CONSTRUCTING ELLIPTIC CURVES OF PRIME ORDER

Reinier Bröker, Peter Stevenhagen

Abstract. We present a very efficient algorithm to construct an elliptic curve E and a finite field $\mathbf{F}$ such that the order of the point group $E(\mathbf{F})$ is a given prime number N. Heuristically, this algorithm only takes polynomial time $\widetilde{O}((\log N)^3)$, and it is so fast that it may profitably be used to tackle the related problem of finding elliptic curves with point groups of prime order of prescribed size.

We also discuss the impact of the use of high level modular functions to reduce the run time by large constant factors and show that recent gonality bounds for modular curves imply limits on the time reduction that can be obtained.

1. Introduction

For almost twenty years, the discrete logarithm problem in the group of points of on an elliptic curve over a finite field has been used as the basis of elliptic curve cryptography. Partly because of this application, the mathematically natural question on how to generate elliptic curves over finite fields with a given number of points has attracted considerable attention [16, 15, 2, 5]. More in particular [22, 14], one is led to the question of how to efficiently generate 'cryptographic' elliptic curves for which the order of the point group is a *prime* number. For elliptic curves of prime order N, the discrete logarithm problem is currently supposed to be untractable for $N \gg 10^{60}$.

Section 2 deals with the problem of constructing a finite field $\mathbf{F}$ and an elliptic curve $E/\mathbf{F}$ having a prescribed prime number N of $\mathbf{F}$-rational points. We show that, on prime input N, such an elliptic curve can be constructed efficiently, in heuristic polynomial time $\widetilde{O}((\log N)^3)$, using traditional complex multiplication (CM) methods. Here the $\widetilde{O}$-notation indicates that factors that are of logarithmic order in the main term have been disregarded. The finite field $\mathbf{F}$ over which E is constructed will be of prime order p for some p sufficiently close to N. The algorithm takes less time then algorithms that *prove* the primality of the input N. However, if the given input is known to be prime, the output $E/\mathbf{F}_p$ is guaranteed to be an elliptic curve over a prime field $\mathbf{F}_p$ having exactly N points over $\mathbf{F}_p$. Because of

2000 *Mathematics Subject Classification.* Primary 14H52, Secondary 11G15.

This paper was completed at the Fields Institute in Toronto. We thank this institute for its hospitality and support.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX
version 20061108

its efficiency, the range of our method amply exceeds the range of prime values in current cryptographic use.

In Section 3, we discuss the related problem of constructing an elliptic curve that has a point group of prime order of prescribed *size*. Unlike the earlier problem, this may be tackled efficiently by 'naive' methods that generate curves using trial and error and exploit the efficiency of point counting on elliptic curves. We describe the 'traditional CM-algorithm' that constructs, on input of an integer $k \in \mathbf{Z}_{\geq 3}$, an elliptic curve with prime order of k decimal digits, and show that the run time of this algorithm is $O(k^{4+\varepsilon})$ for every $\varepsilon > 0$. It becomes $\widetilde{O}(k^4)$ if we are content with probable primes instead of proven primes. As a consequence, we deduce that the fastest way to tackle the problem in this Section is to first *fix* a (probable) prime N of k digits and then apply our CM-algorithm from Section 2 for that N.

From a practical point of view, CM-methods are hampered by the enormous size of the auxiliary *class polynomials* entering the construction, and since the time of Weber [25], extensive use has been made of 'small' modular functions to perform CM-constructions. We discuss the practical improvements of this nature in Section 4, and show how recent results on the gonality of modular curves imply upper bounds on the gain that can result from such methods.

A final section contains numerical illustrations of the methods discussed.

2. An efficient CM-construction

We start with the fundamental problem of realizing a prime number $N > 3$ as the group order of an elliptic curve E defined over some finite field $\mathbf{F}_q$. By Hasse's theorem, the order of the point group $E(\mathbf{F}_q)$ is an element of the Hasse interval

$$\mathcal{H}_q = [q + 1 - 2\sqrt{q}, q + 1 + 2\sqrt{q}]$$

around $q + 1$. The relation $N \in \mathcal{H}_q$ is actually symmetric in N and q, as we have $N \in \mathcal{H}_q \iff q \in \mathcal{H}_N$. Consequently, a necessary condition for the existence of a curve with N points is that the Hasse interval $\mathcal{H}_N$ contains a prime power q. As the set of integers N for which $\mathcal{H}_N$ contains a 'true' prime power is a zero density subset of $\mathbf{Z}_{>0}$, we may and will restrict to elliptic curves defined over *prime fields* $\mathbf{F}_q = \mathbf{F}_p$. If p is a prime number in $\mathcal{H}_N$, then an elliptic curve $E/\mathbf{F}_p$ with $\#E(\mathbf{F}_p) = N$ always exists. It follows from $p = N \in \mathcal{H}_N$ that elliptic curves of prime order N exist for every prime N, but our algorithm will typically construct curves over prime fields different from $\mathbf{F}_N$. This is certainly desirable from a cryptographic point of view, as curves of order N over $\mathbf{F}_N$ are cryptographically unsafe: the discrete logarithm problem on them can be transformed [20] into a discrete logarithm problem for the *additive* group of $\mathbf{F}_N$ that is easily solved.

