Volume 00, Number 0, Pages 000–000
S (XX)0000-0

CONSTRUCTING ELLIPTIC CURVES OVER FINITE FIELDS WITH PRESCRIBED TORSION

ANDREW V. SUTHERLAND

Abstract. The modular curve $X_1(N)$ parametrizes elliptic curves with a point of order N. For $N \leq 50$ we obtain plane models of $X_1(N)$ that have been optimized for fast computation, and provide explicit birational maps to transform a point on our model of $X_1(N)$ to an elliptic curve. Over a finite field, these allow us to quickly construct elliptic curves containing a point of order N, and can accelerate the search for an elliptic curve whose order is divisible by N.

1. Introduction

By Mazur's theorem [12], the order of a nontrivial torsion point on an elliptic curve over the rational numbers must belong to the set

$$\mathcal{T} = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 12\}.$$

Conversely, for each $N \in \mathcal{T}$, an infinite family of elliptic curves over $\mathbb{Q}$ containing a point of order N is exhibited by the parametrizations of Kubert [11].[1] Over a finite field $\mathbb{F}_q$, these parametrizations provide an easy way to generate universal families of curves whose order is divisible by N. This can accelerate applications that search for an elliptic curve with a particular property, such as a curves with smooth order (as in the elliptic curve factorization method [1]), or curves with a particular endomorphism ring (as when computing Hilbert class polynomials with the Chinese Remainder Theorem [2]).

To generate an elliptic curve $E/\mathbb{F}_q$ with non-trivial 7-torsion, for example, one applies [11]. Pick $r \in \mathbb{F}_q$, then use $b = r^3 - r^2$ and $c = r^2 - r$ to define

$$E(b, c): \quad y^2 + (1 - c)xy - by = x^3 - bx^2. \tag{1}$$

Provided $E(b, c)$ is nonsingular, we obtain an elliptic curve on which the point $P = (0, 0)$ has order 7. By contrast, obtaining such a curve by trial and error is far more time consuming: testing for 7-torsion typically involves finding the roots of a degree 24 polynomial (the 7-division polynomial), and several curves may need to be tested (approximately six on average) .

Mazur's theorem limits us to $N \in \mathcal{T}$, but we can proceed further if we do not restrict ourselves to curves defined over $\mathbb{Q}$. Reichert treats $N \in \{11, 13, 14, 15, 16, 18\}$ over quadratic extensions of $\mathbb{Q}$ using $X_1(N)$, the modular curve that parametrizes elliptic curves with a point of order N [15]. We may be able to realize a curve defined over $K = \mathbb{Q}[\sqrt{d}]$ in $\mathbb{F}_q$, but only if d is a quadratic residue in $\mathbb{F}_q$.

2000 *Mathematics Subject Classification.* Primary 14H52; Secondary 11G20.

[1]Kubert also addresses the torsion subgroups $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2N\mathbb{Z}$ for $N = 1, 2, 3, 4$. We consider only the subgroups $\mathbb{Z}/N\mathbb{Z}$ here.

Alternatively, we can use a point on $X_1(N)/\mathbb{F}_q$ to directly construct $E(b,c)/\mathbb{F}_q$ containing a point of order N. This applies in any sufficiently large finite field, for all $N > 3$ (see [11] for $N \le 3$). For $N \in \mathcal{T}$ the curve $X_1(N)$ has genus 0 and we simply obtain the Kubert parametrizations, but in general we construct $E(b,c)$ from a point (x,y) on $X_1(N)$ via a birational map that depends on the defining equation we choose for $X_1(N)$.

For example, to obtain a curve with non-trivial 13-torsion, we use a point on

$$X_1(13): \quad y^2 + (x^3 + x^2 + 1)y - x^2 - x = 0,$$

which may be obtained by choosing $x \in \mathbb{F}_q$ at random and attempting to solve the resulting quadratic equation for y in $\mathbb{F}_q$.[2] We then apply the transformation

$$\begin{aligned} r &= 1 - xy, \\ s &= 1 - xy/(y+1), \end{aligned}$$

set $c = s(r-1)$ and $b = cr$, and construct $E(b,c)$. If we obtain a singular curve (or if $y = -1$) we try again with a different point on $X_1(13)$ (this rarely happens).

To apply this method we require a defining equation for $X_1(N)/\mathbb{F}_q$ along with a suitable birational map. For fast computation we seek a plane model $f(x,y) = 0$ that minimizes the degree d of one of its variables. For $N \le 18$ one can derive these from the results of Reichert (and Kubert). Reichert's method can be applied to $N > 18$, but the "raw" form of $X_1(N)$ initially obtained is quite large and of higher degree than necessary. More compact defining equations for $X_1(N)$ are given by Yang [21] for $N \le 22$, but these do not minimize d. The minimal value $d = d(N)$ is a topic of some interest [6, 7, 9, 13, 14], since we can construct (infinitely many) elliptic curves containing a point of order N over number fields of degree d. For $N > 18$, few values of $d(N)$ are known (see sequence A146879 in the OEIS [17]).

Given a plane model for $X_1(N)$, we may attempt to reduce its complexity (degree, number of terms, and coefficient size) through a judiciously chosen sequence of rational transformations. This procedure is somewhat *ad hoc*, however, and finding an optimal (or even good) sequence becomes difficult for larger N. We treat this as a combinatorial optimization problem, applying standard search techniques to obtain a solution that is locally optimal under a relation we define. We cannot claim that the results are globally optimal, but they do yield an upper bound on $d(n)$. For $N \le 22$ we are able to match known lower bounds for $d(N)$ [6, 7, 9], including $d(19) = 5$, which we believe to be new.[3] Results for $N \le 30$ are listed in the appendix, and are available in electronic form for $N \le 50$.

For odd N we also show how to efficiently generate $E/\mathbb{F}_q$ with a point of order $4N$, or satisfying $\#E \equiv 2N \bmod 4N$, using our results for $X_1(2N)$.

2. Computing the raw form of $X_1(N)$

Following Reichert [15], we summarize the method to obtain a plane model for $X_1(N)$ in the form $F(r,s) = 0$.[4] The equation $E(b,c)$ in (1) is the Tate normal form of an elliptic curve (called a Kubert curve in [1]). Any elliptic curve containing

---

[2] When $X_1(N)$ has genus 1 ($N = 11, 14, 15$) we may obtain additional points more efficiently using the group operation on $X_1(N)$ (see Section 4).

[3] But we do not achieve $d(24) = 4$ implied by [6].

