

ELLIPTIC CURVES OF FIBONACCI PRIME ORDER OVER \mathbb{F}_p

ROSINA CAMPBELL, DUC VAN HUYNH¹, TYLER MELTON, AND ANDREW PERCIVAL

ABSTRACT. We will describe an algorithm to construct an elliptic curve E_{f_q} over some prime field \mathbb{F}_p such that $|E_{f_q}(\mathbb{F}_p)| = f_q$, where f_q is a probable Fibonacci prime for some prime index q . The algorithm is a variant of the efficient CM-construction by Bröker and Stevenhagen, which is well suited for Fibonacci primes due to their arithmetic properties. The time complexity of our algorithm is expected to be lower than $\tilde{O}(\log^3(f_q))$. The construction process is a series of algorithms, where each is a test for primality.

1. INTRODUCTION

Let $p > 3$ be a rational prime. Henceforth, for each elliptic curve E over \mathbb{F}_p , we say that the order of E is $|E(\mathbb{F}_p)|$, that is, the number of \mathbb{F}_p -rational points on E . There exists an elliptic curve E of order N for each integer N in the Hasse interval $H_p = [p + 1 - 2\sqrt{p}, p + 1 + 2\sqrt{p}]$ [Cox13, Theorem 14.18]. Note that $N \in H_p$ if and only if $p \in H_N$, which is a motivation behind the algorithm in [BS08]. Hence, the construction of E is possible exactly when H_N contains a prime p . Under General Riemann Hypothesis (GRH), we can safely assume the existence of a prime p in H_N (see [BS07]).

This paper is on the study of constructing elliptic curves of Fibonacci prime order over finite fields. A Fibonacci prime is a Fibonacci number that is also prime. It is not known whether there is an infinite number of Fibonacci primes, though heuristics regarding elliptic divisibility sequence (EDS) from [EEW01] suggests it may be finite. Constructing elliptic curves of Fibonacci order is of interest due to the fact that Fibonacci numbers grow exponentially, and the large width of the Hasse interval H_{f_q} is expected to contain many primes, and Conjecture 4 of [BL07] suggests that the time complexity of the construction may be smaller than it is for other primes. We will also see that the arithmetic properties of Fibonacci primes make some of the computations relatively easier. Furthermore, the construction process allows us to test its primality along the way, with the Elliptic Curve Primality Proving (ECPP) being the main test.

Acknowledgment This work is a year-long undergraduate research project with the students Rosina Campbell, Tyler Melton, and Andrew Percival. We were partially supported by the Armstrong State University Summer Research Session Grant and by the Armstrong Active Learning Grant. We would like to thank Ayman Bagabas, Chanukya Badri, Donald Hinton, and Keyur Patel for helpful discussion

DEPARTMENT OF MATHEMATICS, ARMSTRONG STATE UNIVERSITY, SAVANNAH, GA 31419
E-mail addresses: rc2283@stu.armstrong.edu, duc.huynh@armstrong.edu,
 tm3746@stu.armstrong.edu, ap8822@stu.armstrong.edu.

Date: October 17, 2017.

¹Corresponding author.

on computing square roots in finite field. Finally, we are grateful of Armstrong's Center for Applied Cyber Education for their continual support.

2. MAIN ALGORITHM

Henceforth, we will use the following notations. We will denote the natural logarithm by \log and for convenience we will write $\log^r(x) = (\log(x))^r$ for real numbers $r > 0$. The notation $O(x)$ denotes the standard big O notation, and the notation $\tilde{O}(x)$ means logarithmic terms in x are disregarded.

Given a finite field \mathbb{F}_p , we will assume that all computations in \mathbb{F}_p are done using the best known methods. For example, for fast multiplication we will assume the Fast Fourier Transform (FFT) method of [SS71], which has time complexity $\tilde{O}(\log(p))$. The fast multiplication technique is applicable here since the probable Fibonacci primes are at least $2^{50,000}$. For fast exponentiation, we will assume the method of exponentiation by squaring (see [Coh93, ch. 1]), which has time complexity $\tilde{O}(\log^2(p))$ if combined with the fast multiplication method. Similarly, using FFT, the time complexities of multiplication and exponentiation in $\mathbb{F}_p[X]$ are the same as in \mathbb{F}_p for polynomials of small degrees.

Let f_q be a probable Fibonacci prime. Here, we say that f_q is a probable prime if at minimum q is a prime (See Lemma 11.7). Our work is based on the list of probable Fibonacci primes given at [on17]. We assume that these Fibonacci numbers have been tested under various algorithms, so we do not expect the algorithm to fail before Step 8.. Here in this work we try to construct an elliptic curve of order f_q and test the primality of f_q along the way. The guiding principle behind Algorithm 2.1 is to interpret all computations as primality tests, though some may be primitive, and we try to reuse all computations whenever possible.

Algorithm 2.1. Let f_q be a probable Fibonacci prime. This algorithm attempts to construct an elliptic curve $E/(\mathbb{Z}/p\mathbb{Z})$ of order f_q over some ring $\mathbb{Z}/p\mathbb{Z}$ and performs multiple primality verifications along the way.

1. Apply the Density Test 13.1.
2. From Step 1. we obtain the set P_q of primes $\ell < 2 \log(f_q)$ such that $\left(\frac{f_q}{\ell}\right) = 1$, and we also obtain the first prime n such that n is a quadratic non-residue modulo f_q . This n may be greater than $2 \log(f_q)$.
3. Use Algorithm 5.2 to obtain a list S_q of good discriminants, and let $N = |S_q|$.
4. Use n to perform square root precomputations using Algorithm 9.1.
5. Verify that $f_{(q+1)/2}/f_{(q-1)/2} \pmod{f_q}$ is a square root of $-1 \pmod{f_q}$.
6. Apply the Exceptional Cases Test 12.1.
7. Let $k = 0$.
8. If $k = N$, then go to Step 18., else take $D = S_q[k] \in S_q$ and find a square root $\sqrt{D} \pmod{f_q}$ of D using Algorithm 9.2. Here if D consists of two primes ℓ_1 and ℓ_2 , then use the previously computed $\sqrt{\ell_1} \pmod{f_q}$ and $\sqrt{\ell_2} \pmod{f_q}$.
9. Apply Algorithm 9.3 to determine if f_q split completely in H_K , the Hilbert class field of $K = \mathbb{Q}(\sqrt{D})$. If f_q does split completely, we obtain $4f_q = x^2 + y^2|D|$ for some positive integers x, y . If this step is not successful, increase k by 1 and return to Step 8..

10. Precompute the classical modular polynomials $\Phi_\ell(X, Y)$ for primes $\ell < 6 \log^2(4 \log^2(f_q))$.
11. Let $p = f_q + 1 \pm x$. If it is easy to recognize that $p = k\eta$ for some prime $\eta > (f_q^{1/4} + 1)^2$, then construct $E/(\mathbb{Z}/f_q\mathbb{Z})$ of order $f_q + 1 \pm x$ following Step 13. and Step 14. and apply ECPP (Theorem 13.4) to test the primality of f_q , else go to the next step.
12. Apply the Rabin-Miller Primality Test (Algorithm 13.3) to p . If p passes the test, then go the next step, else increase k by 1 and return to Step 8..
13. Compute $H_D(X) \pmod{p}$ using Algorithm 6.1.
14. Find a root $r \neq 0, 1728$ of $H_D(X) \pmod{p}$. If no root is found, increase k by 1 and return to Step 8., otherwise construct the curve

$$E : Y^2 = X^3 + aX - a, \quad (2.1)$$

where

$$a = \frac{27r}{4(1728 - r)} \pmod{p}. \quad (2.2)$$

15. Let \mathcal{E}_q be an empty list.
16. Append one of (E, p, D) or (E_{twist}, p, D) to the list \mathcal{E}_q for which $f_q \cdot (1, 1) = 0$ is satisfied. If neither of the twists satisfy $f_q \cdot (1, 1) = 0$, then increase k by 1 and return to Step 8., else go to the next step.
17. If f_q is a confirmed prime, then p is prime as well by ECPP. Output (E, p, D) and stop the algorithm. If the primality of f_q is not confirmed, then increase k by 1 and return to Step 8..
18. Apply Algorithm 10.1 (Elkies Primes Verification) to the list \mathcal{E}_q . If \mathcal{E}_q is empty, then f_q is likely composite, else go to the next step.
19. Apply Algorithm 10.2 (Eigenvalue Verification) to the list \mathcal{E}_q . If \mathcal{E}_q is empty, then f_q is likely composite, else go to the next step.
20. Output a random element (E, p, D) from \mathcal{E}_q .

A small issue with Algorithm 2.1 is the primality of p . By ECPP (Theorem 13.4), the verification $f_q \cdot (1, 1) = 0$ in Step 14. does confirm that $p = f_q + 1 \pm x$ is a prime if f_q is known to be prime. However, f_q is a probable Fibonacci prime that we wish to test the primality of.

If p is confirmed to be prime, then f_q is automatically prime by ECPP in Step 11. and we have an elliptic curve E/\mathbb{F}_p of order f_q . Even when p fails to be prime, we can still apply ECPP to determine the primality of f_q . Explicitly, if $p = k\eta$ for some recognizable prime $(f_q^{1/4} + 1)^2 < \eta < p$, then we have that f_q is a prime assuming that $k \cdot (1, 1)$ is defined and not equal to 0. If it is not easy to confirm the primality of η , then we move on to another discriminant. This is a common technique in applying ECPP (see [ACD⁺06, pp. 597]).