Let p be any prime in $\mathcal{H}_N$, and write $N = p + 1 - t$. Then we have $t \neq 0$, as the primes p and $N > 3$ are not consecutive numbers. It is well known that a curve $E/\mathbf{F}_p$ has N points over $\mathbf{F}_p$ if and only if the Frobenius morphism $F_p : E \to E$ satisfies the quadratic equation

$$F_p^2 - tF_p + p = 0$$

in the endomorphism ring $\mathrm{End}(E)$. The means that the subring $\mathbf{Z}[F_p] \subseteq \mathrm{End}(E)$ generated by Frobenius is isomorphic to the imaginary quadratic order $\mathcal{O}_\Delta$ of discriminant $\Delta = t^2 - 4p < 0$, with F_p corresponding to the element $(t + \sqrt{\Delta})/2 \in \mathcal{O}_\Delta$ of trace t and norm p. As t is nonzero, the curve is ordinary. Conversely, if the endomorphism ring $\mathrm{End}(E)$ of an ordinary elliptic curve $E/\mathbf{F}_p$ contains an element F of degree p and trace $F + \hat{F} = t$, and therefore a subring isomorphic to $\mathcal{O}_\Delta$, then one of the twists of E over $\mathbf{F}_p$ has N points. Thus, constructing an elliptic curve having N points over $\mathbf{F}_p$ is the same problem as constructing an ordinary elliptic curve over $\mathbf{F}_p$ for which the endomorphism ring is isomorphic to some quadratic order containing $\mathcal{O}_\Delta$.

Over the complex numbers, the j-invariants of curves with endomorphism ring isomorphic to $\mathcal{O}_\Delta$ are the roots of the *Hilbert class polynomial*

$$P_\Delta = \prod_{[Q] \in \mathrm{Pic}(\mathcal{O}_\Delta)} (X - j(\tau_Q)) \in \mathbf{Z}[X].$$

Here $j : \mathbf{H} \to \mathbf{C}$ is the classical modular function on the complex upper half plane $\mathbf{H}$ with Fourier expansion $j(z) = 1/q + 744 + \ldots$ in $q = \exp(2\pi i z)$, and the points $\tau_Q = \frac{-b+\sqrt{\Delta}}{2a} \in \mathbf{H}$ correspond in the standard way to the ideal classes

$$[Q] = [\mathbf{Z} \cdot a + \mathbf{Z} \cdot \frac{-b + \sqrt{\Delta}}{2}] \in \mathrm{Pic}(\mathcal{O}_\Delta).$$

The polynomial P_Δ has integer coefficients, so it can be computed by approximating the roots $j(\tau_Q) \in \mathbf{C}$ with sufficient accuracy. Alternatively, one can use p-adic algorithms [7, 4, 5] to compute P_Δ.

The polynomial P_Δ splits completely modulo p, and its roots in $\mathbf{F}_p$ are the j-invariants of the elliptic curves $E/\mathbf{F}_p$ with endomorphism ring isomorphic to $\mathcal{O}_\Delta$. If $j_0 \neq 0, 1728 \in \mathbf{F}_p$ is one of these roots, then the elliptic curve

$$E : Y^2 = X^3 + aX - a \qquad \text{with } a = \frac{27 j_0}{4(1728 - j_0)} \in \mathbf{F}_p \tag{2.1}$$

has j-invariant j_0. If we have $N \cdot P = 0$ for our prime number N and $P = (1, 1) \in E(\mathbf{F}_p)$, then $E(\mathbf{F}_p)$ has order N. Otherwise the quadratic twist $E' : Y^2 = X^3 + g^2 aX - g^3 a$ with $g \in \mathbf{F}_p^*$ a non-square has N points over $\mathbf{F}_p$. In the special cases $j_0 = 0, 1728$ there are a few more twists to consider.

As we only need $\mathrm{End}(E)$ to *contain* an order isomorphic to $\mathcal{O}_\Delta$, we can replace Δ in the argument above by the field discriminant $D = \mathrm{disc}(\mathbf{Q}(\sqrt{\Delta}))$. For most t, the discriminant $\Delta = \Delta(p) = t^2 - 4p$ is of roughly the same size as p and N. Moreover, the associated field discriminant D, which is essentially the squarefree part of Δ, will be of the same size as Δ itself for most Δ. As computing the Hilbert class polynomial $P_D \in \mathbf{Z}[X]$, which has degree $h(D) \approx \sqrt{|D|}$ and coefficients of size $\widetilde{O}(\sqrt{|D|})$, takes time at least linear in D, the CM-algorithm will have *exponential* run time $\widetilde{O}(N)$ for 'most' choices of primes $p \in \mathcal{H}_N$.

There is however a way to select primes $p \in \mathcal{H}_N$ for which the field discriminant $D = D(p) = \mathrm{disc}(\mathbf{Q}(\sqrt{\Delta(p)}))$ is only of *polynomial size* in $\log N$. What we want is a discriminant D such that the order $\mathcal{O}_D$ contains an element π of prime norm p for which we have $N = p+1-\mathrm{Trace}(\pi) = \mathrm{Norm}(1-\pi)$. Exploiting the symmetry in p and N and writing $\alpha = 1-\pi$, we can also say equivalently that we want an order $\mathcal{O}_D$ containing an element α of norm N with the property that $p = N+1-\mathrm{Trace}(\alpha) = \mathrm{Norm}(1-\alpha)$ is prime. Note that if $\pi \in \mathcal{O}_D$ has prime norm $p > 2$, then $\alpha = 1-\pi$ will have *even* norm in case the residue class field of the primes over 2 in $\mathcal{O}_D$ is the field of 2 elements. For prime values $N > 5$, or more generally for odd $N > 5$, this means that we can only use discriminants D congruent to 5 modulo 8.