[4] Reichert uses auxiliary variables $m = s(1-r)/(1-s)$ and $t = (r-s)/(1-s)$. We find it preferable to work directly with r and s.

a point of order greater than 3 can be put in this form (see V.5 of [10]). The discriminant of $E(b,c)$ is

$$\Delta(b,c) = b^3(16b^2 - 8bc^2 - 20bc + b + c(c-1)^3). \tag{2}$$

To ensure $E(b,c)$ is nonsingular we require $\Delta(b,c) \neq 0$, so assume $b \neq 0$. Applying the group law for elliptic curves [16, III.2.3], we double the point $P = (0,0)$ to obtain $2P = (b, bc)$, and for $n > 1$ compute the point $(n+1)P = (x_{n+1}, y_{n+1})$ in terms of $nP = (x_n, y_n)$ using

$$x_{n+1} = by_n/x_n^2, \quad y_{n+1} = b^2(x_n^2 - y_n)/x_n^3. \tag{3}$$

We find that the inverse of $nP = (x_n, y_n)$ is

$$-nP = (x_n, b + (c-1)x_n - y_n). \tag{4}$$

If P is an N-torsion point and $m + n = N$, then we must have $mP = -nP$. If $m \neq n$ this implies $x_m = x_n$, and if $m = n$ we have $2y_n = b + (c-1)x_n$. For $b \neq 0$ this requires $N > 3$. When $x_m = x_n$ either $mP = nP$ or $mP = -nP$, and in the latter case P is an N-torsion point. If we choose $m = \lceil \frac{N+1}{2} \rceil$ and $n = \lfloor \frac{N-1}{2} \rfloor$ we ensure that $mP \neq nP$, obtaining a necessary and sufficient condition for N-torsion:

$$NP = 0_E \iff x_m = x_n, \tag{5}$$

valid for $N > 3$. The first three multiples of P are:

$$\begin{aligned} P &= (0,0), \\ 2P &= (b, bc), \\ 3P &= (c, b-c). \end{aligned}$$

We see immediately that P is a point of order 4 exactly when $c = 0$, and P is a point of order 5 exactly when $b = c$. For $N > 5$ define:

$$\begin{aligned} r &= b/c, & b &= rs(r-1), \\ s &= c^2/(b-c), & c &= s(r-1), \end{aligned}$$

and note that $r \notin \{0,1\}$, and $s \neq 0$.

We now apply (3) to compute x_n in terms of r and s. Values for $n \leq 10$ are listed in Table 1. To obtain the raw form of $X_1(N)$, we start with the equation $x_m = x_n$ from (5), then clear denominators and subtract to obtain an equation of the form $F^*(r,s) = 0$, where the polynomial $F^*(r,s)$ has integer coefficients. We then remove from F^* any factors prohibited by our assumptions (namely r, $r-1$, and s), and also factors corresponding to M-torsion for any $M > 5$ dividing N (such as $s - 1$ for $M = 6$ and $r - s$ for $M = 7$).[5]

Let $F(r,s)$ denote the polynomial that remains. We claim that $F(r,s) = 0$ is a defining equation for $X_1(N)$. By construction, any solution to $F(r,s) = 0$ will produce a curve $E(b,c)$, with $c = s(r-1)$ and $b = rc$, on which P is a point of order N, provided that $\Delta(b,c) \neq 0$. Conversely, any curve $E(b,c)$ on which P has order $N > 5$ yields a solution $r = b/c$, $s = c^2/(b-c)$ to $F(r,s) = 0$. These statements hold for any field K, provided that we verify $\Delta(b,c) \neq 0$ in K.

[5] More generally, these can be recognized by computing the raw form of each $X_1(M)$. In practice $F(r,s)$ is simply the largest irreducible factor of $F^*(r,s)$.

$x_1 = 0$
$x_2 = rs(r-1)$
$x_3 = s(r-1)$
$x_4 = r(r-1)$
$x_5 = rs(s-1)$
$x_6 = s(r-1)(r-s)\ /\ (s-1)^2$
$x_7 = rs(r-1)(s-1)(rs-2r+1)\ /\ (r-s)^2$
$x_8 = r(r-1)(r-s)(r-s^2+s-1)\ /\ (rs-2r+1)^2$
$x_9 = s(r-1)(rs-2r+1)(rs^2-3rs+r+s^2)\ /\ (r-s^2+s-1)^2$
$x_{10} = rs(r-s^2+s-1)(r^2-rs^3+3rs^2-4rs+s)\ /\ (rs^2-3rs+r+s^2)^2$

TABLE 1. x-coordinates of nP for $n \leq 10$.

When $N = 16$, for example, putting $x_9 = x_7$ in the form $F^*(r,s) = 0$ yields

$$\begin{aligned} F^*(r,s) =& s(r-1)(r-s)^2(rs-2r+1)(rs^2-3rs+r+s+s^2) \\ & - rs(r-1)(s-1)(rs-2r+1)(r-s^2+s-1)^2. \end{aligned}$$

The nonzero factors s and $r-1$ may be removed, and also the factor $rs-2r+1$, which can be zero only when P has order 8. Thus we obtain

$$F(r,s) = (r-s)^2(rs^2-3rs+r+s+s^2) - r(s-1)(r-s^2+s-1)^2.$$

When expanded, this yields the entry for $N = 16$ in Table 4. The polynomials $F(r,s)$ for N up to 50 are available in electronic form from the author. The largest of these has 1,791 terms and maximum coefficient on the order of 10^{19}.

3. Reducing the complexity of $F(r,s) = 0$

To facilitate fast computation we wish to simplify the raw from of $X_1(N)$. We seek a birationally equivalent curve $f(x,y) = 0$ that minimizes the degree of one of its variables (say y). Subject to this constraint, we would like to make f monic in y and also to minimize the degree in x, the number of terms, and the size of the coefficients (roughly in that order of priority). One typically approaches this problem by attempting to remove singularities from $F(r,s)$ through a combination of translations and inversions (see [15] for examples). We take a more naïve approach that allows us to easily automate the process.

There are three basic types of transformations we will use:

(1) Translate:	$x \rightsquigarrow x + a$	or	$y \rightsquigarrow y + a$.
(2) Invert:	$x \rightsquigarrow 1/x$	or	$y \rightsquigarrow 1/y$.
(3) Separate:	$x \rightsquigarrow 1/x,\ \ y \rightsquigarrow y/x$	or	$x \rightsquigarrow x/y,\ \ y \rightsquigarrow 1/y$.

These are clearly all invertible operations. The third type combines an inversion and a division, but we find it works well as an atomic unit. In order to bound the number of atomic operations, we let $a \in \{\pm 1\}$, giving a total of eight.

Consider the directed graph G on the set $\mathcal{C}$ of plane curves that can be obtained from $F(r,s) = 0$ by applying a finite sequence of the transformations above, with edges labeled by the corresponding operation. A path in G defines a birational map (the composition of the operations labeling its edges), and any path can be reversed to yield the inverse map. Starting from the curve C_0 defined by $F(r,s) = 0$, we want to find a path to a "better" curve C. To make this precise, we associate to each integer polynomial $f(x,y)$ a vector of nonnegative integers

$$v(f) = (d_y, m_y, d_x, d_{\mathrm{tot}}, t, S),$$

whose components are defined by:

- d_y is the degree of f in y;
- m_y is 0 if no term of f is a multiple of xy^{d_y} and 1 otherwise;
- d_{tot} is the total degree of f;
- t is the number of terms in f;
- S is the sum of the absolute values of the coefficients of f.

The component m_y will be zero exactly when f can be made monic as a polynomial in y. We order the vectors $v(f)$ lexicographically, and to each $C \in \mathcal{C}$ assign the vector $v(C) = \min\{v(f(x,y)), v(f(y,x))\}$, where $f(x,y) = 0$ defines C. We compare curves by comparing their vectors, obtaining a prewellordering of $\mathcal{C}$. In particular, any subset of $\mathcal{C}$ contains a (not necessarily unique) minimal element.

