whose timings on a particular machine are relevant to the strategy are:

1. Sieving and subsequent factorization of the numbers of points,

2. Exponentiation modulo p (and equivalent square roots, pseudoprime tests),

3. Exponentiation on an elliptic curve modulo p,

4. Solution of polynomial congruences modulo p.

Usually 4. can be reduced to finding a small number of square roots, but an occasional discriminant with large class number which is unfavorable for p can be very expensive. As to sieving, it is worth pointing out that it is much more effective here relative to other factorization methods than usual. Once -1 and $+1$ have been done (as discriminants) there is available a list of $(N+1) \bmod q$. For any particular discriminant $-D$, one only needs to use half the sieving primes q, dividing numbers of size the square root of N, and applicable to two possible numbers of points, a total improvement factor of 8 (16 or 24 for $D = 4$ or 3). On the other hand, $(p-1)$-factorization and ECM are no better than usual (except that one can in a few cases use an elliptic curve with complex multiplication to good effect in ECM.)

A further remark is that the timings of these operations depend not only on the machine, but on the trouble which the programmer has been prepared to take. For example, some critics purport to "prove" that the Weierstrass normal form is not the best one to use in 3. above, but they rely on an unproved (and possibly unconscious) assumption that finding inverses is slow. The first author is fortunate in having the use of a very fast g.c.d. routine written by N. W. Rickert [81], which alters his choice of algorithm in this and other cases. We will now assume that all these operations have been optimised as far as they are going to be, and that the timings for various typical numbers of decimal digits are known.

We feel that the optimal strategy will probably have more backtracking facility that either of us uses at the moment. At a given point in the downrun, one has basically to choose four parameters: the size of the sieve, the additional factoring to be used, the minimum acceptable downrun, and how many discriminants to try before modifying the parameters. There is no doubt that sieving represents by far the best value for time spent, so that for inputs of 500 decimal digits or more one should probably think in terms of a sieve with several passes and recomputed lists of primes. We hope to implement some of these ideas and report further in due course.

Acknowledgments. Many people are to be thanked for the help they gave to various stages of this work. J. Cougnard read a preliminary version of [65] that is incorporated in this article. J. McKay gave some advice concerning Galois theory and a pointer to [39]. V. Miller and J.-F. Mestre communicated their work on $X_0(s)$. Finally, H. Cohen carefully read a preliminary version of this paper. We also acknowledge with gratitude the numerous detailed suggestions of the referee.

References

[1] L. Adleman, K. Manders, and G. L. Miller. On taking roots in finite fields. In *Proc. 18th Annual IEEE Symp. Foundations of Computer Science* (1977), pp. 175–178. Oct. 31–Nov. 2, Providence, Rhode Island.

[2] L. M. Adleman and M. A. Huang. Recognizing primes in random polynomial time. In *Proceedings 19th STOC* (1986), ACM Press, pp. 462–469. New-York City, May 25–27, 1987.

[3] L. M. Adleman, C. Pomerance, and R. S. Rumely. On distinguishing prime numbers from composite numbers. *Annals of Math. 117* (1983), 173–206.

[4] A. O. L. Atkin. Manuscript. Lecture Notes of a conference, Boulder (Colorado), Aug. 1986.

[5] A. O. L. Atkin. The number of points on an elliptic curve modulo a prime. Preprint, 1991.

[6] A. O. L. Atkin and F. Morain. Elliptic curves and primality proving. Research Report 1256, INRIA, June 1990.

[7] A. O. L. Atkin and F. Morain. Finding suitable curves for the elliptic curve method of factorization. To appear in *Mathematics of Computation.* Also available as INRIA Research Report no 1547, Mar. 1991.

[8] R. Balasubramanian and M. R. Murty. Elliptic pseudoprimes, II. Submitted for publication.

[9] P. Beauchemin, G. Brassard, C. Crépeau, C. Goutier, and C. Pomerance. The generation of random numbers that are probably prime. *J. Cryptology 1* (1988), 53–64.