In principle, one can find the *smallest* D for which $\mathcal{O}_D$ contains an element α of norm N such that $\mathrm{Norm}(1-\alpha)$ is prime. To do so, one splits the prime N in the imaginary quadratic orders $\mathcal{O}_D$ with $\left(\frac{D}{N}\right) = 1$ as $(N) = \mathfrak{a}\bar{\mathfrak{a}}$ for descending values of $D = -3, -11, -19, \ldots$ congruent to 5 mod 8 until we find a value of D such that $\mathfrak{a} = \alpha\mathcal{O}_D$ is principal with generator α and $N+1 \pm \mathrm{Trace}(\alpha) = \mathrm{Norm}(1 \pm \alpha)$ is prime. Now assume the standard *heuristical* arguments that the prime $\mathfrak{a} \subset \mathcal{O}_D$ over N will be principal with 'probability' $1/h(D)$ and that $\mathrm{Norm}(1 \pm \alpha) \approx N$ will be prime with 'probability' $1/\log N$. Then it is shown in [6, Theorem 4.1] that the expected value of the smallest suitable discriminant D found in this way will be

$$D = \widetilde{O}((\log N)^2).$$

Moreover, as the principality of the ideal $\mathfrak{a} \subset \mathcal{O}_D$ lying over N can be tested effiently using the 1908 algorithm of Cornacchia [23], we can expect to find this D in time $O((\log N)^{4+\varepsilon})$.

Cornacchia's algorithm explicitly computes the positive integers x, y that satisfy

$$x^2 - Dy^2 = 4N$$

in case such integers exist. For $D < -4$, such x, y are uniquely determined by N. If found, the element $\alpha = (x+\sqrt{D})/2 \in \mathcal{O}_D$ has norm N, and we hope that one of the elements $\mathrm{Norm}(1 \pm \alpha) = N+1 \pm x$ is prime. Cornacchia's algorithm consists of the computation of a square root x_0 mod N of D mod N followed by what is basically the Euclidean algorithm for x_0 and N. It takes probabilistic time $\widetilde{O}((\log N)^2)$ for each D. Performing Cornacchia's algorithm for $D = -3, -11, \ldots$ up to a bound of size $(\log N)^2$ takes time $\widetilde{O}((\log N)^4)$, and this dominates the run time of the algorithm. We will lower the heuristic run time to $\widetilde{O}((\log N)^3)$ by applying an idea attributed to J. Shallit in [18] to speed up the algorithm.

We start from the observation that N splits into principal primes in $\mathcal{O}_D$ if and only if N splits completely in the Hilbert class field H_D of $\mathbf{Q}(\sqrt{D})$. If this is the case, then N also splits completely in the genus field $G_D \subseteq H_D$, which is obtained by adjoining the square roots of $p^* = (-1)^{(p-1)/2}p$ to $\mathbf{Q}(\sqrt{D})$ for all odd prime divisors $p \mid D$. We have $\left(\frac{p^*}{N}\right) = \left(\frac{N}{p}\right) = 1$ for all odd primes dividing D, and we can save time if we do not try increasing values of D until we hit the smallest suitable D, but rather construct a suitable discriminant D from a generating set of 'good' primes p for which we know that p^* is a square modulo N. If we only consider primes p of size $O(\log N)$, the time needed to compute the values $\sqrt{p^*} \bmod N$ for these primes is $\widetilde{O}((\log N)^3)$.

Our algorithm consists of multiple 'search rounds' for a suitable discriminant D, where in each round we increase the size of the 'basis' of primes we use. First we take the primes between 0 and $\log N$ and see whether we can find a suitable $D \equiv 5 \bmod 8$ with $|D| < (\log N)^2$ made up of primes from this basis. If no such D exists, we add the 'good' primes between $\log N$ and $2\log N$ to our basis, and look for a suitable D with $|D| < (2\log N)^2$ created from this enlarged basis, and so on. In this way, we encounter in the r-th round all discriminants D with $|D| < (r\log N)^2$ that are made up of prime factors below $r\log N$. Asymptotically (cf. the 'analytic tidbit' in [19]), this is a *positive* fraction $1 - \log 2 \approx 0.30685$ of all discriminants below $(r\log N)^2$. As the smoothness properties of D play no role in our heuristics, we still expect to find a suitable discriminant of size $\widetilde{O}((\log N)^2)$. Thus, we expect the algorithm below to terminate after a number of rounds that is polynomial in $\log\log N$. In practice, (cf. Section 4), this number is usually 1.

2.2. Algorithm.
Input: a prime number N.
Output: a prime p number and an elliptic curve $E/\mathbf{F}_p$ with $\#E(\mathbf{F}_p) = N$.

```
1. Put $r \leftarrow 0$, and create an empty table $S$.
2. Compute for all odd primes $p \in [r\log N, (r+1)\log N]$ that satisfy $\left(\frac{N}{p}\right) = 1$ a
   square root $\sqrt{p^*} \bmod N$, and add the pairs $(p^*, \sqrt{p^*} \bmod N)$ to the table $S$.
3. For each product $(D, \sqrt{D} \bmod N) = (\prod_i p_i^*, \prod_i \sqrt{p_i^*} \bmod N)$ of distinct ele-
   ments of $S$ that satisfies $\prod_i p_i^* < (r\log N)^2$ and $D \equiv 5 \bmod 8$, do the following.
   3a. Use the value $\sqrt{D} \bmod N$ and Cornacchia's algorithm to compute $x, y > 0$
       satisfying $x^2 - Dy^2 = 4N$.
   3b. For each solution found in step 3a, test whether $p = N + 1 \pm x$ is a probable
       prime. If it is, compute the Hilbert class polynomial $P_D \in \mathbf{Z}[X]$, compute
       a root $j_0$ of $\overline{P}_D \in \mathbf{F}_p[X]$, return the twist of the elliptic curve (2.1) that
       has $N$ points, and stop. If no root or no twist is found, then $p = N + 1 \pm x$
       is not prime and we continue with the next solution.
4. Put $r \leftarrow r + 1$ and go back to step 2.
```

A heuristic analysis of the Algorithm above leads to the following.