We now give a simple algorithm to search the graph G for a curve that is locally optimal within a radius R. We use $N(C,k)$ to denote the set of curves connected to C by a path of length at most k in G. For $C' \in N(C,k)$ we let $\phi(C',C)$ denote the birational map defined by the path from C' back to C.

1. Set $C \leftarrow C_0$, $k = 1$, and let φ be the identity map.
2. While $k \leq R$:
   a. Determine a minimal element C' of $N(C,k)$.
   b. If $v(C') < v(C)$, then set $\varphi \leftarrow \varphi \circ \phi(C',C)$, $C \leftarrow C'$, and $k \leftarrow 0$.
   c. Set $k \leftarrow k+1$.
3. Output $C_1 = C$ and φ.

The curve C_1 output by the algorithm is our optimized plane model for $X_1(N)$. It is birationally equivalent to the curve C_0 defined by $F(r,s) = 0$, and the map φ carries points on C_1 to points on C_0.

To enumerate the neighbors of the curve C defined by $f(x,y) = 0$, the algorithm applies each of the eight atomic operations. The result of applying the birational map ϕ with inverse $\tilde{\phi}$ is computed by expanding $f(\tilde{\phi}_x(x,y), \tilde{\phi}_y(x,y))$ as a formal substitution of variables and clearing any denominators that result. Thus the translation $x \rightsquigarrow x-1$ is obtained by expanding $f(x+1,y)$, and the inversion $x \rightsquigarrow 1/x$ effectively replaces x^i in $f(x,y)$ with $x^{d_x - i}$. To enumerate $N(C,k)$ involves applying up to 8^k possible sequences of operations (this number can be reduced by eliminating obviously redundant sequences), so the bound R cannot be very large. We have tested up to $R = 10$, but find that $R = 8$ suffices to obtain the results given here. When $R = 8$ the algorithm takes less than an hour (on a 2.8 GHz AMD Athlon processor) for $N \leq 50$.

Table 2 illustrates the algorithm's execution for $N = 16$. We begin with the curve C_0 defined by $F(r,s) = 0$, as listed in Table 4, and set $C = C_0$ with $f(x,y) = F(x,y)$. The algorithm finds $v(C) = v(f(y,x)) = (3,1,5,6,13,40)$, indicating that the $f(x,y)$ has degree 3 in x (in this case $v(f(y,x))$ is less than $v(f(x,y))$ so the roles of x and y are reversed). Additionally, $f(x,y)$ is not monic in x, has degree 5 in y, total degree 6, 13 terms, and the absolute values of its coefficients sum to 40.

No curves within a distance $k = 1$ are found that improve $v(C)$, but for $k = 2$ a curve C' is found that is monic in x (and also degree 3), which implies $v(C') < v(C)$.

steps	$C: f(x,y)=0$	$v(C)$
-	$x^3y^2 - 4x^3y + 2x^3 + 3x^2y^2 + 2x^2y - 2x^2 - xy^5 + 4xy^4$ $-\ 10xy^3 + 6xy^2 - 3xy + x + y^4$	(3,1,5,6,13,40)
5,8	$x^3 + x^2y^5 - 3x^2y^4 + 6x^2y^3 - 10x^2y^2 + 4x^2y - x^2 - 2xy^6$ $+\ 2xy^5 + 3xy^4 + 2y^7 - 4y^6 + y^5$	(3,0,7,7,13,40)
1,3,8,6	$x^3 + x^2y^4 + 2x^2y^3 + 4x^2y^2 - 5x^2 - 2xy^4 - 8xy^3 - 13xy^2$ $+\ 8x + 2y^4 + 8y^3 + 10y^2 - 4$	(3,0,4,6,13,68)
1	$x^3 + x^2y^4 + 2x^2y^3 + 4x^2y^2 - 2x^2 - 4xy^3 - 5xy^2 + x$ $+\ y^4 + 2y^3 + y^2$	(3,0,4,6,11,24)
1	$x^3 + x^2y^4 + 2x^2y^3 + 4x^2y^2 + x^2 + 2xy^4 + 3xy^2 + 2y^4$	(3,0,4,6,8,16)
5,6	$2x^3 + 3x^2y^2 + 2x^2 + xy^4 + 4xy^2 + 2xy + x + y^4$	(3,0,4,5,8)
2,4,5,6,8	$-x^3 + x^2y^3 - 4x^2y^2 + 4x^2y + 2x^2 + 3xy^2 - 6xy - x + 2y$	(3,0,3,5,9,24)
3	$-x^3 + x^2y^3 - x^2y^2 - x^2y + 3x^2 + 3xy^2 - 4x + 2y + 2$	(3,3,0,5,9,18)
4,5,1,7	$-x^2y^2 - 2x^2y - x^2 + xy^3 + 2xy^2 + y^3 + 3y^2 + 2y$	(2,1,3,4,8,13)
4	$-x^2y^2 + xy^3 - xy^2 - xy + x + y^3 - y$	(2,1,3,4,7,7)
8	$-x^2 + xy^3 - xy^2 - xy + x - y^3 + y$	(2,0,3,4,7,7)
1	$-x^2 + xy^3 - xy^2 - xy - x - y^2$	(2,0,3,4,6,6)

TABLE 2. Optimization of $X_1(16)$.

$$1: x \rightsquigarrow x-1, \qquad 2: x \rightsquigarrow x+1, \qquad 3: y \rightsquigarrow y-1, \qquad 4: y \rightsquigarrow y+1$$
$$5: x \rightsquigarrow 1/x, \quad 6: y \rightsquigarrow 1/y, \quad 7: x \rightsquigarrow 1/x,\ y \rightsquigarrow y/x, \quad 8: x \rightsquigarrow x/y,\ y \rightsquigarrow 1/y.$$

C' is a minimal curve in $N(C,2)$, so C is replaced by C' and the map φ becomes

$$x \rightsquigarrow y/x, \quad y \rightsquigarrow 1/y.$$

This reverses the sequence of steps 5,8 (as identified in the key to Table 2) used to reach C' from C_0 (so φ maps points on C' back to points on C_0). The next improvement occurs when $k = 4$. In this case reversing the path 1,3,8,6 from C to C' yields the sequence 6,8,4,2, and φ becomes

$$x \rightsquigarrow (y+1)/(xy+1), \quad y \rightsquigarrow 1/(y+1).$$

The algorithm continues in this fashion, finding the sequence of curves listed in Table 2, until it is unable to find a better curve within the maximum search radius R. The resulting curve has minimal degree in x rather than y, so we swap variables (and adjust signs) to obtain the optimized curve

$$X_1(16): \quad y^2 + (x^3 + x^2 - x + 1)y + x^2 = 0, \tag{6}$$

which appears in Table 6. Corresponding changes to φ yield the birational map

$$r = 1 + (y+1)/(xy+y^2), \qquad s = 1 + (y+1)/(xy-y^2), \tag{7}$$

listed in Table 7, which carries points on the curve in (6) to points on C_0.

Table 5 shows the improvement in the minimal degree $d(C_i)$ and the number of terms $t(C_i)$ obtained when the initial curve C_0 is transformed to the locally optimal curve C_1 output by the algorithm. For comparison, we also list the genus of $X_1(N)$, obtained from sequence $A029937$ in the OEIS [17] (see Theorem 1.1 of [8] for a general formula).