[10] E. R. Berlekamp. Factoring polynomials over large finite fields. *Math. Comp. 24*, 111 (1970), 713–735.

[11] W. E. H. Berwick. Modular invariants expressible in terms of quadratic and cubic irrationalities. *Proc. London Math. Soc. 28* (1928), 53–69.

[12] B. J. Birch. Weber's class invariants. *Mathematika 16* (1969), 283–294.

[13] A. Borel, S. Chowla, C. S. Herz, K. Iwasawa, and J. P. Serre. *Seminar on complex multiplication.* No. 21 in Lect. Notes in Math. Springer, 1966.

[14] W. Bosma. Primality testing using elliptic curves. Tech. Rep. 85-12, Math. Instituut, Universiteit van Amsterdam, 1985.

[15] W. Bosma and M.-P. van der Hulst. Faster primality testing. In *Advances in Cryptology* (1990), J.-J. Quisquater, Ed., vol. 434 of *Lect. Notes in Computer Science*, Springer-Verlag, pp. 652–656. Proc. Eurocrypt '89, Houthalen, April 10–13.

[16] G. Brassard. *Modern Cryptology*, vol. 325 of *Lect. Notes in Computer Science.* Springer-Verlag, 1988.

[17] R. P. Brent. Some integer factorization algorithms using elliptic curves. *Australian Computer Science Communications 8* (1986), 149–163.

[18] J. Brillhart, D. H. Lehmer, and J. L. Selfridge. New primality criteria and factorizations of $2^m \pm 1$. *Math. Comp. 29*, 130 (1975), 620–647.

[19] J. Brillhart, D. H. Lehmer, J. L. Selfridge, B. Tuckerman, and S. S. Wagstaff, Jr. *Factorizations of* $b^n \pm 1$, $b = 2, 3, 5, 6, 7, 10, 11, 12$ *up to high powers*, 2 ed. No. 22 in Contemporary Mathematics. AMS, 1988.

[20] D. A. Buell. Small class numbers and extreme values of L-functions of quadratic fields. *Math. Comp. 31*, 139 (1977), 786–796.

[21] J. P. Buhler, R. E. Crandall, and M. A. Penk. Primes of the form $n! \pm 1$ and $2 \cdot 3 \cdot 5 \cdots p \pm 1$. *Math. Comp. 38*, 158 (Apr. 1982), 639–643.

[22] J. W. S. Cassels. Diophantine equations with special reference to elliptic curves. *J. London Math. Soc. 41* (1966), 193–291.

[23] D. Chatelain. Bases normales de l'anneau des entiers de certaines extensions abéliennes de Q. *Comptes Rendus de l'Académie des Sciences de Paris 270* (mars 1970), 557–560. Ser. A.

[24] S. Chowla. An extension of Heilbronn's class number theorem. *Quarterly J. Math. Oxford Ser. 5* (1934), 304–307.

[25] D. V. Chudnovsky and G. V. Chudnovsky. Sequences of numbers generated by addition in formal groups and new primality and factorization tests. Research report RC 11262, IBM, Yorktown Heights, 1985.

[26] H. Cohen. Cryptographie, factorisation et primalité : l'utilisation des courbes elliptiques. In *Comptes Rendus de la Journée annuelle de la Société Mathématique de France* (Paris, Jan. 1987).

[27] H. Cohen and A. K. Lenstra. Implementation of a new primality test. *Math. Comp. 48*, 177 (1987), 103–121.

[28] H. Cohen and H. W. Lenstra, Jr. Primality testing and Jacobi sums. *Math. Comp. 42*, 165 (1984), 297–330.

[29] H. Cohn. *A classical invitation to algebraic numbers and class fields.* Universitext. Springer-Verlag, 1978.

[30] H. Cohn. *Advanced number theory.* Dover, New York, 1980.

[31] H. Cohn. *Introduction to the construction of class fields.* No. 6 in Cambridge studies in advanced mathematics. Cambridge University Press, 1985.