2.3. Theorem. *On input of a prime N, Algorithm 2.2 returns a prime p and an elliptic curve $E/\mathbf{F}_p$ with $\#E(\mathbf{F}_p) = N$. Under heuristic assumptions, its run time is $\widetilde{O}((\log N)^3)$ for every $\varepsilon > 0$.*

Proof. As the smoothness properties of D are irrelevant in the heuristic analysis detailed in [6], the smallest suitable D found by our Algorithm, which restricts to the positive density subset of discriminants, will be of size $\widetilde{O}((\log N)^2)$. The expected number r of rounds of our Algorithm will therefore be small, at most polynomial in $\log\log N$, and in view of our $\widetilde{O}$-notation we may prove our Theorem by focusing on the time needed for a single round of the Algorithm, which consists of Steps 2 and 3.

In Step 2 we have to find primes up to $(r+1)\log N$. As we only need to test primality of integers of size $\widetilde{O}(\log N)$, the time needed to find these primes is negligible. For all the $\widetilde{O}(\log N)$ primes we find, we need to test which primes are 'good', i.e., which primes satisfy $\left(\frac{N}{p}\right) = 1$. The time needed for this computation is also negligible. The bottleneck in Step 2 is the computation of the square roots of p^* mod N for the good primes p. Each square root computation takes time $\widetilde{O}((\log N)^2)$, so Step 2 takes time $\widetilde{O}((\log N)^3)$.

For each of the $O((\log N)^2)$ products $(D, \sqrt{D} \bmod N)$ formed in Step 3, we run the Euclidean algorithm part of Cornacchia's algorithm in Step 3a in time $\widetilde{O}(\log N)$. This takes time $\widetilde{O}((\log N)^3)$. We expect to find $O(\log N)$ solutions (x, y) from Step 3a for which we have to test primality of $N+1\pm x$ in Step 3b. A cheap Miller-Rabin test, which takes time $\widetilde{O}((\log N)^2)$, suffices for our purposes, and leads to a total time $\widetilde{O}((\log N)^3)$ spent on primality testing.

Once we encounter a probable prime $p = N + 1 \pm x$ for some discriminant D, we compute the Hilbert class polynomial P_D. As D is of size $O((r \log N)^2)$, this takes time $\widetilde{O}((\log N)^2)$. Computing a root j_0 of P_D, a polynomial of degree $h(D) = \widetilde{O}(\log N)$, modulo the prime $p \approx N$ once more takes time $\widetilde{O}((\log N)^3)$. To test which curve of j-invariant j_0 has N points, we may have to compute all isomorphism classes over $\mathbf{F}_p$ of elliptic curves with j-invariant j_0 until we find one. There are at most 6 of these classes ('twists'), and for the class of E we need to test the equality $N \cdot P = 0$ for a point P on E. This only takes time $\widetilde{O}((\log N)^2)$, and we conclude that the entire round of the algorithm runs in time $\widetilde{O}((\log N)^3)$.

Even though we have only found a *probable* prime p in the beginning of Step 3b, the equality $N \cdot P = 0$ on E tested in this Step exhibits a point of order N on E, which *proves* that p is actually prime. □

The low asymptotic running time of our Algorithm is illustrated by the size of some of the examples in Section 4. As several steps in the algorithm are no faster than $\widetilde{O}((\log N)^3)$, it seems that we have obtained an optimal result for a CM-solution to our problem.

3. Point groups of given prime size

Closely related to the problem of constructing an elliptic curve of prescribed prime order N is the problem of constructing a curve for which the group order is a prime in a given interval. For concreteness sake, we take the interval as $[10^{k-1}, 10^k)$, so the problem becomes the efficient construction of an elliptic curve over a finite field such that the group order is a prime of exactly k decimal digits.

If we insist on a curve with *proven* prime order, we cannot hope for an algorithm with a faster run time than $O(k^4)$, since the fastest known algorithm [3] to rigorously prove primality of an integer $N \approx 10^k$ has expected run time $O((\log N)^{4+\varepsilon}) = O(k^{4+\varepsilon})$ for all $\varepsilon > 0$. The naive algorithm of selecting a prime p of k decimal digits and trying random elliptic curves over $\mathbf{F}_p$ until we find one of prime order already has a heuristic run time that comes close to this 'optimal run time'. Indeed, counting the number of points of an elliptic curve $E/\mathbf{F}_p$ takes heuristic time $\widetilde{O}((\log p)^4)$ using the improvements made by Atkin and Elkies to Schoof's original point counting algorithm [23]. Even though the distribution of group orders of elliptic curves over $\mathbf{F}_p$ over the Hasse interval is not exactly uniform, it follows as in [17, Section 1] that, heuristically, we have to try $O(\log p)$ curves over $\mathbf{F}_p$ until we find one of prime order. This leads to a heuristic run time $\widetilde{O}(k^5)$.

As was noted by many people [8, 14], we can also use complex multiplication techniques to tackle the problem. Unlike our Algorithm 2.2, which starts with a desired prime value N for the group order and computes a suitable prime field $\mathbf{F}_p$ over which the curve can be constructed, these algorithms compute primes p splitting into principal primes π and $\bar{\pi}$ in some *fixed* quadratic ring $\mathcal{O}_D$, and construct a curve over $\mathbf{F}_p$ having CM by $\mathcal{O}_D$ and N points when $N = \mathrm{Norm}(1-\pi)$ is found to be prime. As before, we can test whether a given prime p splits into principal primes in $\mathcal{O}_D$ by computing a value of $\sqrt{D} \bmod p$ for $\left(\frac{D}{p}\right) = 1$ and applying Cornacchia's algorithm. In case $\mathcal{O}_D$ has class number 1, i.e., for $D = -3, -11, -19, -43, -67, -163$, we can see whether p splits in $\mathcal{O}_D$ by only looking at $p \bmod D$.