The search procedure described above can be applied to any plane curve defined over $\mathbb{Q}$, but its effectiveness depends largely on finding singularities with small integer coordinates. Empirically, this works well with $X_1(N)$, but other applications may wish to modify the list of atomic operations to incorporate more general translations. Alternative search strategies, such as simulated annealing, may also be worth investigating.

4. Application to finite fields

We can use the optimized form of $X_1(N)$ to efficiently generate elliptic curves containing a point of order N over the finite field $\mathbb{F}_q$, as described in the introduction. Here we briefly address a few topics relevant to practical implementation. We assume that C_1 is defined by $f(x, y) = 0$, with $d_y \leq d_x$, and consider how we may use $f(x, y)$ to efficiently generate a set of m elliptic curves over $\mathbb{F}_q$, each containing a point of order N.

Except for a small set of points (those leading to singular curves and those for which φ is undefined), there is a one-to-one correspondence between points on C_1 and nonsingular curves in Tate normal form on which the point $P = (0, 0)$ has order N (see Section 2). For large q, each possible j-invariant in $\mathbb{F}_q$ (and each twist) is represented by an approximately equal number of curves in Tate normal form. It follows that we can obtain a (nearly) uniform distribution over isomorphism classes of elliptic curves defined over $\mathbb{F}_q$ containing a point of order N, provided that we have a uniformly distributed sample of points on $f(x, y) = 0$.

When $d > 2$ it is not a trivial task to efficiently generate a sample with uniform distribution. It is impractical to test random solutions to $f(x, y) = 0$, so instead we pick $x_i \in \mathbb{F}_q$ at random and compute the roots y_{ij} (if any) of the degree d polynomial $h_i(y) = f(x_i, y)$ over $\mathbb{F}_q$. For each root y_{ij} of h_i we include the point (x_i, y_{ij}) in our set of m points. Assuming $m \gg d$ this gives us an approximately uniform distribution (if we used only one root of h_i this would *not* be true), but the points obtained are not all independent. In practice this does not pose a problem. At most d points share a common x value, and after mapping the points back to $F(r, s) = 0$ and constructing $E(b, c)$ it is very difficult to discern any relationship among the curves.[6] With this approach we expect to compute the roots of m polynomials $h_i(y)$, on average, in order to obtain m points on $f(x, y) = 0$.

When $X_1(N)$ has genus 1, the curve $f(x, y) = 0$ is an elliptic curve, and we may use a more efficient approach: select one point at random, then compute multiples of it via the group operation. We can generate m random multiples using $O(\log q + m \log q / \log \log q)$ group operations via standard multi-exponentiation techniques [22], or we can compute multiples in an arithmetic sequence using just $m + O(\log q)$ group operations. The latter approach does not generate independent points, but it is highly efficient: only $O(1)$ operations in $\mathbb{F}_p$ are required per point (assuming $m \gg \log q$). During this computation it is convenient to work with a model for $X_1(N)$ in short Weierstrass form. These are provided in Table 3, along with the corresponding maps back to $F(r, s) = 0$.

Having generated a set of m points on $f(x, y) = 0$, we apply the appropriate birational map to obtain points on $F(r, s) = 0$. When doing so, we invert the

[6] Alternatively, we could obtain a uniform independent distribution using as most one root of each h_i, provided we discard it with a certain probability depending on the number of roots h_i has. We do not regard this as a practical solution.

N	$X_1(N)$
11	$y^2 = x^3 - 432x + 8208$ $r = (y + 108)/216$ $s = 1 + (y - 108)/(6x + 72)$
14	$y^2 = x^3 - 675x + 13662$ $r = 1 + (108x - 36y + 3564)/(3x^2 - xy - 342x + 75y + 999)$ $s = (6x - 234)/(9x - y - 135)$
15	$y^2 = x^3 - 27x + 8694$ $t = (6x - 90)(18x + 6y - 918)$ $r = 1 - t/(x^2y - 189x^2 + 42xy - 4050x - 3y^2 + 441y - 1701)$ $s = 1 - t/(x^2y - 81x^2 + 6xy - 3402x - 3y^2 + 981y - 35721)$

Table 3. Short Weierstrass form of $X_1(N)$ with genus 1.

denominators in parallel, via the usual Montgomery trick [3, Alg. 11.15]. We then compute (b, c) pairs (using $c = s(r - 1)$ and $b = rc$). In a field of characteristic not 2 or 3, we may convert the curve $E(b, c)$ to the short Weierstrass form:

$$y^2 = x^3 + Ax + B. \tag{8}$$

Let $d = c - 1$ and $e = d^2 - 4b$. Through the admissible change of variables

$$x = 36x' - 3e, \qquad y = 216y' + 108(dx' + b), \tag{9}$$

we find that

$$A = 27(24bd - e^2), \qquad B = 54(e^3 - 36bde + 216b^2),$$

and $(3d, -108b)$ is a point of order N on $y^2 = x^3 + Ax + B$.[7]

At some point during the process described above, we need to check that the discriminant Δ of each curve obtained is nonzero. This is most efficiently done at the end using $\Delta = -4A^3 - 27B^2$. This may result in fewer than m curves being generated, but we can always obtain more points on $X_1(N)$ if necessary.

5. Prescribing 4-torsion

For odd N, we can use $X_1(2N)$ to generate elliptic curves which contain a point of order $4N$ over $\mathbb{F}_q$ in a manner that may be more efficient than using $X_1(4N)$. Alternatively, we can generate curves which contain a point of order $2N$ but do *not* contain a point of order $4N$. These results rely on efficiently computing the 4-torsion of an elliptic curve using a known a point of order 2, which we obtain from the point $P = (0, 0)$ of order $2N$. For odd N, a curve with a point of order N has a point of order $4N$ if and only if it has a point of order 4.

In fact, we only need the x-coordinate of NP, which can be computed as described in Section 2 (see Table 1 for $N \leq 10$). It will be convenient to work with the short Weierstrass form (8), so we assume that the point NP has been translated via (9) to the 2-torsion point $\beta = (x_0, 0)$ on the curve E defined by $y^2 = f(x) = x^3 + Ax + B$.

[7]For $N \in \mathcal{T}$, parametrizations which additionally provide a point with infinite order over $\mathbb{Q}$ are considered by Atkin and Morain in [1].

Our strategy is to use the value x_0 to determine whether E contains a point of order 4 or not. In the best case this requires only a single test for quadratic residuacity in $\mathbb{F}_q$, and even in the worst case, a square root and two tests for quadratic residuacity suffice. If the result is not as desired, we discard E and test another curve with a point of order $2N$. On average we expect to test two curves. This is typically faster than either finding a point on $X_1(4N)$, or using $X_1(N)$ and computing 4-torsion without a known point of order 2.

Lemma 1. *If $\alpha = (u, v)$ and $\beta = (x_0, 0)$ are points on a nonsingular elliptic curve E defined by $y^2 = f(x) = x^3 + Ax + B$ over a field of characteristic not 2 then*

$$2\alpha = \beta \qquad \Longleftrightarrow \qquad (u - x_0)^2 = f'(x_0),$$

where $f'(x) = 3x^2 + A$.