The philosophy behind Step 18. and Step 19. is to prolong computations in $\mathbb{Z}/p\mathbb{Z}$ to detect the compositeness of p , and we want to verify the order of each curve $E/(\mathbb{Z}/p\mathbb{Z})$ as well. We will use Schoof's algorithm to verify $t \pmod{\ell}$ for some small Elkies primes ℓ relative to f_q . In fact, the largest probable Fibonacci prime from the list [onl17] has index 2904353, so we need to compute $\Phi_\ell(X, Y)$ for Elkies primes $\ell < 5287$. Of course if storage capacity is not limited, then it is practical to precompute the modular polynomials $\Phi_\ell(X, Y)$ for large Elkies primes ℓ if multiple probable Fibonacci primes are to be tested. Note here that we can not use modular polynomials for Weber's function since $D \equiv 5 \pmod{8}$. A

combination of isogeny volcanoes with other class invariant such as Ramanujan's class invariant (see [Kon14]) can allow one to work with larger Elkies primes, but such implementation is beyond our capacity. See [BLS12] for computation of the the class modular polynomials $\Phi_\ell(X, Y)$ for large primes ℓ via isogeny volcanoes, which suggests it is best to compute $\Phi_\ell(X, Y) \pmod{p}$ as needed.

Theorem 2.2. Assuming GRH, the time complexity of the Algorithm 2.1 is $\tilde{O}(\log^3(f_q))$. Furthermore, the space required is of size $\tilde{O}(\log^2(f_q))$.

Proof. Algorithm 2.1 is similar to the algorithm from [BS08], which has time complexity $\tilde{O}(\log^3(f_q))$. The only difference here is that our algorithm is a bit more convoluted and we have the extra final verifications using Schoof's algorithm in Step 18. and Step 19., which have time complexity $\tilde{O}(\log^2(f_q))$ and $\tilde{O}(\log^3(f_q))$, respectively.

Each step has time complexity of at most $\tilde{O}(\log^2(f_q))$. Even though our algorithm has $O(\log^2(f_q))$ loops, we will show that the steps that have time complexity $\tilde{O}(\log^2(f_q))$ get called only $O(\log(f_q))$ times.

The square root algorithm (Algorithm 9.2) get called only in the case $D = -\ell$, which happens $O(\log(f_q))$ times. Since each square root computation takes time $\tilde{O}(\log^2(f_q))$, the total time on computing square roots is $\tilde{O}(\log^3(f_q))$.

By the Chebotarev Density Theorem, it is expected to find $O(\log(f_q))$ discriminants D for which $4f_q = x^2 + y^2|D|$ for some positive integer x, y ; we will see this fact in Section 5. Since the time complexity of each step from Step 9. to Step 16. is at most $\tilde{O}(\log^2(f_q))$, we have a total time complexity of $\tilde{O}(\log^3(f_q))$ if we loop through all such discriminants. Therefore, the entire algorithm has time complexity $\tilde{O}(\log^3(f_q))$.

The main step in Algorithm 2.1 requiring the most storage is in computing $H_D(X) \pmod{p}$. By [Sut11], the storage needed to computed $H_D(X) \pmod{p}$ is $O(|D|^{1/2+\epsilon} \log(p))$. As $D = O(\log(f_q)^2)$ and $p = O(f_q)$, it follows that the storage required for Algorithm 2.1 is $\tilde{O}(\log^2(f_q))$. □

It is of small concern (or of great fortune) if Conjecture 4 of [BL07] is true. Let d be an integer, and let \mathcal{N}_d be the set of positive integers defined by

$$\mathcal{N}_d = \{n > 0 : f_n = |x^2 + dy^2| \text{ for some integers } x \text{ and } y\}. \quad (2.3)$$

The lower asymptotic density $\underline{\delta}(\mathcal{N}_d)$ of \mathcal{N}_d is defined by

$$\underline{\delta}(\mathcal{N}_d) = \liminf_{n \rightarrow \infty} \frac{1}{n} \sum_{k=1}^n \chi_{\mathcal{N}_d}(k) = \liminf_{n \rightarrow \infty} \frac{|\mathcal{N}_d \cap [1, n]|}{n}, \quad (2.4)$$

where $\chi_{\mathcal{N}_d}$ is the characteristic function of the set \mathcal{N}_d . The conjecture states that the lower asymptotic density $\underline{\delta}_{\mathcal{N}_d}$ is 0 for all but finitely many integers d not a square or square.

For a Fibonacci prime f_q and a positive square-free integer d , we have $f_q = x^2 + dy^2$ for some integers x, y exactly when f splits completely in the ring class field of the imaginary quadratic field $K = \mathbb{Q}(\sqrt{-d})$. Here the ring class field is the extension R_K/K corresponding to the ideal class group $C(\mathbb{Z}[\sqrt{-d}])$ given by Class Field Theory. The extension R_K/H_K is of degree 2 exactly when $-d \equiv 1$

(mod 4) and degree 1 when $-d \equiv 3 \pmod{4}$, where H_K is the Hilbert class field corresponding to the maximal order \mathcal{O}_K .

Let h_d be the class number of \mathcal{O}_K . By the Chebotarev Density Theorem, the density of the rational primes that split completely in H_K is $1/2h_d$, and half of those split completely in R_K , that is, 1 out of $4h_d$ rational primes split completely in R_K . The conjecture implies that if d is large enough, only a few Fibonacci primes split completely in R_K , at best it is an infinite set of density 0, which implies that the Fibonacci primes splitting completely in H_K has lower asymptotic density 0 as well. This may pose difficulty to Algorithm 2.1 due to its reliance on finding a Hilbert class field H_K for which f_q splits completely. On the other hand, the conjecture also suggests that the Fibonacci primes only split completely in Hilbert class fields (induced by *small* discriminant). This is fortunate as we wish to find a small fundamental discriminant D for which $4f_q = x^2 + y^2|D|$ for some integers x, y such that $f_q + 1 \pm x$ is a prime, so Step 9. has a higher chance of success. Hence, the time complexity may be lower than $\tilde{O}(\log^3(f_q))$.

3. OVERVIEW

In Section 4, we will see that one could in theory construct an elliptic curve of order f_q by picking a prime p in the Hasse interval H_{f_q} and finding a root of $H_D(X) \pmod{p}$, where $D = (p + 1 - f_q)^2 - 4p$. Of course, this is highly impractical as the discriminant D may be too large, and there does not exist an efficient method to find the fundamental discriminant induced by D . In [BS08], it is observed that

$$D = (p + 1 - f_q)^2 - 4p = (f_q + 1 - p)^2 - 4f_q. \quad (3.1)$$

This observation allows us to construct a suitable fundamental discriminant from a basis of primes (Section 5). This is a small discriminant that induces a class field in which both of the primes p and f_q split completely.

We will be mainly implementing the complex multiplication method for the construction, and we will use the algorithm from [BS08] to find a small discriminant. One major hurdle of [BS08] in constructing an elliptic curve of prime order N is to compute a square root of $(-1)^{\frac{\ell-1}{2}} \ell$ modulo N for various primes ℓ , but we will see that for each probable Fibonacci prime f_q it is a straightforward application of the Tonelli-Shanks algorithm because the 2-Sylow subgroup of $\mathbb{Z}/(f_q - 1)\mathbb{Z}$ is small.

We will provide in Section 4 a brief overview of complex multiplication and its application toward constructing elliptic curves of prescribed torsion. In Section 5 we will describe the algorithm from [BS08] to efficiently find small discriminant. We will provide a quick overview of the CRT method in Sections 6 to 8 following [Sut11]. We will go over computing square roots in finite field and Cornacchia's algorithm in Section 9. In Section 10 we will discuss Schoof's algorithm and provide a way to verify that we have the correct curve. We will go over some elementary properties of Fibonacci numbers in Section 11. We will look into equations of the form $4f_q = x^2 + dy^2$ for special d 's in Sections 12. Finally, in Section 13 we will discuss the primality tests that are used in our algorithm.

4. COMPLEX MULTIPLICATION AND APPLICATION

In this section, we will provide an overview of complex multiplication and its application in constructing elliptic curves of prescribed order. For further discussion

on complex multiplication see [Sch95], [AM93], [Cox13, pp. 190–196], [Sil94, pp.95-100], [ACD⁺06, pp. 455-460], and [Che12]

Henceforth, let D be a negative discriminant. The polynomial $F(x, y) = aX^2 + bXY + cY^2$ is called a binary quadratic form, where $a, b, c \in \mathbb{Z}$. We say that the form F is reduced if $\gcd(a, b, c) = 1$, and a, b , and c satisfy the condition

$$|b| \leq |a| \leq c \text{ and } b \geq 0 \text{ whenever } |b| = a \text{ or } a = c. \quad (4.1)$$

The discriminant of F is defined by $D = b^2 - 4ac$.

To each form F , we associate the matrix

$$M_F = \begin{bmatrix} a & \frac{b}{2} \\ \frac{b}{2} & c \end{bmatrix}. \quad (4.2)$$

Two forms F_1, F_2 are said to be equivalent if there exists a matrix $N \in \mathrm{SL}_2(\mathbb{Z})$ such that

$$M_{F_2} = N^{-1}M_{F_1}N. \quad (4.3)$$

The relation defines an equivalence relation on quadratic forms. The set of equivalence classes $C(D)$ of quadratic forms with discriminant D forms an abelian group, and each class contains exactly one reduced form by Theorem 2.8 of [Cox13]. We will call $C(D)$ the class group induced by discriminant D .