Subject to the congruence condition $D \equiv 5 \bmod 8$, we can take *any* fundamental discriminant. The run time depends on the value of D we choose, the value $D = -3$ being 'optimal'. For cryptographic purposes we need to select D such that the class number of $\mathcal{O}_D$ is at least 200, cf. Section 5.

3.1. Algorithm.
Input: an integer $k \in \mathbf{Z}_{\geq 3}$, and a negative discriminant $D \equiv 5 \bmod 8$.
Output: primes p, q of k decimal digits and an elliptic curve $E/\mathbf{F}_p$ with CM by $\mathcal{O}_D$ and $\#E(\mathbf{F}_p) = q$.

1. Compute $P_D \in \mathbf{Z}[X]$.

2. Pick a random probable prime p that splits into principal primes in $\mathcal{O}_D$ and satisfies
$$10^{k-1} + 2 \cdot 10^{\frac{k-1}{2}} < p < 10^k - 2 \cdot 10^{\frac{k}{2}}.$$

3. Write $p = \pi\overline{\pi} \in \mathcal{O}_D$. If $q = \mathrm{Norm}(1 - \varepsilon\pi)$ is a probable prime for some $\varepsilon \in \mathcal{O}_D^*$, prove the primality of q, compute a root $j \in \mathbf{F}_p$ of $P_D \in \mathbf{F}_p[X]$ and return an elliptic curve $E/\mathbf{F}_p$ with j-invariant j with q points. Else, go to Step 2.

A heuristic analysis of the Algorithm above leads to the following.

3.2. Theorem. *On input of an integer $k \in \mathbf{Z}_{\geq 3}$ and a negative discriminant $D \equiv 5 \bmod 8$, Algorithm 2.4 returns primes p, q of k decimal digits each and an elliptic curve $E/\mathbf{F}_p$ with CM by $\mathcal{O}_D$ and $\#E(\mathbf{F}_p) = q$. Under heuristic assumptions, the run time for fixed D is $O(k^{4+\varepsilon})$ for every $\varepsilon > 0$.*

Proof. To prove that the output of Algorithm 3.1 is correct, we only need to check that the norms q found in Step 2 have k decimal digits. This follows from Hasse's theorem $q \in \mathcal{H}_p$ and the choice of our interval for p.

In Step 1 we have to find a prime p of k decimal digits that splits into principal primes in $\mathcal{O}_D$. Finding a probable prime p of k digits takes time $\widetilde{O}(k^3)$, and with positive probability $(2h(D))^{-1}$ such a prime p splits into principal primes in O_D. For each p found we can test this in time $\widetilde{O}(k^2)$ by computing a value $\sqrt{D} \bmod p$ in case it exists, and use it to apply the Euclidean algorithm part of Cornacchia's algorithm.

If p factors as $p = \pi\overline{\pi}$ in $\mathcal{O}_D$, the 'probability' that $\mathrm{Norm}(1 - \varepsilon\pi)$ is prime is about $1/k$. We expect that we need to perform Step 2 roughly k times, and except for the primality proof of q this takes us time $\widetilde{O}(k^4)$.

A rigorous primality proof of q in Step 3 takes time $O(k^{4+\varepsilon})$ for every $\varepsilon > 0$. Just as in Theorem 2.3, this also proves the primality of p. □

The proof shows that if we are only insist that p, q are *probable* primes of k digits, the run time becomes $\widetilde{O}(k^4)$. This is *slower* than Algorithm 2.2. The fastest way of constructing a curve for which the group order is a probable prime of k digits is therefore to find a random probable prime N of k digits and then run Algorithm 2.2 on this input. Indeed, finding a probable prime N takes time $\widetilde{O}(k^3)$, and so does the application of Algorithm 2.2 on N.

4. Class invariants and gonality

In large examples, the practical performance of Algorithm 2.2 is hampered by the computation of a Hilbert class polynomial P_D in Step 3b. As we noted already, the run time $\widetilde{O}(|D|)$ needed for computing P_D cannot be seriously improved, as the degree $h(D)$ of P_D is of order of magnitude $\sqrt{|D|}$ by the Brauer-Siegel theorem, and the number of digits of its coefficients has a similar order of magnitude $\sqrt{|D|}$. However, already for the moderately small values of D used by our algorithm, the coefficients of P_D are notoriously large.

It was discovered by Weber [25] that one can often work with 'smaller' modular functions than the j-function to generate the Hilbert class field H_D of $\mathbf{Q}(\sqrt{D})$. There are many of these functions, and each of them works for some positive proportion of discriminants. A good example is provided by the Weber function $\mathfrak{f} = \zeta_{48}^{-1}\eta(\frac{z+1}{2})/\eta(z)$, which is related to j by an irreducible polynomial relation

$$\Psi(\mathfrak{f}, j) = (\mathfrak{f}^{24} - 16)^3 - j\mathfrak{f}^{24} = 0$$

of degree 72 in $\mathfrak{f}$ and degree 1 in j. It can be used for all $D \equiv 1 \bmod 8$ coprime to 3. For $D = -71$, the value $\mathfrak{f}(\tau)$ for an appropriate generator τ of $\mathcal{O}_{-71} = \mathbf{Z}[\tau]$ has the irreducible polynomial