Proof. If $2\alpha = \beta$ then the duplication formula for elliptic curves [16, p. 59] implies

$$x_0 = \frac{u^4 - 2Au^2 - 8Bu + A^2}{4(u^3 + Au + B)}.$$

Therefore u must satisfy

$$u^4 - 4x_0u^3 - 2Au^2 - (4Ax_0 + 8B)u - 4Bx_0 + A^2 = 0.$$

Since $\beta = (x_0, 0) \in E$, we have $x_0^3 + Ax_0 + B = 0$. Substituting for B yields

$$u^4 - 4x_0u^3 - 2Au^2 + (8x_0^3 + 4Ax_0)u + 4x_0^4 + 4Ax_0^2 + A^2.$$

We now set $u = z + x_0$ and rewrite this as

$$(z^2 - (3x_0^2 + A))^2 = 0.$$

Therefore

$$(u - x_0)^2 = 3x_0^2 + A = f'(x_0),$$

as desired. Reversing the argument yields the converse, provided $f(u) \neq 0$. But if u is a root of f, then one can show that $(u - x_0)^2 = f'(x_0)$ implies $D(f) = 0$, contradicting the fact that E is nonsingular. $\square$

There may be 1 or 3 points of order 2 on E. The x-coordinates of the other two (if they exist) are the roots x_1 and x_2 of $f(x)/(x - x_0)$, which we can determine with the quadratic formula. We now give our main result for treating 4-torsion.

Proposition 1. *Let $(x_0, 0)$ be a point of order 2 on a nonsingular elliptic curve E defined by $y^2 = f(x) = x^3 + Ax + B$ over the field $\mathbb{F}_q$, with quadratic character χ. Let n be the number of roots of $f(x)$ in $\mathbb{F}_q$, and for $n = 3$, let x_1 and x_2 denote the other two roots.*

For $q \equiv 3 \bmod 4$:

1. *If $\chi(f'(x_0)) = 1$ then E has a point of order 4.*
2. *Otherwise, E has a point of order 4 if and only if $n = 3$ and $\chi(f'(x_1)) = 1$.*

For $q \equiv 1 \bmod 4$:

1. *If $n = 1$ then E has a point of order 4 if and only if $\chi(f'(x_0)) = 1$.*
2. *Otherwise, if $\chi(f'(x_0)) = 1$ (resp., $\chi(f'(x_0)) = -1$) then E has a point of order 4 if and only if $\chi(x_0 - x_1) = 1$ (resp., $\chi(x_1 - x_2) = 1$).*

Proof. Note that $f(x_i) = 0$ implies $f'(x_i) \neq 0$, since E is nonsingular, hence $\chi(f'(x_i)) = \pm 1$. Let $\tilde{E}$ denote the quadratic twist of E over $\mathbb{F}_q$. By Lemma 1, each root x_i of $f(x)$ for which $\chi(f'(x_i)) = 1$ yields 4 points of order 4 (two pairs of inverses), either all on E, all on $\tilde{E}$, or split 2-2 between them. Recall that $\#E = q + 1 - t$ and $\#\tilde{E} = q + 1 + t$, where t is the trace of Frobenius, so $4|\#E$ if and only if $4|\#\tilde{E}$, and for $q \equiv 3 \bmod 4$, $8|\#E$ if and only if $8|\#\tilde{E}$.

We first consider $q \equiv 3 \bmod 4$.

Suppose $\chi(f'(x_0)) = 1$. If $n = 1$ then E and $\tilde{E}$ each have 2 points of order 4. If $n = 3$ then at least one of E and $\tilde{E}$ has order 8, but if one does, then so must the other, and again E has a point of order 4.

Suppose $\chi(f'(x_0)) = -1$. If $n = 1$ then E cannot have a point of order 4, so assume $n = 3$. By Lemma 2, for $q \equiv 3 \bmod 4$ we have $\chi(f'(x_1)) = \chi(f'(x_2))$, and if their common value is -1 then E cannot have a point of order 4. If it is 1 then at least one of $\#E$ or $\#\tilde{E}$ is divisible by 8, but then they both are and both contain a point of order 4.

We now consider $q \equiv 1 \bmod 4$.

If $n = 1$ then E can have a point of order 4 if and only if $\chi(f'(x_0)) = 1$, as above. Now assume $n = 3$. It follows from Theorem 4.2 of [10] that E has a point of order 4 if and only if at least two of $x_0 - x_1$, $x_1 - x_2$, and $x_2 - x_0$ are squares in $\mathbb{F}_q$. We have $f'(x_0) = (x_0 - x_1)(x_0 - x_2)$, so if $\chi(f'(x_0)) = 1$ then it suffices to check $\chi(x_0 - x_1)$, and if $\chi(f'(x_0)) = -1$ then it suffices to check $\chi(x_1 - x_2)$. □

Lemma 2. *Let $f(x)$ be a monic cubic polynomial with distinct roots x_0,x_1,x_2 in $\mathbb{F}_q$, with q odd. We have*

$$\chi(-1)\chi(f'(x_0))\chi(f'(x_2))\chi(f'(x_2)) = 1.$$

In particular, the number of squares in the set $\{f'(x_0), f'(x_1), f'(x_2)\}$ is even when $q \equiv 1 \bmod 4$ and odd when $q \equiv 3 \bmod 4$.

Proof. Recall that for a monic f of degree $n = 3$, the discriminant of f is given by

$$D(f) = (-1)^{n(n-1)/2} R(f, f') = -R(f, f'),$$

where $R(f, f')$ is the resultant. Since f is monic, we have $R(f, f') \prod f'(x_i)$, thus

$$D(f) = -f'(x_0)f'(x_1)f'(x_2).$$

The roots of f are distinct, so $D(f) \neq 0$. By the Stickelberger-Swan Theorem (Corollary 1 in [19]), $D(f)$ must be a square in $\mathbb{F}_q$, since f is degree 3 and has 3 irreducible factors. The lemma then follows, since $\chi(D(f)) = 1$. □

As a final remark, we note that when (1) fails to hold in Proposition 1, it is quite likely that E has trivial 4-torsion. On average, this probability is about 90% (this can be computed precisely, see [4, 5]). As a practical optimization, when seeking a point of order $4N$, if condition (1) fails we may simply discard the curve and test another. When $q \equiv 3 \bmod 4$ this reduces to a test for quadratic residuacity in $\mathbb{F}_q$, and we expect two tests of curves generated with $X_1(2N)$ will suffice to produce a curve with a point of order $4N$.