Let $K = \mathbb{Q}(\sqrt{d})$ for some rational integer $d < 0$, and let O_K be its ring of integers. Let $\mathcal{O} \subset O_K$ be an order of index f , which is called the conductor of \mathcal{O} . The discriminant of \mathcal{O} is $D = \mathrm{Disc}(\mathcal{O}) = f^2 d_K$, where d_K is the field discriminant (or fundamental discriminant) of K . From [Cox13, Theorem 5.30], we have

$$C(D) \cong C(\mathcal{O}), \quad (4.4)$$

where $C(\mathcal{O})$ is the ideal class group of \mathcal{O} . Explicitly, the isomorphism $C(D) \xrightarrow{\sim} C(\mathcal{O})$ above is given by

$$aX^2 + bXY + cY^2 \mapsto [a, (-b + \sqrt{D})/2], \quad (4.5)$$

where $aX^2 + bXY + cY^2$ is a reduced form of discriminant D . This provides an easy way to study the group $C(\mathcal{O})$, and in particular, to compute the class order.

As $C(\mathcal{O})$ is a quotient of the ray class group of conductor $\mathfrak{f} = fO_K$ of K , Class Field Theory (see [Cox13, Theorem 8.6]) tells us that there exists a unique abelian extension L/K such that

$$C(\mathcal{O}) \cong \mathrm{Gal}(L/K), \quad (4.6)$$

where the isomorphism is given by the Artin map.

Let \mathbb{H} be the upper half of the complex plane, and let h be the order of $C(D)$. Then the minimal polynomial $H_D(x)$ of L/K is given by

$$H_D(x) = \prod_{i=1}^h (x - j(\tau_i)), \quad (4.7)$$

where $\tau_i \in \mathbb{H}$ is a root of $F(x, 1)$, the reduced form representing the class $[F(x, y)]$ in $C(D)$, and j is the well-known j -invariant function (\mathbb{C} -isomorphism)

$$j : X(1) \rightarrow \mathbb{P}^1(\mathbb{C}), \quad (4.8)$$

where $X(1)$ is the modular curve

$$X(1) = \frac{\mathbb{H} \cup \mathbb{P}^1(\mathbb{Q})}{\mathrm{SL}_2(\mathbb{Z})} \quad (4.9)$$

and $\mathbb{P}^1(\mathbb{C})$ is the Riemann sphere. Moreover, the Fourier series expansions of j begins with

$$j(\tau) = q^{-1} + 744 + 196884q + 2149376q^2 + \dots, \quad (4.10)$$

where $q = e^{2\pi i\tau}$. One of the miracles in Explicit Class Field Theory is that $L = K(j(\tau))$, where $j(\tau)$ is any root of $H_D(X)$ - fulfilling Kronecker's jugendtraum.

The polynomial $H_D(X)$ in Equation 4.7 is called the Hilbert class polynomial associated with the order \mathcal{O} . It seems that it is standard to call $H_D(X)$ a Hilbert class polynomial regardless of whether \mathcal{O} is maximal. We will continue such naming standard.

Computing the Hilbert class polynomial can be quite difficult. Besides the already daunting time complexity in computing $H_D(X)$, the storage required to store its coefficients may be beyond practical purpose. For example, it requires 47.2 petabytes of storage in constructing $H_D(X)$ for $D = -(10^{16} + 135)$ (see [Sut12a]). The complex-analytic method of computing $H_D(X)$ is to approximate each root using the expansion 4.10, and to verify for accuracy, we use the fact that $H_D(x)$ has integer coefficients, and $\sqrt[3]{H_D(0)} \in \mathbb{Z}$, a consequence of the work of Gross and Zagier [AM93, Proposition 7.1]. There are two other known methods to compute $H_D(X)$: the p -adic lifting method (see [Bro08]), and application of isogeny volcanoes and Chinese Remainder Theorem (see [Sut11] and its accelerated version [Sut12a]). We will see a quick overview of isogeny volcanoes in Section 6. It is interesting to note that the complex-analytic method has to deal with rounding errors, while the p -adic lifting method circumvent that by working in a non-archimedean setting.

There is a correspondence between representatives of $C(\mathcal{O})$ and \mathbb{C} -isomorphism class of elliptic curves with endomorphism ring isomorphic to \mathcal{O} (see [Cox13, Corollary 10.20]). Viewing each ideal \mathfrak{a} of \mathcal{O} as a lattice of \mathbb{C} , the correspondence is given by

$$\mathfrak{a} \mapsto \mathbb{C}/\mathfrak{a}, \quad (4.11)$$

and in view of the correspondence from 4.5, we have

$$j((a + b\sqrt{D})/2) = j(\mathbb{C}/\mathfrak{a}). \quad (4.12)$$

It is now straightforward to obtain an algebraic model for each curve \mathbb{C}/\mathfrak{a} . For each root $r \neq 0, 1728$ of $H_D(X)$, let E_r/L be the elliptic curve given

$$Y^2 = X^3 + aX - a, \quad (4.13)$$

where $a = \frac{27r}{4(1728-r)}$. While if $r = 0$, let E_r/L be given by

$$Y^2 = X^3 + 1, \quad (4.14)$$

and if $r = 1728$, let E_r/L be given by

$$Y^2 = X^3 + X. \quad (4.15)$$

The elliptic curve E_r/L has coefficients in L , and its j -invariant is $j = r$. Furthermore, the endomorphism ring $\text{End}(E_r)$ is isomorphic to \mathcal{O} . Henceforth, we will identify $\text{End}(E_r)$ with \mathcal{O} .

Let p be a rational prime that splits in K , and let \mathfrak{p} be a prime ideal of \mathcal{O}_L that divides the ideal (p) . Assume that $\mathfrak{p} \nmid \Delta(E_r)$, where $\Delta(E_r) = -16(4a^3 + 27a^2)$ is the discriminant of E_r , then E has good reduction at \mathfrak{p} . The reduction \overline{E} of $E \bmod \mathfrak{p}$ has coefficients in some finite extension \mathbb{F}_q of \mathbb{F}_p , so in the case that p splits completely in L or the class group $C(D)$ has order 1, the reduction \overline{E}

has coefficients in \mathbb{F}_p . The endomorphism ring of \overline{E} is isomorphic to \mathcal{O} and its j -invariant is a root of $H_D(X) \pmod{q}$. Deuring's Reduction Theorem tells us that every elliptic curve over \mathbb{F}_q with endomorphism ring isomorphic to \mathcal{O} arises this way (see [Cox13, Theorem 14.16]). Moreover, we have

$$|\overline{E}(\mathbb{F}_q)| = q + 1 - t, \quad (4.16)$$

where $t = \pi + \overline{\pi}$ for some $\pi \in \mathcal{O}$ such that $q = \pi\overline{\pi}$. If we have an element $\beta \in \mathcal{O}$ such $q = \beta\overline{\beta}$, then $\beta/\pi \in \mathcal{O}^\times$. As we will be working with \mathcal{O} with discriminant $D < -4$, the group of units $\mathcal{O}^\times = \{\pm 1\}$, so $\beta + \overline{\beta}$ may differ from t by a negative sign.

We observe that the roots of $H_D(X)$ are the j -invariants of all elliptic curves E/L with endomorphism ring \mathcal{O} . Let $\text{Ell}_{\mathcal{O}}(L)$ be the set of all roots of $H_D(X)$. The ideal class group $C(\mathcal{O})$ provides a free transitive group action on the set $\text{Ell}_{\mathcal{O}}(L)$. To see the group action, let $\left(\frac{\cdot}{L/K}\right) : C(\mathcal{O}) \rightarrow \text{Gal}(L/K)$ be the Artin map. For an invertible ideal $\mathfrak{a} \in \mathcal{O}$, we have

$$\left(\frac{\mathfrak{a}}{L/K}\right)(j(E)) = j(E/E[\mathfrak{a}]), \quad (4.17)$$

where $E[\mathfrak{a}]$ is the \mathfrak{a} -torsion of E (see [Cox13, ch. 11]). The fact that this action is free and transitive follows from the fact that \mathcal{C}/\mathfrak{a} determines an isomorphism class of elliptic curves over L with endomorphism ring \mathcal{O} (see [Sut11], [Cox13, Corollary 10.20] and [Bro08] for further details).

As $C(\mathcal{O}) \cong C(D)$ and there is a bijection between $\text{Ell}_{\mathcal{O}}(\mathbb{F}_p)$ and $\text{Ell}_{\mathcal{O}}(L)$ by Deuring lifting theorem, there is a free transitive group action of $C(D)$ on $\text{Ell}_{\mathcal{O}}(\mathbb{F}_p)$. Hence, if we have one root j_0 of $H_D(X) \pmod{p}$, we can obtain the rest by computing the orbit of the group action of $C(D)$ on j_0 . This fact is used in [Sut11], which we will see an overview of in Section 6.