$$P_{-71}^{\mathfrak{f}} = X^7 + X^6 - X^5 - X^4 - X^3 + X^2 + 2X + 1 \in \mathbf{Z}[X]$$

that requires less precision to compute from its complex zeroes then it does to compute the Hilbert class polynomial

$$\begin{aligned}P_{-71} = \ & X^7 + 313645809715\, X^6 - 3091990138604570\, X^5 \\ & + 98394038810047812049302\, X^4 - 823534263439730779968091389\, X^3 \\ & + 5138800366453976780323726329446\, X^2 \\ & - 425319473946139603274605151187659\, X \\ & + 737707086760731113357714241006081263\end{aligned}$$

coming from the j-function. The polynomials P_{-71} and $P^{\mathfrak{f}}_{-71}$ have the same type of splitting behavior modulo primes as they generate the same field H_{-71} over $\mathbf{Q}(\sqrt{-71})$. Moreover, the zeroes modulo p of $P^{\mathfrak{f}}_{-71}$ readily give the zeroes of P_{-71} modulo p by the formula $j = \mathfrak{f}^{-24}(\mathfrak{f}^{24} - 16)^3$. A significant speed up in the practical performance of CM-algorithms can be obtained by using functions such as $\mathfrak{f}$ instead of j.

In cases where the value $f(\tau)$ of a modular function f at some $\tau \in \mathbf{Q}(\sqrt{D})$ generates the Hilbert class field H_D over $\mathbf{Q}(\sqrt{D})$, we call $f(\tau)$ a *class invariant.* Class invariants have been well studied, and it is now a rather mechanical process [24, 12] to check for which D class invariants can be obtained from a given modular function f, and, in case $f(\tau)$ is a class invariant for $\mathbf{Q}(\sqrt{D})$, to find its Galois conjugates and to compute its minimal polynomial P^f_D over $\mathbf{Q}$.

If f yields class invariants, the logarithmic height of the zeroes of P^f_D will asymptotically, for $D \to -\infty$, differ from those of $P_D = P^j_D$ by some *constant factor* depending the function f. This is the factor we gain in the size of the coefficients of P^f_D when compared to P_D. For the Weber function $\mathfrak{f}$ above, we get class invariants for discriminants $D \equiv 1 \bmod 8$ not divisible by 3, and the length of the coefficients is a factor 72 smaller for $P^{\mathfrak{f}}_D$ than it is for P_D. For other discriminants, such as the discriminants congruent to 5 mod 8 from the previous sections, similar but somewhat smaller factors may be gained by using double eta-quotients $\eta(z/p)\eta(z/q)\eta(z)^{-1}\eta(z/pq)^{-1}$ as in [9].

The 'reduction factor' that is obtained when using a modular function f instead of j depends on the degree of the irreducible polynomial relation $\Psi(j, f) = 0$ that exists between j and f. In terms of the polynomial $\Psi(j, f) \in \mathbf{C}[X]$, we define the *reduction factor* of our modular function f as

$$r(f) = \frac{\deg_f(\Psi(f, j))}{\deg_j(\Psi(f, j))}.$$

By [13, Proposition B.3.5], this is, asymptotically, the *inverse* of the factor

$$\lim_{h(j(\tau))\to\infty} \frac{h(f(\tau))}{h(j(\tau))}.$$

Here h is the absolute logarithmic height, and we take the limit over all CM-points $\mathrm{SL}_2(\mathbf{Z}) \cdot \tau \in \mathbf{H}$. The reduction factor 72 obtained for the Weber function above is close to optimal in view of the following theorem.

4.1. Theorem. *The reduction factor of a modular function f satisfies*

$$r(f) \leq 800/7 \approx 114.28.$$

If Selberg's eigenvalue conjecture in [21] holds, then we have

$$r(f) \leq 96.$$

Proof. Let f be modular of level $N \geq 1$, and $\Gamma(f) \subset \mathrm{SL}_2(\mathbf{Z})$ the stabilizer of f inside $\mathrm{SL}_2(\mathbf{Z})$. Then $\Gamma(f)$ contains the principal congruence subgroup $\Gamma(N)$ of level N, and the inclusions $\Gamma(N) \cdot \{\pm 1\} \subset \Gamma(f) \subset \mathrm{SL}_2(\mathbf{Z})$ correspond to coverings

$$X(N) \longrightarrow X(f) \xrightarrow{j} \mathbf{P}^1_{\mathbf{C}}$$

of modular curves. Here $X(N)$ is the full modular curve $X(N)$ of level N, which maps to the j-line $\mathbf{P}^1_{\mathbf{C}}$ under j. This map factors via the intermediate modular curve $X(f)$, which has function field $\mathbf{C}(j, f)$. Note that the degree of the map $j : X(f) \to \mathbf{P}^1_{\mathbf{C}}$ is equal to

$$\deg_f(\Psi(f, j)) = [\mathrm{SL}_2(\mathbf{Z}) : \Gamma(f)].$$

We now consider the *gonality* $\gamma(X(f))$ of the modular curve $X(f)$, i.e., the minimal degree of a non-constant morphism $\pi : X(f) \to \mathbf{P}^1_{\mathbf{C}}$. Abramovich [1] proved in 1996 that the gonality of *any* modular curve X_H corresponding to some congruence subgroup $H \subset \mathrm{SL}_2(\mathbf{Z})$ is bounded from below by $c \cdot [\mathrm{SL}_2(\mathbf{Z}) : H]$ for some universal constant $c > 0$. His proof yields the value $c = 7/800$, and under assumption of Selberg's eigenvalue conjecture [21] the constant c can be taken equal to $1/96$.