References

1. A.O.L. Atkin and F. Morain, *Finding suitable curves for the elliptic curve method of factorization*, Mathematics of Computation **60** (1993), 399–405.
2. Juliana Belding, Reinier Bröker, Andreas Enge, and Kristin Lauter, *Computing Hilbert class polynomials*, Algorithmic Number Theory Symposium–ANTS VIII, Lecture Notes in Computer Science, vol. 5011, Springer, 2008, pp. 282–295.
3. Henri Cohen (Ed.) et al., *Handbook of elliptic and hyperelliptic curve cryptography*, Chapman and Hall, 2006.
4. Ernst-Ulrich Gekeler, *The distribution of group structures on elliptic curves over finite prime fields*, Documenta Mathematica **11** (2006), 119–142.
5. Everett W. Howe, *On the group orders of elliptic curves over finite fields*, Compisitio Mathematica **85** (1993), 229–247.
6. Daeyeol Jeon, Chang Heon Kim, and Euisung Park, *On the torsion of elliptic curves over quartic number fields*, Journal of the London Mathematical Society **74** (2006), 1–12.
7. Daeyeol Jeon, Chang Heon Kim, and Andreas Schweizer, *On the torsion of elliptic curves over cubic number fields*, Acta Arithmetica **113** (2004), 291–301.
8. Daeyeol Joen and Chang Heon Kim, *On the arithmetic of certain modular curves*, 2006, preprint, `http://arxiv.org/abs/math/0607611v1`.
9. Sheldon Kamienny and Barry Mazur, *Rational torsion of prime order in elliptic curves over number fields*, Astérisque (1995), no. 228, 81–100.
10. Anthony W. Knapp, *Elliptic curves*, Princeton University Press, 1992.
11. Daniel Sion Kubert, *Universal bounds on the torsion of elliptic curves*, Proceedings of the London Mathematical Society **33** (1976), 193–237.
12. Barry Mazur, *Rational points on modular curves*, Modular forms of one variable V, Lecture Notes in Mathematics, vol. 601, Springer-Verlag, 1977, pp. 107–148.
13. Loïc Merel, *Bornes pour la torsion des courbes elliptiques sur les corps de nombres*, Inventiones Mathematicae **124** (1996), 437–449.
14. Pierre Parent, *Bornes effectives pour la torsion des courbes elliptiques sur les corps de nombres*, Journal für die reine und angewandte Mathematik **506** (1999), 85–116.
15. Markus A. Reichert, *Explicit determination of nontrivial torsion structures of elliptic curves over quadratic number fields*, Mathematics of Computation **46** (1986), no. 174, 637–658.
16. Joseph H. Silverman, *The arithmetic of elliptic curves*, Springer, 1986.
17. N. J. A. Sloane, *The on-line encyclopedia of integer sequences*, 2008, `www.research.att.com/~njas/sequences/`.
18. Andrew V. Sutherland, *Computing Hilbert class polynomials with the Chinese Remainder Theorem*, in preparation.
19. Richard G. Swan, *Factorization of polynomials over finite fields*, Pacific Journal of Mathematics **12** (1962), no. 3, 1099–1106.
20. S.G. Vlăduţ, *Cyclicity statistics for elliptic curves over finite fields*, Finite Fields and Their Applications **5** (1999), 13–25.
21. Yifan Yang, *Defining equations for modular curves*, Advances in Mathematics **204** (2006), no. 2, 481–508.
22. Andrew C. Yao, *On the evaluation of powers*, SIAM Journal of Computing **5** (1976), 100–103.

6. Appendix

For reasons of space, most of the tables that follow give data only for $N \leq 30$ (in one case we also omit the full entry for $N = 29$). Full results are available in electronic form for $N \leq 50$ from

`http://math.mit.edu/~drew`.

Massachusetts Institute of Technology
E-mail address: `drew@math.mit.edu`

N	$F(r,s)$
8	$rs - 2r + 1$
9	$r - s^2 + s - 1$
10	$rs^2 - 3rs + r + s^2$
11	$r^2 - rs^3 + 3rs^2 - 4rs + s$
12	$r^2s - 3r^2 + rs + 3r - s^2 - 1$
13	$r^3 - r^2s^4 + 5r^2s^3 - 9r^2s^2 + 4r^2s - 2r^2 - rs^3 + 6rs^2 - 3rs + r - s^3$
14	$r^2s^3 - 5r^2s^2 + 6r^2s - r^2 + rs^4 - 3rs^3 + 6rs^2 - 7rs + r + s$
15	$r^3 - r^2s^5 + 7r^2s^4 - 18r^2s^3 + 19r^2s^2 - 10r^2s - rs^5 + 4rs^4 - 5rs^2 + 5rs - s^5$
	$+\ s^4 - s^3 + s^2 - s$
16	$r^3s^2 - 4r^3s + 2r^3 + 3r^2s^2 + 2r^2s - 2r^2 - rs^5 + 4rs^4 - 10rs^3 + 6rs^2 - 3rs$
	$+\ r + s^4$
17	$r^5 - r^4s^6 + 9r^4s^5 - 31r^4s^4 + 50r^4s^3 - 39r^4s^2 + 10r^4s - 3r^4 - r^3s^6 + 3r^3s^5$
	$+\ 12r^3s^4 - 46r^3s^3 + 54r^3s^2 - 15r^3s + 3r^3 - r^2s^6 - 3r^2s^5 + 9r^2s^4 + r^2s^3$
	$-\ 21r^2s^2 + 6r^2s - r^2 + rs^7 - 3rs^6 + 6rs^5 - 10rs^4 + 11rs^3 - s^3$
18	$r^4s^3 - 6r^4s^2 + 9r^4s - r^4 + r^3s^5 - 7r^3s^4 + 20r^3s^3 - 19r^3s^2 - 8r^3s + r^3 + r^2s^4$
	$-\ 11r^2s^3 + 28r^2s^2 + rs^4 - 5rs^3 - 8rs^2 + s^4 + s^3 + s^2$
19	$r^6 - r^5s^7 + 11r^5s^6 - 48r^5s^5 + 105r^5s^4 - 121r^5s^3 + 69r^5s^2 - 20r^5s - r^5$
	$-\ 2r^4s^7 + 12r^4s^6 - 9r^4s^5 - 60r^4s^4 + 144r^4s^3 - 105r^4s^2 + 35r^4s - 3r^3s^7$
	$+\ 3r^3s^6 + 21r^3s^5 - 30r^3s^4 - 41r^3s^3 + 51r^3s^2 - 21r^3s + r^2s^9 - 6r^2s^8 + 21r^2s^7$
	$-\ 50r^2s^6 + 66r^2s^5 - 31r^2s^4 + 25r^2s^3 - 18r^2s^2 + 7r^2s + 3rs^6 - 15rs^5 + 10rs^4$
	$-\ 6rs^3 + 3rs^2 - rs + s^6$
20	$r^5s^2 - 5r^5s + 5r^5 + 5r^4s^2 - 10r^4 - r^3s^7 + 9r^3s^6 - 35r^3s^5 + 70r^3s^4 - 85r^3s^3$
	$+\ 51r^3s^2 - 9r^3s + 10r^3 + 10r^2s^5 - 35r^2s^4 + 60r^2s^3 - 50r^2s^2 + 10r^2s - 5r^2$
	$-\ rs^7 + 3rs^6 - 6rs^5 + 10rs^4 - 15rs^3 + 16rs^2 - 3rs + r - s^2$
21	$r^6 - r^5s^8 + 13r^5s^7 - 69r^5s^6 + 192r^5s^5 - 300r^5s^4 + 261r^5s^3 - 119r^5s^2$
	$+\ 21r^5s - 4r^5 - r^4s^9 + 10r^4s^8 - 45r^4s^7 + 141r^4s^6 - 345r^4s^5 + 576r^4s^4$
	$-\ 551r^4s^3 + 273r^4s^2 - 49r^4s + 6r^4 - r^3s^10 + 10r^3s^9 - 51r^3s^8 + 159r^3s^7$
	$-\ 316r^3s^6 + 450r^3s^5 - 551r^3s^4 + 489r^3s^3 - 247r^3s^2 + 42r^3s - 4r^3 + 3r^2s^8$
	$-\ 31r^2s^7 + 109r^2s^6 - 172r^2s^5 + 203r^2s^4 - 181r^2s^3 + 97r^2s^2 - 14r^2s + r^2$
	$+\ 2rs^8 - 11rs^7 + 8rs^6 + 2rs^5 - 13rs^4 + 19rs^3 - 14rs^2 + rs + s^8 - s^7 + s^6 - s^5$
	$+\ s^4 - s^3 + s^2$
22	$r^6s^5 - 9r^6s^4 + 28r^6s^3 - 35r^6s^2 + 15r^6s - r^6 + r^5s^8 - 12r^5s^7 + 59r^5s^6$
	$-\ 148r^5s^5 + 205r^5s^4 - 186r^5s^3 + 133r^5s^2 - 49r^5s + 3r^5 + r^4s^8 - 6r^4s^7$
	$-\ 8r^4s^6 + 118r^4s^5 - 260r^4s^4 + 249r^4s^3 - 164r^4s^2 + 58r^4s - 3r^4 + r^3s^8$
	$-\ 30r^3s^6 + 34r^3s^5 + 70r^3s^4 - 106r^3s^3 + 80r^3s^2 - 30r^3s + r^3 + r^2s^8$
	$+\ 6r^2s^7 - 7r^2s^6 - 25r^2s^5 + 5r^2s^4 + 14r^2s^3 - 16r^2s^2 + 7r^2s - rs^9 + 3rs^8$
	$-\ 8rs^7 + 21rs^6 - 15rs^5 + 10rs^4 - 6rs^3 + 3rs^2 - rs - s^7$
23	$r^9 - r^8s^9 + 15r^8s^8 - 94r^8s^7 + 319r^8s^6 - 636r^8s^5 + 756r^8s^4 - 520r^8s^3$
	$+\ 189r^8s^2 - 35r^8s - 2r^8 - 4r^7s^9 + 39r^7s^8 - 120r^7s^7 + 28r^7s^6 + 597r^7s^5$
	$-\ 1341r^7s^4 + 1256r^7s^3 - 525r^7s^2 + 105r^7s + r^7 - 10r^6s^9 + 45r^6s^8 + 24r^6s^7$
	$-\ 357r^6s^6 + 324r^6s^5 + 570r^6s^4 - 1130r^6s^3 + 576r^6s^2 - 126r^6s + r^5s^{13} - 14r^5s^{12}$
	$+\ 93r^5s^{11} - 370r^5s^{10} + 970r^5s^9 - 1827r^5s^8 + 2553r^5s^7 - 2296r^5s^6 + 1095r^5s^5$
	$-\ 480r^5s^4 + 686r^5s^3 - 369r^5s^2 + 84r^5s + r^4s^{12} - 21r^4s^{11} + 165r^4s^{10} - 650r^4s^9$
	$+\ 1530r^4s^8 - 2562r^4s^7 + 2957r^4s^6 - 2046r^4s^5 + 780r^4s^4 - 415r^4s^3 + 171r^4s^2$
	$-\ 36r^4s + r^3s^{12} - 15r^3s^{11} + 66r^3s^{10} - 84r^3s^9 - 45r^3s^8 + 402r^3s^7$
	$-\ 833r^3s^6 + 837r^3s^5 - 351r^3s^4 + 145r^3s^3 - 48r^3s^2 + 9r^3s + r^2s^{12} - 9r^2s^{11}$
	$+\ 13r^2s^{10} - r^2s^9 - 24r^2s^8 + 28r^2s^7 + 42r^2s^6 - 126r^2s^5 + 56r^2s^4 - 21r^2s^3$
	$+\ 6r^2s^2 - r^2s + rs^{12} - 3rs^{11} + 6rs^{10} - 10rs^9 + 15rs^8 - 21rs^7 + 21rs^6 - s^6$