As mentioned earlier, the reduction \overline{E} at \mathfrak{p} has coefficients in \mathbb{F}_p exactly when \mathfrak{p} splits completely in L and p splits in K , which happens exactly when $H_D(X) \pmod{p}$ splits completely over \mathbb{F}_p (see [Cox13, Theorem 5.1, Theorem 9.2]). Since L/K is Galois, $H_D(X) \pmod{p}$ splits completely exactly when $H_D(X) \pmod{p}$ has a root in \mathbb{F}_p for $p \nmid D$. On the other hand, from Class Field Theory the prime ideal \mathfrak{p} splits completely in L exactly when \mathfrak{p} is principal, which happens exactly when the rational prime p is a norm in \mathcal{O} , that is, there exists integers x, y such that

$$4p = x^2 + y^2|D|. \quad (4.18)$$

Hence, $H_D(X) \pmod{p}$ has a root if and only if the $4p = x^2 + y^2|D|$ for some positive integers x, y .

The previous paragraphs provide a method to construct elliptic curves over a prime field of prescribed order. Indeed, let N be a positive integer and let $t = p + 1 - N$, where p is a prime so that $|t| \leq 2\sqrt{p}$, that is, p is in the Hasse's interval H_N . Let $D = (p + 1 - N)^2 - 4p$, and compute $H_D(X) \pmod{p}$. We find a root $r \neq 0, 1728$ in \mathbb{F}_p of $H_D(X) \pmod{p}$, which exists because the equation $4p = X^2 + Y^2|D|$ has the solution $(t, 1)$. Compute $a = \frac{27r}{4(1728-r)} \pmod{p}$, and consider the elliptic curve E/\mathbb{F}_p defined by $E : Y^2 = X^3 + aX - a$. The order of E is $p + 1 \pm t$, so we may have to compute its quadratic twist (see Proposition 5.4 of [Sil09])

$$E_{\text{twist}} : Y^2 = X^3 + g^2aX - g^3a \quad (4.19)$$

if necessary to find the one with order $N = p + 1 - t$, where g is any quadratic non-residue modulo p . The fact that the point $(1, 1)$ lies on E allows us to quickly determine which curve has the correct order. For the cases $r = 1728$ or $r = 0$, the set of twists of the curves $Y^2 = X^3 + X$ and $Y^2 = X^3 + 1$ correspond to $\mathbb{F}_p^*/(\mathbb{F}_p^*)^4$ and $\mathbb{F}_p^*/(\mathbb{F}_p^*)^6$, respectively (see Proposition 5.4 of [Sil09]).

Algorithm 4.1. Complex Multiplication algorithm to construct an elliptic curve of order N over some prime field.

1. Find a prime $p \in H_N$.
2. Compute $D = (p + 1 - N)^2 - 4p$.
3. Compute the Hilbert class polynomial $H_D(X)$.
4. Find a root r of $H_D(X) \pmod{p}$.
5. If $r \neq 0, 1728$, construct the curve $E : Y^2 = X^3 + aX - a$, where $a = (27r)/(4(1728 - r)) \pmod{p}$.
6. If $r = 0$, take $E : Y^2 = X^3 + 1$, and if $r = 1728$, take $E : Y^2 = X^3 + X$.
7. Test the point $(1, 1)$ of E , that, is verify

$$(p + 1)(1, 1) = t(1, 1). \quad (4.20)$$

8. Compute a twist of E if necessary.

A careful analysis shows that instead of using $D = (p + 1 - N)^2 - 4p$, we could use the fundamental discriminant $D_K = D/k^2$, for some integer k . Here, fundamental discriminant is equivalent to the field discriminant of $K = \mathbb{Q}(\sqrt{D})$. As the fundamental discriminant of D is essentially its square-free part, computing the fundamental discriminant of D is not practical as computing the square-free part of an integer is exceedingly difficult.

Constructing elliptic curves of prescribed torsion has its applications in cryptography (ECC), mainly in creating keys for encryption systems such as AES. In the coming age (or current age) of quantum computers, a system such as ECC will be (is) breakable. There has been active research into post-quantum cryptography in the last decade to find a system that could withstand a quantum computer. For example, the Supersingular Isogeny Diffie-Hellman Key Exchange has been shown to be a great candidate (see [DFJP14]).

Remark 4.2. Instead of using the Hilbert class polynomial, one could also use *smaller* class polynomials such as Weber class polynomials (see [KKSZ09]) and Ramanujan's class polynomial (see [KK10]). Among the well-known class polynomials, the data from [KK10] shows that the Ramanujan's class polynomials are best for generating elliptic curves of prime order. One could also in theory find other class invariants with smaller class polynomials using a variant of Shimura Reciprocity (see [Gee99] and [Kon14]). Here we say that $f(\tau)$ is a class invariant of a Hilbert class field H_K if $H_K = K(f(\tau))$, where f is a modular function of some level and $\tau \in \mathcal{O}_K$. However, as worded best by Kontogeorgis in [Kon14]: *So far it seems that all known class invariants were found out of luck or by extremely ingenious people like Ramanujan.*

5. AN EFFICIENT CM-CONSTRUCTION

The following is a discussion of an algorithm described in [BS07] and [BS08]. The algorithm reduces the time in calculating the Hilbert class polynomial by minimizing $|D|$, which can be done by constructing D from a set of basis of primes.

Let N be a rational prime. Recall that in constructing an elliptic curve of order N , a bottleneck is to construct the Hilbert class polynomial $H_D(X)$, where $D = (p + 1 - N)^2 - 4p$ and p belongs in the Hasse's interval H_N . Even though it is best to use the field discriminant of $\mathbb{Q}(\sqrt{D})$, which is essentially the square-free part of D , the time complexity is too large for practical purpose as there does not exist a known polynomial time algorithm in computing the square-free part of an integer.

Instead of using a top-down approach as above in finding D , we can construct D from a set of basis of primes. Note that given a prime $p \in H_N$, we have the discriminant

$$(p + 1 - N)^2 - 4p = (N + 1 - p)^2 - 4N = k^2 D, \quad (5.1)$$

for some fundamental discriminant D . It follows that for a fundamental discriminant D , if we can find a solution to the equation

$$x^2 + y^2 |D| = 4N, \quad (5.2)$$

for some positive integers x, y with $p = N + 1 \pm x$ prime, then we can construct an elliptic curve E/\mathbb{F}_p of order N . Here we are using the symmetry $N \in H_p$ if and only if $p \in H_N$. Hence, we are trying to find a discriminant D for which both $H_D(X) \pmod{p}$ and $H_D(X) \pmod{N}$ split completely.

From equation 5.1, we note that for any odd prime $\ell \mid D$, we have

$$1 = \left(\frac{N}{\ell}\right) = \left(\frac{(-1)^{\frac{\ell-1}{2}} \ell}{N}\right), \quad (5.3)$$

where the second equality comes from the Law of Quadratic Reciprocity. Letting $\ell^* = (-1)^{(p-1)/2} \ell$, we find that D consists of primes ℓ for which ℓ^* is a quadratic residue modulo N . Moreover, since N is assumed to be a rational prime, we must have $D \equiv 5 \pmod{8}$. Hence, we see that D is a product of the primes ℓ satisfying Equation 5.3.

Remark 5.1. The fact above regarding the primes ℓ dividing D can be easily seen using Class Field Theory. Recall that for a rational prime N , the equation $4N = x^2 + y^2 |D|$ has a solution in \mathbb{Z}^2 exactly when N splits completely in the Hilbert class field K_D of the quadratic field $K = \mathbb{Q}(\sqrt{D})$. In particular, N splits completely in any subfield of K_D of K . Hence, N must splits completely in the genus field G_D (see [Cox13, Theorem 6.1]) and all of its quadratic subfields $\mathbb{Q}(\sqrt{\ell^*})$ for primes $\ell \mid D$, which happens exactly when ℓ^* is a square modulo N .

The Prime Number Theorem states that

$$\lim_{x \rightarrow \infty} \frac{\pi(x)}{x/\log(x)} = 1, \quad (5.4)$$

where $\pi(x)$ is the number of rational primes less than or equals to x . It follows that for integer N sufficiently large, we expect 1 out $\log(N)$ to be prime. Hence, instead of searching for all suitable primes ℓ in the interval $[1, N]$ at once, we search for ℓ in one sub-interval at a time starting with $[1, \log(N)]$ and ends with $[(m-1)\log(N) + 1, N]$, where $m = \lfloor N/\log(N) \rfloor$. However, such a partition of the interval $[1, N]$ is only for ease of computing time complexity; in practice, we work with one prime ℓ at a time, which is better for Fibonacci primes due to Conjecture 4 of [BL07]. Furthermore, It is noted in [BS07] that it is enough to consider D comprising of at most two odd primes, where each prime $\ell < 2 \log(N)$.

In the case of a Fibonacci prime f_q , note that $\left(\frac{\ell^*}{f_q}\right) = \left(\frac{\ell}{f_q}\right)$ due to the fact that $f_q \equiv 1 \pmod{4}$, by Lemma 11.9. Let P_q be the list of all primes $\ell < 2 \log(f_q)$ such that $\left(\frac{\ell}{f_q}\right) = 1$. We will describe an algorithm to construct a list S_q of good discriminants from P_D . Here, we say that D is a good discriminant if D is of the form $D = -\ell_0$ or $D = -\ell_1 \ell_2$, and $D \equiv 5 \pmod{8}$, where $\ell_0, \ell_1, \ell_2 \in P_q$.

Algorithm 5.2. Suppose that we are given a list P_q of odd primes $\ell < 2 \log(f_q)$ such that $\left(\frac{\ell}{f_q}\right) = 1$. Let N be the cardinality of P_q and assume that the primes ℓ are listed in increasing order. This algorithm create the list S_q of good discriminants.