[32] G. Cornacchia. Su di un metodo per la risoluzione in numeri interi dell' equazione $\sum_{h=0}^{n} C_h x^{n-h} y^h = P$. *Giornale di Matematiche di Battaglini 46* (1908), 33–90.

[33] D. A. Cox. *Primes of the form* $x^2 + ny^2$. John Wiley & Sons, 1989.

[34] M. Deuring. Die Klassenkörper der komplexen Multiplikation. In *Enzyklopädie der mathematischen Wissenschaften mit Einschluss ihrer Anwendungen*, vol. Bd 1, H. 10, T. 2. Teubner, Stuttgart, 1958.

[35] L. E. DICKSON. *History of the Theory of Numbers*, vol. I, II, III. Chelsea, New York, 1952.

[36] D. R. DORMAN. Special values of the elliptic modular function and factorization formulae. *J. für die reine und angew. Math. 383* (1988), 207–220.

[37] R. FRICKE. *Lehrbuch der Algebra, III.* F. Vieweg and Sohn, Braunschweig, 1928.

[38] C. F. GAUSS. *Disquisitiones Arithmeticae*, 1st ed. G. Fleischer, 1801. Leipzig; English translation by A. A. Clarke, Yale Univ. Press, New York, 1966; revised English translation by W. C. Waterhouse, Springer-Verlag, New York, 1988.

[39] K. GIRSTMAIR. Über die praktische Auflösung von Gleichungen höheren Grades. *Mathematische Semesterberichte Band XXXIV/1987*, Heft 2 (1987), 213–245.

[40] S. GOLDWASSER AND J. KILIAN. Almost all primes can be quickly certified. In *Proc. 18th STOC* (1986), ACM, pp. 316–329. May 28–30, Berkeley.

[41] D. M. GORDON. On the number of elliptic pseudoprimes. *Math. Comp. 52*, 185 (Jan. 1989), 231–245.

[42] A. G. GREENHILL. Table of complex multiplication moduli. *Proc. London Math. Soc. (1) 21* (1891).

[43] B. H. GROSS. *Arithmetic on Elliptic Curves with Complex Multiplication*, vol. 776 of *Lect. Notes in Math.* Springer Verlag, 1980.

[44] B. H. GROSS AND D. B. ZAGIER. On singular moduli. *J. für die reine und angew. Math. 355* (1985), 191–220.

[45] J.-C. HERVÉ, F. MORAIN, D. SALESIN, B. SERPETTE, J. VUILLEMIN, AND P. ZIMMERMANN. Bignum: A portable and efficient package for arbitrary precision arithmetic. Rapport de Recherche 1016, INRIA, avril 1989.

[46] M.-D. A. HUANG. Factorization of polynomials over finite fields and factorization of primes in algebraic number fields. In *Proc. 16th ACM STOC* (1984), ACM, pp. 175–182. Washington, apr. 30 – may 2.

[47] M.-D. A. HUANG. Riemann hypothesis and finding roots over finite fields. In *Proc. 17th ACM STOC* (1985), ACM, pp. 121–130. Providence, Rhode Island, May 6–8.

[48] K. IRELAND AND M. ROSEN. *A classical introduction to modern number theory*, vol. 84 of *Graduate Texts in Mathematics*. Springer, 1982.

[49] E. KALTOFEN, T. VALENTE, AND N. YUI. An improved Las Vegas primality test. Research Report 89-12, Rensselaer Polytechnic Institute, Troy, New York, May 1989.

[50] E. KALTOFEN AND N. YUI. Explicit construction of the Hilbert class fields of imaginary quadratic fields with class numbers 7 and 11. In *Proc. EUROSAM '84* (1984), J. Fitch, Ed., vol. 174 of *Lect. Notes in Computer Science*, Springer–Verlag, pp. 310–320. Cambridge (England), July 9–11.