TABLE 4. Raw form of $X_1(N) : F(r,s) = 0$.

N	g	$d(C_0)$	$d(C_1)$	$t(C_0)$	$t(C_1)$	$k_{\max}$	$\ell(C_0, C_1)$
10	0	4	**0**	1	1	2	10
11	1	2	**2**	5	4	2	4
12	0	2	**0**	6	1	2	13
13	2	3	**2**	11	6	2	13
14	1	2	**2**	10	4	2	11
15	1	3	**2**	15	5	3	18
16	2	3	**2**	13	6	5	23
17	5	5	**4**	28	12	5	23
18	2	4	**2**	19	6	5	24
19	7	6	**5**	39	18	4	23
20	3	5	**3**	28	6	4	23
21	5	6	**4**	55	11	4	18
22	6	6	**4**	50	17	7	40
23	12	9	**7**	87	38	7	25
24	5	6	**5**	41	20	6	25
25	12	10	**8**	114	46	6	20
26	10	8	**7**	82	27	5	32
27	13	11	**8**	135	52	4	19
28	10	10	**7**	115	30	2	16
29	22	14	**11**	214	88	8	32
30	9	10	**8**	109	46	7	23
31	26	16	**13**	279	124	6	23
32	17	13	**10**	190	78	7	19
33	21	16	**12**	319	109	6	29
34	21	14	**11**	235	88	7	22
35	25	19	**15**	438	142	4	19
36	17	14	**11**	224	94	7	23
37	40	23	**18**	582	225	4	19
38	28	18	**14**	383	140	6	27
39	33	22	**17**	586	212	4	20
40	25	19	**15**	412	171	5	22
41	51	28	**22**	870	336	8	49
42	25	20	**15**	442	165	8	27
43	57	31	**24**	1065	408	6	23
44	36	24	**19**	654	208	3	21
45	41	29	**23**	960	368	4	19
46	45	26	**21**	791	285	6	23
47	70	37	**29**	1526	1768	6	33
48	37	26	**19**	773	257	7	23
49	69	39	**31**	1791	900	6	37
50	48	30	**23**	1040	391	8	42

TABLE 5. Search algorithm statistics for $X_1(N)$.

C_0 and C_1 are (respectively) the raw and optimized forms of $X_1(N)$. The column $d(C_i)$ denotes the minimum of the degree of C_i in x or y, and $t(C_i)$ denotes the number of terms. The column $\ell(C_0, C_1)$ gives the length of the path traveled by the algorithm of Section 3 to reach C_1 from C_0 (typically not a shortest path), and $k_{\max}$ is the maximum value of k prior to reaching C_1.