1. Let S_q be an empty list.
2. Let $k = 0$.
3. If $k = N$, end the algorithm, else let $D = -P_q[k]$.
4. If $D = -P_q[k] \equiv 5 \pmod{8}$, append D to S_q .
5. For $m = 0, \dots, k - 1$, if $D = -P_q[m] \cdot P_q[k] \equiv 5 \pmod{8}$, append D to S_q .
6. Increase k by 1 and return to Step 3.

This efficient construction of D provides a degree of control of the class number. For security reason, we do not want D to be too small as ECC can be attacked via using an isogenous curve. To ensure that D is not too small, we could use the well known fact that the class number of $C(D)$ is approximately $\sqrt{-D}$ by Brauer-Siegel Theorem. Furthermore, the work of Goldfeld, Gross, Zagier and Osterle in the 1980s provide an easily computable lower bound (see [Zag84] and [Cox13, pp. 135]):

$$h(D) > \frac{1}{K} \log(-D) \prod_{p|D}^* \left(1 - \frac{2\sqrt{p}}{p+1}\right), \quad (5.5)$$

where $K = 55$ if $\gcd(D, 5077) = 1$ and $K = 7000$ otherwise, and the product is taken over all prime divisors p of D except the largest prime.

Now we will approximate a lower bound for the expected number of such D following [BS08], but we will provide a bit more details. Recall that in the Algorithm 2.1, we loop through the good discriminants $D \in S_q$ until we can find one such that $4f_q = x^2 + y^2|D|$ and $f_q + 1 \pm x$ is a prime for some positive integers x, y . As mentioned in Theorem 2.2, we should expect to find $O(\log(f_q))$ many D by the Chebotarev Density Theorem, which follows from Lemma 5.6 with the bound $B = O(\log^2(f_q))$

Theorem 5.3. Let K_1, \dots, K_n be distinct imaginary quadratic fields of odd class number. Let H_1, \dots, H_n be ring class fields of K_1, \dots, K_n respectively, such that $[H_i : K_i] = n_i$ are all odd. Let $H = \prod_{i=2}^n H_i$. Then $H_1 \cap H = \mathbb{Q}$.

Proof. See [DLR15, Theorem 4.4]. □

Lemma 5.4. The order of the class group $C(D)$ is odd exactly for discriminants D of the form $D = -q$, where $q \equiv 3 \pmod{4}$.

Proof. Recall that Genus Theory states that the number of elements of order 2 in $C(D)$ is $2^{t-1} - 1$, where t is the number of odd prime divisors of D . The result follows. See also the argument preceding Proposition 11.11. □

Corollary 5.5. Let q_1, q_2, \dots, q_n be a set of primes such that $q_i \equiv 3 \pmod{4}$ for $i = 1, \dots, n$. Let H_{q_1}, \dots, H_{q_n} be the Hilbert class fields of the imaginary

quadratic fields $\mathbb{Q}(\sqrt{-q_1}), \dots, \mathbb{Q}(\sqrt{-q_n})$, respectively. Let $H = \prod_{k=2}^n H_{q_k}$. Then $H_{q_1} \cap H = \mathbb{Q}$.

Corollary 5.5 can be proven quickly by looking at the ramified primes, as noticed in [BS07]. Indeed, note that q_k is the only rational prime that ramified in H_{q_k} , and q_k does not ramify in $\prod_{m \neq k} H_{q_m}$. Hence the intersection must be \mathbb{Q} . It follows that the class fields H_{q_k} are linearly independent over \mathbb{Q} .

Lemma 5.6. Let N be a rational prime and let $P(B) = \{\ell : \ell \text{ is prime and } \ell \leq B\}$. The number $S(B)$ of primes $\ell \in P(B)$ such that $4N = x^2 + \ell y^2$ for some integers x, y is approximately $\sqrt{B}/\log(B)$.

Proof. Chebotarev Density tells us that $P(B)$ is of size $O(B/(2 \log(B)))$. Recall that $4N = x^2 + \ell y^2$ for some integers x, y if and only if N splits completely in the Hilbert class field H_ℓ of $K_\ell = \mathbb{Q}(\sqrt{-\ell})$ if and only if N is the norm of some principal element of \mathcal{O}_ℓ , the ring of integers of K_ℓ . By the Chebotarev Density Theorem, the density of rational primes that split in H_ℓ is $1/2h_\ell$, where h_ℓ is the class number of \mathcal{O}_ℓ , the ring of integers of K_ℓ . If N splits completely in H_ℓ , then there exist two elements in $\alpha, \beta \in \mathcal{O}_\ell$ such that their norm is equal to N . It follows that the expected number of elements in \mathcal{O}_ℓ for which N is the norm of is $1/h_\ell \approx 1/\sqrt{\ell}$. Hence, since the class fields H_ℓ are linearly disjoint over \mathbb{Q} , the expected number of primes ℓ for which N split completely in H_ℓ is approximately

$$\sum_{\ell \in P(B)} \frac{1}{\sqrt{\ell}} \approx \left(\frac{B}{\log(B)} \right) \frac{1}{\sqrt{B}} = \frac{\sqrt{B}}{\log(B)}. \quad (5.6)$$

□

6. ISOGENY VOLCANOES

To compute the Hilbert class polynomial $H_D(X) \pmod{p}$, it is best avoid the complex-analytic method when $|D| > 10^{10}$, as this is the practical upper limit due to storage size (see [Sut12a]). In general, all methods in computing $H_D(X) \pmod{p}$ has time complexity $\tilde{O}(|D|)$, with the difference being the storage required for each method. Here the discriminants in Algorithm 2.1 is of size $O(\log^2(f_q))$, so the time complexity is at most $\tilde{O}(\log^2(f_q))$ each time the CRT method is called. Now in the case that $C(D)$ is composite such as the case when $D = -\ell_1 \ell_2$, a root of $H_D(X) \pmod{q}$ can be obtained directly without even knowing their coefficients (see [Sut12a]). As each discriminant D in Algorithm 2.1 is either of the form $D = -\ell_0$ or $D = -\ell_1 \ell_2$, the class group $C(D)$ is highly cyclic, so we may only have walk around the surface of one ℓ -volcano. Furthermore, each such discriminant D is fundamental so we do not have to worry about explicitly computing the endomorphism ring of elliptic curves as outlined in Algorithm 1.2 of [Sut11]. It is expected that the CRT method to be faster in our scenario.

We will now provide a quick overview of the method of computing $H_D(X) \pmod{q}$ using Chinese Remainder Theorem (CRT) following [Sut11]. By the Chebotarev Density Theorem, given a negative discriminant D the set of primes

$$\mathcal{P}_D = \{\eta > 3 \text{ prime} : 4\eta = t_\eta^2 + v_\eta^2 |D| \text{ for some } t_\eta, v_\eta \in \mathbb{Z}^+\} \quad (6.1)$$

is infinite and of density $1/2h_D$, where h_D is the order of the class group $C(D)$. Suppose that we wish to compute $H_D(X) \pmod{p}$ for some (very large) prime p .

The CRT method is to compute $H_D(X) \pmod{\eta}$ for an optimized finite set $S(D)$ of primes $\eta \in \mathcal{P}$ so that

$$\prod_{\eta \in S(D)} \eta > 2B, \quad (6.2)$$

where B is an upper bound for the coefficients of $H_D(X)$. By the CRT, we can explicitly determine $H_D(X) \pmod{p}$. Now we will describe how to find $H_D(X) \pmod{\eta}$ for $\eta \in S(D)$.

Let K be the imaginary quadratic field $K = \mathbb{Q}(\sqrt{D})$, and suppose $4\eta = t^2 + v^2|D|$ for some positive integers t, v . By Complex Multiplication, each root r of $H_D(X) \pmod{\eta}$ corresponds to an isomorphism class of elliptic curves with endomorphism ring isomorphic to \mathcal{O}_K . Furthermore, each curve has trace $\pm t$.

Let $\text{Ell}_t(\mathbb{F}_\eta)$ be the set of the j -invariants of all elliptic curves over \mathbb{F}_η with trace equals to t . Hence, each element j_0 of $\text{Ell}_t(\mathbb{F}_\eta)$ represents the class of E/\mathbb{F}_η and its twists, where E is a curve with j -invariant j_0 . Let $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ be the set of all roots of $H_D(X) \pmod{p}$, which correspond to the j -invariant of all elliptic curves over \mathbb{F}_η with endomorphism ring equals to \mathcal{O} . We have the following set inclusions

$$\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta) \subset \text{Ell}_t(\mathbb{F}_\eta) \subset \mathbb{F}_\eta. \quad (6.3)$$

A key observation is that the set $\text{Ell}_t(\mathbb{F}_\eta)$ consists of isogenous curves as they have trace t over the same finite field. Hence, given $j(E) \in \text{Ell}_t(\mathbb{F}_\eta)$, it is discovered that there exists an efficient method in obtaining an isogenous curve E' such that $j(E') \in \text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$, the foundation of which is based on Kohel's thesis [Koh96].

Kohel's work describes a method to explicitly compute the endomorphism ring of an ordinary elliptic curve E over finite field, which is isomorphic to an order \mathcal{O} of some imaginary quadratic field K . We have the following containment:

$$\mathbb{Z}[\pi_E] \subset \mathcal{O} \subset \mathcal{O}_K. \quad (6.4)$$

Let $u = [\mathcal{O}_K : \mathcal{O}]$ and let $v = [\mathcal{O} : \mathbb{Z}[\pi_E]]$. The index $w = [\mathcal{O}_K : \mathbb{Z}[\pi_E]]$ is equal to uv . Let ν_ℓ be the standard ℓ -adic valuation. Kohel's work [Koh96] shows that computing the endomorphism ring of E is equivalent to know the $\nu_\ell(w)$ for various primes ℓ (See [Sut11, Proposition 2]).