[51] E. Kaltofen and N. Yui. Explicit construction of the Hilbert class fields of imaginary quadratic fields by integer lattice reduction. Research Report 89-13, Renseelaer Polytechnic Institute, May 1989.

[52] D. E. Knuth. *The Art of Computer Programming: Seminumerical Algorithms.* Addison-Wesley, 1981.

[53] S. Lang. *Elliptic functions.* Addison-Wesley, 1973.

[54] A. K. Lenstra and H. W. Lenstra, Jr. Algorithms in number theory. In *Handbook of Theoretical Computer Science*, J. van Leeuwen, Ed., vol. A: Algorithms and Complexity. North Holland, 1990, ch. 12, pp. 674–715.

[55] A. K. Lenstra and M. S. Manasse. Factoring by electronic mail. In *Advances in Cryptology* (1990), J.-J. Quisquater, Ed., vol. 434 of *Lect. Notes in Computer Science*, Springer-Verlag, pp. 355–371. Proc. Eurocrypt '89, Houthalen, April 10–13.

[56] H. W. Lenstra, Jr. Elliptic curves and number theoretic algorithms. Tech. Rep. Report 86-19, Math. Inst., Univ. Amsterdam, 1986.

[57] H. W. Lenstra, Jr. Factoring integers with elliptic curves. *Annals of Math. 126* (1987), 649–673.

[58] J.-F. Mestre. La méthode des graphes. Exemples et applications. In *Proc. of the International Conference on class numbers and fundamental units of algebraic number fields* (Nagoya, 1986), Nagoya Univ., pp. 217–242. Katata (Japan).

[59] J.-F. Mestre and V. S. Miller. Computing j via $X_0(N)$. In preparation, Mar. 1990.

[60] P. Mihailescu. A primality test using cyclotomic extensions. In *Applied Algebra, Algebraic Algorithms and Error–Correcting Codes* (1989), vol. 357 of *Lect. Notes in Computer Science*, Springer-Verlag, pp. 310–323. Proc. AAECC-6, Rome, July 1988.

[61] I. Miyamoto and M. R. Murty. Elliptic pseudoprimes. *Math. Comp. 53*, 187 (July 1989), 415–430.

[62] P. L. Montgomery. Speeding the Pollard and elliptic curve methods of factorization. *Math. Comp. 48*, 177 (Jan. 1987), 243–264.

[63] F. Morain. Implementation of the Atkin-Goldwasser-Kilian primality testing algorithm. Rapport de Recherche 911, INRIA, Octobre 1988.

[64] F. Morain. Construction of Hilbert class fields of imaginary quadratic fields and dihedral equations modulo p. Rapport de Recherche 1087, INRIA, Septembre 1989.

[65] F. Morain. Résolution d'équations de petit degré modulo de grands nombres premiers. Rapport de Recherche 1085, INRIA, Septembre 1989.

[66] F. Morain. Solving generalized dihedral equations. Manuscript, Aug. 1990.

[67] F. Morain. Distributed primality proving and the primality of $(2^{3539}+1)/3$. In *Advances in Cryptology – EUROCRYPT '90* (1991), I. B. Damgård, Ed., vol. 473 of *Lect. Notes in Computer Science*, Springer–Verlag, pp. 110–123. Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques, Aarhus, Denmark, May 21–24, 1990.

[68] F. Morain. Elliptic curves, primality proving and some Titanic primes. In *Journées Arithmétiques 1989* (1992), vol. 198–199-200 of *Astérisque*, SMF, pp. 245–251.

[69] F. Morain. Prime values of partition numbers and the primality of $p(1840926)$. Preprint, Aug. 1992.

[70] F. Morain and J. Nicolas. On Cornacchia's algorithm. In preparation, Mar. 1990.

[71] F. Morain and J. Olivos. Speeding up the computations on an elliptic curve using addition-subtraction chains. *R.A.I.R.O. Theoretical Informatics and Applications 24*, 6 (1990), 531–543.