N	$f(x,y)$
11	$y^2+(x^2+1)y+x$
13	$y^2+(x^3+x^2+1)y-x^2-x$
14	$y^2+(x^2+x)y+x$
15	$y^2+(x^2+x+1)y+x^2$
16	$y^2+(x^3+x^2-x+1)y+x^2$
17	$y^4+(x^3+x^2-x+2)y^3+(x^3-3x+1)y^2-(x^4+2x)y+x^3+x^2$
18	$y^2+(x^3-2x^2+3x+1)y+2x$
19	$y^5-(x^2+2)y^4-(2x^3+2x^2+2x-1)y^3+(x^5+3x^4+7x^3+6x^2+2x)y^2$ $-(x^5+2x^4+4x^3+3x^2)y+x^3+x^2$
20	$y^3+(x^2+3)y^2+(x^3+4)y+2$
21	$y^4+(3x^2+1)y^3+(x^5+x^4+2x^2+2x)y^2+(2x^4+x^3+x)y+x^3$
22	$y^4+(x^3+2x^2+x+2)y^3+(x^5+x^4+2x^3+2x^2+1)y^2$ $+(x^5-x^4-2x^3-x^2-x)y-x^4-x^3$
23	$y^7+(x^5-x^4+x^3+4x^2+3)y^6+(x^7+3x^5+x^4+5x^3+7x^2-4x+3)y^5$ $+(2x^7+3x^5-x^4-2x^3-x^2-8x+1)y^4$ $+(x^7-4x^6-5x^5-6x^4-6x^3-2x^2-3x)y^3$ $-(3x^6-5x^4-3x^3-3x^2-2x)y^2+(3x^5+4x^4+x)y-x^2(x+1)^2$
24	$y^5+(x^4+4x^3+3x^2-x-2)y^4-(2x^4+8x^3+7x^2-1)y^3$ $-(2x^5+4x^4-3x^3-5x^2-x)y^2+(2x^5+5x^4+2x^3)y+x^6+x^5$
25	$y^8+(4x^2+7x-4)y^7-(x^5-x^4-14x^3-4x^2+24x-6)y^6$ $-(x^7+4x^6-3x^5-18x^4+15x^3+33x^2-30x+4)y^5$ $-(x^8+2x^7-8x^6-14x^5+24x^4+17x^3-41x^2+16x-1)y^4$ $+(x^8+6x^7+3x^6-20x^5-3x^4+28x^3-19x^2+3x)y^3$ $-(3x^7+9x^6-3x^5-13x^4+11x^3-3x^2)y^2+(3x^6+4x^5-4x^4+x^3)y-x^5$
26	$y^6+(3x^2+4x-2)y^5+(3x^4+10x^3-9x+1)y^4$ $+(x^6+7x^5+8x^4-14x^3-11x^2+6x)y^3$ $+(x^7+4x^6-x^5-13x^4+2x^3+10x^2-x)y^2$ $-(x^6-7x^4-4x^3+2x^2)y-x^4-x^3$
27	$y^8+(3x^2+6x-3)y^7-(3x^5-18x^3-9x^2+18x-3)y^6$ $-(x^8+8x^7+13x^6-21x^5-48x^4+20x^3+42x^2-18x+1)y^5$ $-(x^{10}+6x^9+12x^8-14x^7-72x^6-27x^5+93x^4+33x^3-45x^2+6x)y^4$ $+(x^{10}+11x^9+40x^8+36x^7-69x^6-105x^5+33x^4+54x^3-15x^2)y^3$ $-(4x^9+30x^8+63x^7+10x^6-69x^5-24x^4+19x^3)y^2$ $+(6x^8+27x^7+27x^6-6x^5-12x^4)y-3x^7-6x^6-3x^5$
28	$y^7+3xy^6+(x^5+3x^4+5x^3+9x^2+2x)y^5-(2x^5-6x^3+2x^2+2x)y^4$ $+(3x^6+16x^5+18x^4-2x^2)y^3+(x^7-2x^6-20x^5-28x^4-12x^3-2x^2)y^2$ $-(2x^7+3x^6-5x^5-10x^4-5x^3-x^2)y+x^7+2x^6+x^5$
29	$y^{11}+(2x^3+5x^2+5x-3)y^{10}+(x^6+8x^5+18x^4+11x^3-5x^2-12x+\cdots$
30	$y^8-(2x^3+4x^2+x+5)y^7+(x^6+4x^5+6x^4+9x^3+14x^2+10)y^6$ $-(x^7+4x^6+9x^5+10x^4+4x^3+15x^2-10x+10)y^5$ $+(x^8+4x^7+4x^6-5x^4-20x^3+5x^2-20x+5)y^4$ $+(3x^7+11x^6+15x^5+9x^4+18x^3-9x^2+14x-1)y^3$ $+(3x^6+9x^5+14x^4+2x^3+13x^2-3x)y^2+(x^5+x^4+4x^3-3x^2)y-x^3$

TABLE 6. Optimized form of $X_1(N): f(x,y)=0$.

The polynomial for $N=29$ is not displayed in full. Full polynomials for $N\le 50$ are available at `http://math.mit.edu/~drew`.

N	φ	
6	$r = x,$	$s = 1$
7	$r = x,$	$s = x$
8	$r = 1/(2-x),$	$s = x$
9	$r = x^2 - x + 1,$	$s = x$
10	$r = -x^2/(x^2 - 3x + 1),$	$s = x$
11	$r = 1 + xy,$	$s = 1 - x$
12	$r = (2x^2 - 2x + 1)/x,$	$s = (3x^2 - 3x + 1)/x^2$
13	$r = 1 - xy,$	$s = 1 - xy/(y+1)$
14	$r = 1 - (x+y)/((y+1)(x+y+1)),$	$s = (1-x)/(y+1)$
15	$r = 1 + (xy + y^2)/(x^3 + x^2y + x^2),$	$s = 1 + y/(x^2 + x)$
16	$r = (x^2 - xy + y^2 + y)/(x^2 + x - y - 1),$	$s = (x-y)/(x+1)$
17	$r = (x^2 + x - y)/(x^2 + xy + x - y^2 - y),$	$s = (x+1)/(x+y+1)$
18	$r = (x^2 - xy - 3x + 1)/((x-1)^2(xy+1)),$	
	$s = x^2 - 2x - y)/(x^2 - xy - 3x - y^2 - 2y)$	
19	$r = 1 + x(x+y)(y-1)/((x+1)(x^2 - xy + 2x - y^2 + y)),$	
	$s = 1 + x(y-1)/((x+1)(x-y+1))$	
20	$r = 1 + (x^3 + xy + x)/((x-1)^2(x^2 - x + y + 1)),$	
	$s = 1 + (x^2 + y + 1)/((x-1)(x^2 - x + y + 2))$	
21	$r = 1 + (y^2 + y)(xy + y + 1)/((xy+1)(xy - y^2 + 1)),$	
	$s = 1 + (y^2 + y)/(xy + 1)$	
22	$r = (x^2y + x^2 + xy + y)/(x^3 + 2x^2 + y),$	$s = (xy + y)/(x^2 + y)$
23	$r = (x^2 + x + y + 1)/(x^2 - xy),$	$s = (x + y + 1)/x$
24	$r = (x^2 + x - y + 1)/(x^2 + xy - y^2 + y),$	$s = (x+1)/(x+y)$
25	$r = (x^2 + xy + y^2 - y)/(x^2 + x + y - 1),$	$s = (x+y)/(x+1)$
26	$r = (x^3y + 3x^2y - x^2 + xy^2)/((x+1)(x^2y + x^2 + 3xy + y^2)),$	
	$s = (xy - x)/(xy + y)$	
27	$r = (-x^3 - x^2 - x - y)/(x^2y + xy - x - y),$	$s = (-x^2 - x - y)/(xy - x - y)$
28	$r = 1 + (xy + y)/((y-1)(xy - x + 2y - 1)),$	
	$s = 1 - (xy + y)/((y-1)(x - y + 1))$	
29	$r = (-x^3 - x^2 - x - y)/(x^2y + xy - x - y)$	
	$s = 1 - (x^2 + xy)/(xy - x - y)$	
30	$r = (x^2y + x + y)/(x^2y - xy + x),$	
	$s = (x^2y + xy + x + y)/(x^2y + x)$	

TABLE 7. Birational maps for $X_1(N)$ from $f(x,y) = 0$ to $F(r,s) = 0$.