Let $\ell \neq \eta$ be a prime, and let $\Gamma_{\ell,t}(\mathbb{F}_\eta)$ be the undirected graph with $V = \text{Ell}_t(\mathbb{F}_\eta)$ as vertices. There is an edge between $j(E), j(E') \in V$ exactly when $\varphi_\ell(j(E), j(E')) = 0$, where $\varphi_\ell(x, y)$ is the well-known classical modular polynomial (see [Cox13, ch. 11]). The equation $\varphi_\ell(j(E), j(E')) = 0$ is satisfied exactly when there is an isogeny of degree ℓ between E and E' . These modular polynomials $\Phi_\ell(X, Y)$ are precomputed in Step 10. since Algorithm 6.1 are to be called for $\mathcal{O}(\log(f_q))$ many discriminants, and they will be reused later in Step 18. and Step 19. of Algorithm 2.1.

With at most two exceptions, the components of $\Gamma_{\ell,t}(\mathbb{F}_\eta)$ are ℓ -volcanoes (see [Sut11] for definitions), but since we have excluded $j = 0, 1728$ in Algorithm 2.1 those exceptions do not occur. The graph resembles a volcano as can be seen in Figure 1. Each ℓ -volcano can be partitioned into levels V_0, \dots, V_d , where each level of the volcano represents elliptic curves with the same endomorphism ring, and the depth each ℓ -volcano is $d = \nu_\ell(w)$. The bottom (floor) of the volcano contain curves with endomorphism ring generated by the Frobenius automorphism, while the top (surface) contains curves with the full ring of integers \mathcal{O}_K as their endomorphism ring, where $K = \mathbb{Q}(\sqrt{D})$.

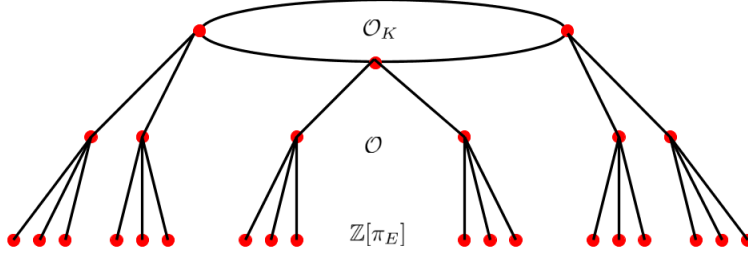


FIGURE 1. A 3-volcano of depth 2, with a 3-cycle on the surface.

The idea is to start with a random curve with trace t , which is actually very difficult to find. As suggested by Sutherland, it is best to use the idea of picking points from modular curves $X_1(m)$ (see Section 7). Once we have a $j_0 \in \text{Ell}_t(\mathbb{F}_\eta)$, we go to the ℓ -volcano that contains j_0 and we replace j_0 with the j -invariant at the level $\nu_\ell(w)$. If we perform this for each $\ell \mid w$, then by Proposition 2 of [Sut11] the final $j_0 \in \text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$, as desired. We then choose variety of primes $\ell \neq p$ so that $\ell \nmid u$, in which case j_0 is on the top of each ℓ -volcano. Finally, we use the action of $C(D)$ on j_0 to obtain all the other elements of $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ by walking on the surface of these ℓ -volcanoes.

The action of $C(D)$ on $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ is a very interesting aspect of the algorithm. Let $\ell \neq p$ be a prime such that $(\frac{D}{\ell}) \neq -1$, which we will classify as an Elkies prime in Section 10. There exists a prime ideal \mathfrak{a} of \mathcal{O} such that $(\ell) = \mathfrak{a}\bar{\mathfrak{a}}$, that is, ℓ is the norm of \mathfrak{a} . There is a *prime form* (ℓ, b_ℓ, c_ℓ) of discriminant D (see Section 8.2) corresponding to the ideal \mathfrak{a} such that ℓ is the norm of the class $[(\ell, b_\ell, c_\ell)]$ of $C(D)$ represented by (ℓ, b_ℓ, c_ℓ) . The order $\text{ord}_D(\ell)$ of $[(\ell, b_\ell, c_\ell)]$ in $C(D)$ is equal to the number of elements of $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ that lie on the surface of the ℓ -volcano (see [Sut11, Proposition 3]).

Suppose $j_0 \in \text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ lies on the surface V_0 of an ℓ -volcano. One walks a path of length d to obtain a list of elements $[j_0, j_1, \dots, j_d]$ of $\text{Ell}_t(\mathbb{F}_\eta)$. If the level of $j_d \notin V_d$, then $j_2 \in \text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$, else we try another path of length d . We perform this walking on each subsequent found $j \in \text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ until the number of elements found is equal to $\text{ord}_D(\ell)$. In general, to obtain elements on the surface, we use primes ℓ so that the depth d is 0, that is, $\ell \nmid v$. We use primes $\ell \mid v$ only when it is easier to find roots of $\Phi_\ell(X, j(E))$. Hence, we observe that most of the orbit computations are done on ℓ -volcanoes of depth 0.

Note that when the depth $d = 0$, walking around the surface V_0 is relatively straightforward from Proposition 6.2 of [Sch95] regarding Elkies prime. If $(\frac{D}{\ell}) = 0$, then $\text{ord}_D(\ell) = 2$. The polynomial $\Phi_\ell(X, j_0)$ has exactly one other root j_1 that lies on the surface V_0 . Hence, the surface V_0 in this case is just a line segment. Now when $(\frac{D}{\ell}) = 1$, the polynomial $\Phi_\ell(X, j_{i-1})/(X - j_i)$ has exactly one root j_{i+1} . Of course, here the surface V_0 is a cycle of length $\text{ord}_D(\ell)$.

There are many considerations must be taken for this method to be efficient. The primes η in $S(D)$ must be chosen carefully so that the density of curves with t is high, that is, we choose primes η that enlarge $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ while keeping the size of $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ the same. We also want primes η so that the index $v = [\mathcal{O} : \mathbb{Z}[\pi_E]]$ has small prime divisors as the depth d of each ℓ -volcano depends entirely on v

when D is fundamental. The primes ℓ must be chosen so that movements on the ℓ -volcanoes are easiest. The action of $C(D)$ on $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$ is easier to compute if we use the prime forms of $C(D)$ (see Section 8.2). Finally, the computations for the CRT must be updated continuously once $H_D(X) \pmod{\eta}$ is obtained for each $\eta \in S(D)$.

Algorithm 6.1. Let D be a fundamental discriminant of size $O(\log^2(f_q))$ from Step 8. from Algorithm 2.1. This algorithm finds $H_D(X) \pmod{p}$ using Chinese Remainder Theorem.

1. Let $\mathcal{P}_D = \{\eta > 3 \text{ prime} : 4\eta = t_\eta^2 + v_\eta^2 | D| \text{ for some } t_\eta, v_\eta \in \mathbb{Z}\}$.
2. Choose an optimized list $S(D)$ of primes from \mathcal{P}_D .
3. Let $k = 0$.
4. Let $\eta = S[k]$.
5. Find a curve E with $j(E) \in \text{Ell}_t(\mathbb{F}_\eta)$ following Section 7.
6. Use the precomputed modular polynomials $\Phi_\ell(X, Y)$ to find an isogenous E' such that $j_0 = j(E') \in \text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$.
7. Compute the prime forms for discriminant D using Algorithm 8.2.
8. Use the precomputed modular polynomials $\Phi_\ell(X, Y)$ and prime forms to compute the orbit of the group action of $C(D)$ on j_0 to obtain all the elements of $\text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)$.
9. Compute $H_D(X) \pmod{\eta}$ by expanding

$$H_D(X) \pmod{p} = \prod_{j \in \text{Ell}_{\mathcal{O}}(\mathbb{F}_\eta)} (X - j) \pmod{\eta}. \quad (6.5)$$

10. Increase k by 1 and return to Step 4 if $k < |C(D)|$, else go to the next step.
11. Use CRT to compute $H_D(X) \pmod{p}$.

Note that precomputing $\Phi_\ell(X, Y)$ is negligible as each prime $\ell < 6 \log^2(4 \log^2(f_q))$, and the time complexity for computing each $\Phi_\ell(X, Y)$ is $\tilde{O}(\log^3(\ell))$.

7. SAMPLING FROM MODULAR CURVE

Recall that if we wish to construct an elliptic curve E/\mathbb{F}_p of order N , then we need to find one with trace $t = p + 1 - N$. The naive method in finding a curve E of trace t is to look for $E : Y^2 = X^3 + aX - a$, where $1 \leq a \leq p - 1$, such that

$$(p + 1)(1, 1) = \pm t(1, 1), \quad (7.1)$$

as the point $(1, 1)$ on E (though E may not have trace t). Then we compute the order of E and find its twist if necessary. However, this naive method requires the computation of the order of around $2\sqrt{p}$ curves. To accelerate the search we need to reduce our sample size, and in [Sut12b] Sutherland does this by searching for points on the modular curve $X_1(d)$ for various $d \mid N$.