[72] M. Newman, D. Shanks, and H. C. Williams. Simple groups of square order and an interesting sequence of primes. *Acta Arithmetica XXXVIII* (1980), 129–140.

[73] A. Ogg. *Modular forms and Dirichlet series.* W. A. Benjamin, Inc., New York and Amsterdam, 1969.

[74] J. C. Parnami and A. R. Rajwade. A new cubic character sum. *Acta Arithmetica XL* (1982), 347–356.

[75] C. Pomerance. Very short primality proofs. *Math. Comp. 48*, 177 (1987), 315–322.

[76] D. Poulakis. Evaluation d'une somme cubique de caractères. *J. Number Theory 27* (1987), 41–45.

[77] V. R. Pratt. Every prime has a succint certificate. *SIAM J. Comput. 4* (1975), 214–220.

[78] A. R. Rajwade. Certain classical congruences via elliptic curves. *J. London Math. Soc. 2*, 8 (1974), 60–62.

[79] A. R. Rajwade. The diophantine equation $y^2 = x(x^2+21dx+112d^2)$ and the conjectures of Birch and Swinnerton-Dyer. *J. Australian Math. Soc. 24* (1977), 286–295. (Series A).

[80] P. Ribenboim. *The book of prime number records*, 2nd ed. Springer, 1989.

[81] N. W. Rickert. Efficient reduction of quadratic forms. In *Computers and Mathematics* (1989), Springer-Verlag, pp. 135–139. Proc. Conf. on Computers ans Mathematics, June 1989, Cambridge (Massachussets).

[82] R. Schertz. Die singulären Werte der Weberschen Funktionen f, f_1, f_2, γ_2, γ_3. *J. für die reine und angew. Math. 286-287* (1976), 46–74.

[83] R. Schoof. Elliptic curves over finite fields and the computation of square roots mod p. *Math. Comp. 44* (1985), 483–494.

[84] J. P. Serre. *Cours d'arithmétique.* PUF, 1970.

[85] D. Shanks. Class number, a theory of factorization, and genera. In *Proc. Symp. Pure Math. vol. 20* (1971), AMS, pp. 415–440.

[86] D. Shanks. Five number theoretic algorithms. In *Proc. 2nd Manitoba Conference on Numerical Mathematics* (1972), pp. 51–70.

[87] G. Shimura and Y. Taniyama. *Complex multiplication of abelian varieties and its applications to number theory*, vol. 6 of *Publ. Math. Soc. Japan.* Math. Soc. Japan, 1961.

[88] H. M. Stark. On the "gap" in a theorem of Heegner. *J. Number Theory 1* (1969), 16–27.

[89] B. Vallée. Une approche géométrique des algorithmes de réduction des réseaux en petite dimension, 1986. Thèse, Université de Caen.

[90] G. N. Watson. Ramanujans Vermutung über Zerfällungsanzahlen. *J. für die reine und angew. Math. 179* (1938), 97–128.

[91] H. Weber. *Lehrbuch der Algebra*, vol. I, II, III. Chelsea Publishing Company, New York, 1902.

[92] H. C. Williams. Primality testing on a computer. *Ars Combinatoria 5* (1978), 127–185.

[93] H. C. Williams. Some primes with interesting digit patterns. *Math. Comp. 32*, 144 (Oct. 1978), 1306–1310.

[94] H. C. Williams. Effective primality tests for some integers of the forms $A5^n - 1$ and $A7^n - 1$. *Math. Comp. 48*, 177 (Jan. 1987), 385–403.

[95] H. C. Williams and H. Dubner. The primality of $R1031$. *Math. Comp. 47*, 176 (1986), 703–711.

[96] H. C. Williams and C. R. Zarnke. Some algorithms for solving a cubic congruence modulo p. *Utilitas Mathematica 6* (1974), 285–306.

[97] M. C. Wunderlich. A performance analysis of a simple prime-testing algorithm. *Math. Comp. 40*, 162 (1983), 709–714.