For our curve $X(f)$, the rational map $f : X(f) \to \mathbf{P}^1_{\mathbf{C}}$ has degree

$$[\mathbf{C}(j, f) : \mathbf{C}(f)] = \deg_j(\Psi(f, j)),$$

and this degree is at least $\gamma(X(f))$. We can now use Abramovich's lower bound to obtain

$$r(f) = \frac{\deg_f(\Psi(f, j))}{\deg_j(\Psi(f, j))} \leq \frac{[\mathrm{SL}_2(\mathbf{Z}) : \Gamma_f]}{\gamma(X(f))} \leq \frac{1}{c}.$$

The proven value $c = 7/800$ and its conditional improvement $c = 1/96$ yield the two statements of our theorem. □

We do not know whether the value 96 is attained for some function f. The factor 72 of Weber's function is the best we know.

5. Numerical examples

We illustrate Algorithm 2.2 by constructing an elliptic curve having exactly

$$N = 123456789012345678901234567890123456789012345678901234568197$$

points. The integer $N \approx 10^{60}$ is prime, and the discrete logarithm problem is believed to be hard for such a curve.

We have $\log N \approx 136$ and there are 15 odd primes $p < 136$ with $\left(\frac{p^*}{N}\right) = 1$. We compute and store $\sqrt{p^*} \bmod N$ for these primes, and we try to find a discriminant $D \equiv 5 \bmod 8$ built from primes out of this 'basis' such that N splits as $N = \alpha\overline{\alpha}$ in the order $\mathcal{O}_D$ and such that $N + 1 \pm \mathrm{Trace}(\alpha)$ is prime. For $D = -41 \cdot 59 = -2419$ we find a solution

$$\begin{aligned} x &= 531376585512740287835890668303 \\ y &= 934980220808901182861 8119329 \end{aligned}$$

to the norm equation $x^2 - Dy^2 = 4N$ for which $p = N + 1 + x$ is prime.

The class group $\mathrm{Pic}(\mathcal{O}_D)$ is cyclic of order 8. The Hilbert class polynomial P_D has degree 8, and coefficients of up to 119 decimal digits. It splits completely modulo

$$p = 123456789012345678901234567890654833374525085966737125236501,$$

and any of its zeroes is the j-invariant of a curve having N points. With $a = 112507913528623610837613885503682230698868883572599681384335 \in \mathbf{F}_p$, the elliptic curve E_a given by

$$Y^2 = X^3 + aX - a$$

has N points, as may be checked by computing $N \cdot (1, 1) = 0 \in E_a(\mathbf{F}_p)$.

We can speed up the algorithm by computing a 'smaller' polynomial than the Hilbert class polynomial. We are in the case where 3 does not divide $D = -2419$, and here the cube root γ_2 of the j-function can be shown to yield class invariants. The polynomial $P^{\gamma_2}_{-2419} \in \mathbf{Z}[X]$ has coefficients up to $40 \approx 119/3$ decimal digits. For a root $x \in \mathbf{F}_p$ of $P^{\gamma_2}_{-2419} \in \mathbf{F}_p[X]$, the cube $x^3 \in \mathbf{F}_p$ is the j-invariant of a curve with N points.

The value of the double η-quotient $f = \frac{\eta(z/5)\eta(z/13)}{\eta(z)\eta(z/65)}$ at $z = \frac{-21+\sqrt{-2419}}{2}$ generates the Hilbert class field H_{-2419}. The minimal polynomial Ψ of f over $\mathbf{C}(j)$ can be computed as in [10]. It has degree 4 in j and degree 84 in X, and we have $r(f) = 84/4 = 21$. Indeed, the polynomial

$$P^f_{-2419} = X^8 + 87X^7 + 14637X^6 - 3810X^5 + 39662X^4 + 42026X^3 + 12593X^2 - 221X + 1$$

has coefficients of no more than $119/21 < 6$ digits, and its roots generate H_{-2419} over $\mathbf{Q}(\sqrt{-2419})$. This polynomial splits completely modulo p. Let $\alpha \in \mathbf{F}_p$ be a root. The polynomial $\Psi(\alpha, X) \in \mathbf{F}_p[X]$ has degree 4, and one of its roots in $\mathbf{F}_p$ is the j-invariant of a curve with N points.

If we are only interested in an elliptic curve whose group order is a prime of 60 decimal digits, we can also use the naive algorithm of trying random curves over a field $\mathbf{F}_p$ with $p \approx 10^{60}$. For $p = 10^{60} + 7 = \text{nextprime}(10^{60})$, the smallest positive integer j such that $j \bmod p$ is the j-invariant of a curve of prime order is $j = 180$.

Alternatively, we can use Algorithm 3.1 to construct an elliptic curve with CM by $\mathbf{Z}[\zeta_3]$ that has prime order of the desired size. In Step 2 we consider consecutive primes $p = 10^{60} + 99, \ldots$ congruent to 1 mod 3. The fourth prime, $p = 10^{60} + 1059$, yields the prime value

$$q = 999999999999999999999999999998130705774503095542609960125197,$$

and the elliptic curve defined by $Y^2 = X^3 + 537824$ has q rational points over $\mathbf{F}_p$.

There is some concern that elliptic curves with 'small' endomorphism ring might be less secure for cryptographic purposes. It is recommended that the class number of the endomorphism ring should be at least 200. This is no problem for our algorithms. Indeed, we can stop in Step 3b in Algorithm 2.2 only if the class number $h(D) = \deg P_D$ of the discriminant D obtained is large enough. In our example, the next higher values after $D = -2419$ for which we also find solutions to $x^2 - Dy^2 = 4N$ with $N + 1 \pm x$ prime are $D = -21003 = -3 \cdot 7001, D = -517147 = -587 \cdot 881$ and $D = -590971 = -17 \cdot 34763$. The class group of the order of discriminant -590971 is cyclic of order $228 > 200$.