Recall that by Mazur's theorem (see [Sil09, Theorem 7.5]), if E is an elliptic curve over \mathbb{Q} , then the order of a non-trivial torsion point P of E is a number in the set $\mathcal{T} = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 12\}$. Let p be an odd prime that does not divide the discriminant of E , that is, p is a prime of good reduction, then the reduction map

$$E_{\text{torsion}} \rightarrow \overline{E}(\mathbb{F}_p) \quad (7.2)$$

is injective (see page 123 of [ST92]). This provides a way to obtain a curve over \mathbb{F}_p with order divisible by $d \in \mathcal{T}$. However, the possible orders of \overline{E} is very limited if E

is defined over \mathbb{Q} . We could try to find a curve E over some finite extension K/\mathbb{Q} ; however, its reduction may not have coefficients in \mathbb{F}_p . For example, the reduction of $E/\mathbb{Q}(d)$ has coefficients in \mathbb{F}_p exactly when $d \pmod{p}$ is a square. As Sutherland suggests in [Sut12b], we should look for points on the curve $Y_1(d)/\mathbb{F}_p$, where $Y_1(d)$ is the affine subcurve of $X_1(d)$.

The modular curve $X_1(m)$ classifies pairs (E, P) of elliptic curves E with a fixed point $P \in E(\mathbb{C})$ of order m up to isomorphism over \mathbb{C} (see [LR11]). To find an elliptic curve of order m , we narrow our search to within $Y_1(d)$ for some $d \mid m$. We want d to be reasonably small due to the cost in finding such points. Furthermore, if $d_1 < d_2$ are small divisors of m , we generate several curves from $X_1(d_1)$ and test them for d_2 -torsion.

To find a curve E/\mathbb{F}_p of trace t , we search for points on $Y_1(d)/\mathbb{F}_p$, where $d \mid p+1-t$ or $d \mid p+1+t$, preferably both. From the given point, we can find its Weierstrass equation, and if the curve is singular, we find another point. In [Sut12b], Sutherland has optimized the search for points on $Y_1(d)/\mathbb{F}_p$ by computing the *raw form* $F_d(r, s)$ of $X_1(d)$. Sutherland and Hoeij has computed $F_d(r, s)$ for d up to 100. It is recommended in [Sut12b] to use such forms for only d up to 40 due to the cost of finding points of $F_d(r, s) = 0$.

In theory, one could directly apply this searching method to construct elliptic curves of prime order N over some prime field \mathbb{F}_p , provided that $|t| = |p+1-N| \leq 2\sqrt{p}$. We will describe this algorithm below; however, it is impractical for large prime p .

To find an elliptic curve over $\mathbb{Z}/p\mathbb{Z}$ of trace $t = p+1-N$, we find a small divisor d of $N_0 = p+1+t$ and search within $Y_1(d)/(\mathbb{Z}/p\mathbb{Z})$. Once we find a curve of trace t , we compute its twist. Of course, this method fails when N_0 is not smooth.

For each of the probable prime p found from Step 9. and verified in Step 12., we can apply the following algorithm to construct an elliptic curve $E/(\mathbb{Z}/p\mathbb{Z})$ of order f_q . Note by construction $|t| = |p+1-f_q| \leq 2\sqrt{p}$ is automatically satisfied.

Algorithm 7.1. Algorithm to find an elliptic curve $E/(\mathbb{Z}/p\mathbb{Z})$ of trace $t = p+1-f_q$.

1. Compute $N_0 = 2(p+1) - f_q$.
2. Find a small divisor d of N_0 .
3. Search for points on $Y_1(d)/(\mathbb{Z}/p\mathbb{Z})$ using the optimized planar equation $F_d(r, s) = 0 \pmod{p}$.
4. Find a Weierstrass equation for $E/(\mathbb{Z}/p\mathbb{Z})$.
5. If E is singular, return to Step 3. and find another point, else go to the next step.
6. Find a twist of E with order f_q if necessary.

8. PRIME FORMS OF CLASS GROUP

Recall that for a negative discriminant D , the elements of $C(D)$ consists of equivalence classes of binary quadratic forms $q(X, Y) = aX^2 + bXY + cY^2$ such that $D = b^2 - 4ac$. Moreover, each class is represented by exactly one reduced form. We wish to find generators for $C(D)$ where a is a rational prime.

Let ℓ be a rational prime. Note that if $D = b^2 - 4\ell c$, then D is a quadratic residue modulo 4ℓ . From this observation, we can easily obtain quadratic forms of $C(D)$ with a prime (see [BV07] and page 251 of [Coh93]). Indeed, assume D is a quadratic residue modulo ℓ , and write b_ℓ for its square root modulo ℓ . By taking $\ell - b_\ell$ if necessary, we may assume that b_ℓ is a square root of D modulo 4ℓ . Then

the form $(\ell, b_\ell, (b_\ell^2 - D)/4\ell)$ has discriminant D . We call such a form to be a *prime form* of $C(D)$. In fact, every form F of $C(D)$ is a product of such forms:

Lemma 8.1. (Lemma 5.5.1 of [Coh93]) Let (a, b, c) be a primitive positive definite quadratic form of discriminant $D < 0$, and $a = \prod_\ell \ell^{\nu_\ell}$ be the prime factorization of a . Then we have up to equivalence:

$$(a, b, c) = \prod_\ell F_\ell^{\epsilon_\ell \nu_\ell}, \quad (8.1)$$

where F_ℓ is a prime form corresponding to ℓ , and $\epsilon_\ell = \pm 1$ is defined by the congruence

$$b \equiv \epsilon_\ell b_\ell \pmod{2\ell}. \quad (8.2)$$

Assuming the Extended Riemann Hypothesis (ERH), restricting ℓ to be primes $\ell \leq 6 \log^2(|D|)$ yields a sequence of generators for $C(D)$ (see [Bac90]). It is suggested in [Coh93] that in practice it is better to search for primes $\ell \leq B(D)$, where

$$B(D) = \max\left(6(\log(|D|))^2, L(|D|)^{1/\sqrt{8}}\right) \quad (8.3)$$

and

$$L(x) = e^{\sqrt{\log(x) \log(\log(x))}}. \quad (8.4)$$

Algorithm 8.2. Algorithm to find prime forms of $C(D)$

1. Let R and \mathcal{F} be empty sets.
2. For each prime odd prime $\ell \leq B(D)$ such that $\left(\frac{D}{\ell}\right) = 1$, find a square root b_ℓ of D modulo ℓ . Take $b_\ell = \ell - b_\ell$ if necessary, we may assume that $b_\ell^2 \equiv D \pmod{4\ell}$, and we store the pair (b_ℓ, ℓ) into the set R .
3. For each pair $(b_\ell, \ell) \in R$, store the form (a_ℓ, b_ℓ, c_ℓ) into \mathcal{F} , where $c_\ell = (b_\ell^2 - D)/(4\ell)$.
4. Return \mathcal{F} .

As mentioned in Section 6, the prime forms for $C(D)$ are used in [Sut11] to create a polycyclic presentation of $C(D)$. These forms provide efficient walks on isogeny volcanoes, and the points on the surface of the volcanoes provide roots of $H_D(X) \pmod{q}$ for some prime q . As suggested by Sutherland in [Sut11], to make sure these prime forms generate $C(D)$ unconditionally, one computes the order of $C(D)$ and add more forms if necessary; however, this is not practical for large discriminants such as the ones in our scenario.

9. COMPUTING SQUARE ROOT AND CORNACCHIA'S ALGORITHM

There are various algorithms to find a square root of an integer a modulo prime p , assuming that there exists one. In the easiest case $p \equiv 3 \pmod{4}$, the square root of a modulo p is given by $x = a^{(p+1)/4} \pmod{p}$. The remaining cases are $p \equiv 5 \pmod{8}$ and $p \equiv 1 \pmod{8}$. For the case $p \equiv 1 \pmod{8}$, we will use Tonelli-Shanks algorithm, which is useful when computing multiple square roots. Finally, the case $p \equiv 5 \pmod{8}$ is relatively straightforward. If $p \equiv 5 \pmod{8}$ and a is a square modulo p , then

$$\sqrt{a} \pmod{p} = \begin{cases} a^{(p+3)/8} \pmod{p}, & \text{if } a^{(p-1)/4} \equiv 1 \pmod{p} \\ 2a(4a)^{(p-5)/8} \pmod{p}, & \text{if } a^{(p-1)/4} \equiv -1 \pmod{8}. \end{cases} \quad (9.1)$$

We will now describe the Tonelli-Shanks algorithm, following [Coh93]. Suppose we wish to compute the square root of a modulo p . Here, we are assuming that $\left(\frac{a}{p}\right) = 1$. Write

$$p - 1 = 2^e m, \quad (9.2)$$

where m is odd. The multiplicative group $(\mathbb{Z}/p\mathbb{Z})^*$ is cyclic of order $p - 1$, and so its 2-Sylow subgroup G is cyclic of order 2^e . Find an integer n such that $\left(\frac{n}{p}\right) = -1$, and compute $g = n^m$, which clearly belongs to G . If $g^{2^{e-1}} = 1 \pmod{p}$, then p is composite, else g generates the 2-Sylow subgroup G . Similarly, the element $a^m \in G$, so $a^m g^k = 1 \pmod{p}$ for some integer k . It follows that a square root of $a \pmod{p}$ is given by

$$\sqrt{a} \pmod{p} = a^{(m+1)/2} g^{k/2} \pmod{p}. \quad (9.3)$$

It would be useful to compute the square roots of all the elements in the 2-Sylow if e is small and multiple square roots are to be computed. This is especially useful for Algorithm 2.1. Furthermore, one amazing fact is that for each of the probable Fibonacci prime f_q from [onl17], the 2-Sylow subgroup of $(\mathbb{Z}/f_q\mathbb{Z})^\times$ is very small (see Observation 11.10). Hence, computing square roots modulo f_q is relatively simple in all cases.