The primes N needed for cryptography are rather 'small' as input for the two Algorithms in Section 2. There is no problem in feeding $N = \text{nextprime}(10^{2006}) = 10^{2006} + 2247$ to Algorithm 2.2. It yields the prime discriminant $D = -15907$ of class number 15. The corresponding class polynomial P_D has coefficients of up to 273 digits, and the desired elliptic curve is readily found. We do not print it here, as it has coefficients modulo a prime of 2006 digits that are not particularly pleasing to the human eye.

References

1. D. Abramovich, *A linear bound on the gonality of modular curves*, Internat. Math. Res. Notices **20** (1996), 1005–1011.
2. A. Agashe, K. Lauter, R. Venkatesan, *Constructing elliptic curves with a known number of points over a prime field*, High primes and misdemeanours: lectures in honour of the 60th birthday of H. C. Williams, Fields Inst. Commun., vol. 41, Amer. Math. Soc., Providence, RI, 2004, pp. 1–17.
3. D. J. Bernstein, *Proving primality in essentially quartic random time*, Math. Comp. **76** (2007), 389–403.
4. R. Bröker, *Constructing elliptic curves of prescribed order*, PhD Thesis, Universiteit Leiden, 2006.
5. R. Bröker, P. Stevenhagen, *Elliptic curves with a given number of points*, Algorithmic Number Theory Symposium VI, Springer Lecture Notes in Computer Science, vol. 3076, 2004, pp. 117–131.
6. R. Bröker, P. Stevenhagen, *Efficient CM-constructions of elliptic curves over finite fields*, Math. Comp. (to appear).

7. J.-M. Couveignes, T. Henocq, *Action of modular correspondences around CM-points*, Algorithmic Number Theory Symposium V, Springer Lecture Notes in Computer Science, vol. 2369, 2002, pp. 234–243.
8. A. Enge, F. Morain, *Comparing invariants for class fields of imaginary quadratic fields*, Algorithmic Number Theory Symposium V, Springer Lecture Notes in Computer Science, vol. 2369, 2002, pp. 252–266.
9. A. Enge, R. Schertz, *Constructing elliptic curves over finite fields from double eta-quotients*, J. Théor. Nombres Bordeaux **16** (2004), 555–568.
10. A. Enge, R. Schertz, *Modular curves of composite level*, Acta Arith. **118** (2005), 129–141.
11. A. C. P. Gee, *Class invariants by Shimura's reciprocity law*, J. Théor. Nombres Bordeaux **11** (1999), 45–72.
12. A. C. P. Gee, P. Stevenhagen, *Generating class fields using Shimura reciprocity*, Algorithmic Number Theory Symposium III, Springer Lecture Notes in Computer Science, vol. 1423, 1998, pp. 441–453.
13. M. Hindry, J. H. Silverman, *Diophantine geometry, an introduction*, Springer Graduate Texts in Mathematics, vol. 201, 2000.
14. E. Konstantinou, Y. C. Stamatiou, C. D. Zaroliagis, *On the construction of prime order elliptic curves*, Progress in cryptology—INDOCRYPT 2003, Springer Lecture Notes in Computer Science 2904, 2003, pp. 309–322.
15. K. Koyama, Y. Tsuruoka, N. Kunihiro, *Modulus Search for Elliptic Curve Cryptosystems*, Advances in Cryptology - ASIACRYPT '99, Lecture Notes in Computer Science, vol. 1716, 1999, pp. 1–7.
16. G.-J. Lay, H. G. Zimmer, *Constructing elliptic curves with given group order over large finite fields*, Algorithmic Number Theory Symposium I, Springer Lecture Notes in Computer Science, 1994.
17. H. W. Lenstra, Jr., *Factoring integers with elliptic curves*, Ann. of Math. (2) **126** (1987), 649–673.
18. F. Morain, *Implementing the asymptotically fast version of the elliptic curve primality proving algorithm*, Math. Comp. **76** (2007), 493–505.
19. C. Pomerance, *Smooth numbers and the quadratic sieve*, Surveys in Algorithmic Number Theory, Cambridge University Press, 2006.
20. H.-G. Rück, *On the discrete logarithm problem in the divisor class group of curves*, Math. Comp. **68** (1999), 805–806.
21. P. Sarnak, *Selberg's eigenvalue conjecture*, Notices of the AMS, vol. 42, 1995, pp. 1272–1277.
22. E. Savaş, T. A. Schmidt, Ç. K. Koç, *Generating elliptic curves of prime order*, Cryptographic hardware and embedded systems—CHES 2001 (Paris), Springer Lecture Notes in Computer Science 2162, 2001, pp. 142–158.
23. R. Schoof, *Counting points on elliptic curves over finite fields*, J. Théorie des Nombres de Bordeaux **7** (1995), 219–254.
24. P. Stevenhagen, *Hilbert's 12th problem, complex multiplication and Shimura reciprocity*, Class field theory – its centenary and prospect, ed. K. Miyake, Adv. studies in pure math., vol. 30, 2001, pp. 161–176.
25. H. Weber, *Lehrbuch der Algebra*, vol. 3, Chelsea Publishing Company (reprint), original edition 1908.

University of Calgary, Department of Mathematics and Statistics, 2500 University Drive NW, Calgary, AB T2N 1N4, Canada

Mathematisch Instituut, Universiteit Leiden, Postbus 9512, 2300 RA Leiden, The Netherlands

E-mail address: reinier@math.ucalgary.ca, psh@math.leidenuniv.nl