Below is a variant of the Tonelli-Shanks algorithm that is tailored for our purpose. We will break the Tonelli-Shanks algorithm into two parts to suit our purpose. If either part fails, then f_q is not a prime.

Algorithm 9.1. (Precomputations) This algorithm computes a square root of each element of the 2-Sylow subgroup G of $(\mathbb{Z}/f_q\mathbb{Z})^\times$. Suppose we are given a quadratic non-residue n modulo f_q .

1. Let $R(2, f_q)$ be an empty list.
2. Factor $f_q - 1$ as $f_q - 1 = 2^e m$, where m is odd.
3. Compute $g = n^d \pmod{f_q}$.
4. Let G be the subgroup generated by g .
5. Compute a square root r_i of each element g^i of G and append the ordered pair (g^i, r_i) to the list $R(2, f_q)$.

Here we are applying the standard Tonelli-Shanks algorithm repeatedly in the final step of Algorithm 9.1. Of course, for some of the elements we do not have to. Now we will describe the Tonelli-Shanks algorithm if we are given $R(2, f_q)$.

Algorithm 9.2. (Variant of Tonelli-Shanks algorithm) Given $R(2, f_q)$ and $\left(\frac{a}{f_q}\right) = 1$, this algorithm computes a root of $a \pmod{f_q}$.

1. Compute $x = a^{(m+1)/2} \pmod{f_q}$ and $y = a^m \pmod{f_q}$.
2. Look up $R(2, f_q)$ for a square root z of y .
3. Then $\sqrt{a} = \pm x/z \pmod{f_q}$.

It is noted in [AM93] that if we fail to find a square root modulo p using the above algorithms, then p is composite. It is unlikely for a composite number to pass Algorithm 9.2. See [Wil87] for a combination of square roots with primality tests.

An efficient algorithm to obtain a square root is useful in solving a variety of Diophantine equations, namely the equation $X^2 + dY^2 = m$. Cornacchia's algorithm (see Section 1.5.2 of [Coh93] and [Sch95]) can provide the unique solution of

positive integers to the equation $X^2 + dY^2 = m$ if there exists one. The algorithm is essentially an application of the Euclidean algorithm. We will provide an overview of the algorithm following [Sch95], where its proof by Lenstra can also be seen. In [Sch95], the Cornacchia's algorithm is used to compute the order of an elliptic curve E/\mathbb{F}_p if its endomorphism ring is known, and he also reversed this process to compute square roots modulo p (see [Sch85]).

If $\left(\frac{-d}{m}\right) = -1$, then clearly the equation $X^2 + dY^2 = m$ does not have a solution. Assume $\left(\frac{-d}{m}\right) = 1$, and let r_0 be a root of $-d \pmod{m}$. Replacing r_0 with $m - r_0$ if necessary, we may assume that $r_0 \leq m/2$. Let $r_{-1} = m$. Use the Euclidean algorithm to find a sequence of non-negative integers r_1, r_2, \dots, r_k such that $r_k < \sqrt{m}$ and

$$r_j \equiv r_{j-2} \pmod{r_{j-1}}, \quad \text{for } 1 \leq j \leq k. \quad (9.4)$$

Then a unique solution of positive integers exists exactly when the real number

$$\sqrt{\frac{m - r_k^2}{d}} \quad (9.5)$$

is an integer, and the solution is given explicitly by

$$(x, y) = \left(r_k, \sqrt{\frac{m - r_k^2}{d}} \right). \quad (9.6)$$

Algorithm 9.3. (Cornacchia's Algorithm) Algorithm to find a solution to $x^2 + dy^2 = m$ if there exists one.

1. Find a square root r_0 of $-d \pmod{m}$.
2. Replace r_0 with $m - r_0$ if necessary, we may assume $r_0 \leq m/2$.
3. Let $r_{-1} = m$.
4. Use the Euclidean algorithm to find a sequence of non-negative integers r_1, r_2, \dots, r_k satisfying $r_k < \sqrt{m}$ and $r_j \equiv r_{j-2} \pmod{r_{j-1}}$, for $1 \leq j \leq k$.
5. Compute $s = (m - r_k^2)/d$.
6. A solution exists exactly when s is an integer squared, and the unique solution of positive integers is given by $(x, y) = (r_k, \sqrt{s})$.

Note that each square root computation $\sqrt{a} \pmod{f_q}$ is just applications of a small number of exponentiations, which has time complexity $\tilde{O}(\log^2(f_q))$. The Cornacchia's Algorithm is just an application of Euclidean Algorithm, which has time complexity $\tilde{O}(\log(f_q))$. Hence, the time complexity for all algorithms in this section has time complexity of at most $\tilde{O}(\log^2(f_q))$.

10. SCHOOF'S ALGORITHM

Let E/\mathbb{F}_p be an ordinary elliptic curve over \mathbb{F}_p of the form $Y^2 - f(X) = 0$ for some cubic polynomial $f(X)$, whose j -invariant is neither 0 or 1728. Recall that the order N and trace t of E are related by $N = p + 1 - t$. Hence, computing the order E is equivalent to computing the trace of E . Schoof's algorithm [Sch85] can compute t in time complexity $O(\log^8(p))$. Improvements by Atkin and Elkies [Sch95] have lead to the SEA algorithm, which has time complexity $\tilde{O}(\log^4(p))$.

Schoof's idea is to compute $t \pmod{\ell}$ for finitely many primes $\ell \neq p$ and compute t using the Chinese Remainder Theorem. Since $|t| \leq 2\sqrt{p}$, we want to choose

the set $S(E)$ of primes $\ell < p$ so that

$$\prod_{\ell \in S(E)} \ell > 4\sqrt{p}, \quad (10.1)$$

in order to be able to determine t uniquely.

To compute $t \pmod{\ell}$, we use the characteristic polynomial

$$X^2 - tX + p = 0 \quad (10.2)$$

of the Frobenius automorphism π_E of E . Let $P = (x, y)$ be a point of the ℓ -torsion subgroup $E[\ell]$ of E . From Equation 10.2, we have

$$(\pi_E^2 - t\pi_E + p)P = 0, \quad (10.3)$$

and explicitly in terms of x and y , we have

$$(x^{p^2}, y^{p^2}) - t(x^p, y^p) + p(x, y) = 0, \quad (10.4)$$

which implies

$$(x^{p^2}, y^{p^2}) + p_\ell(x, y) = t_\ell(x^p, y^p), \quad (10.5)$$

where $t_\ell \equiv t \pmod{\ell}$ and $p_\ell \equiv p \pmod{p}$. Here, the 0 acts as both the point at infinity and the morphism induced by 0. Its meaning can be understood by context.

Schoof's idea to find $t \pmod{\ell}$ is to plug $t_\ell = 1, \dots, \ell - 1$ into Equation 10.5 until the equation is satisfied. In practice, we actually only have to look at $1 \leq t_\ell \leq (\ell - 1)/2$ by looking at the second coordinate. However, instead of working with one ℓ -torsion point at a time, we work with the entire ℓ -torsion group $E[\ell]$ at once, that is, we perform computations in the ring

$$R_\ell = \mathbb{F}_p[X, Y]/(\varphi_\ell(X), Y^2 - f(X)), \quad (10.6)$$

where $\varphi_\ell(X)$ is the ℓ -division polynomial of E (see [Sil09, pp. 105]). The degree of $\varphi_\ell(X)$ is of size $O(\ell^2)$, and each root of $\varphi_\ell(X)$ corresponds to the X -coordinate of some point in $E[\ell]$.

If we choose a prime ℓ of size $O(\log(f_q))$ such that $\left(\frac{t^2 - 4f_q}{\ell}\right) \neq -1$, then we can compute $t_\ell \pmod{\ell}$ much quicker. The prime ℓ is called an Elkies prime. The reason for this drastic reduction in time complexity is due to the fact that one can use a polynomial $F_\ell(X)$ of degree of size $O(\ell)$ in Equation 10.6, instead of the polynomial $\varphi_\ell(X)$ with degree of size $O(\ell^2)$, though the trace $t_\ell \pmod{\ell}$ is computed differently than as above.

Note that since the trace t is unknown, we can not immediately determine whether an integer is an Elkies prime. To determine whether ℓ is an Elkies prime, we such that fact that $\Phi_\ell(X, j(E))$ has a root exactly when ℓ is an Elkies prime (see [Sch95, Proposition 6.2]). Here $\Phi_\ell(X, Y)$ is the classical modular polynomial that parametrizes pairs of ℓ -isogenous elliptic curves over \mathbb{C} (see [Cox13, ch. 11]). This interpretation carries over to finite fields of characteristic prime to ℓ . The work [SS14] on the distribution of Atkin and Elkies primes shows that one should be able to find an Elkies prime quickly.

Since the polynomial $X^p - X$ splits completely over \mathbb{F}_p , the polynomial $\Phi_\ell(X, j(E)) \pmod{p}$ has a root j_0 in \mathbb{F}_p exactly when

$$\gcd(\Phi_\ell(X, j(E)), X^p - X) > 1. \quad (10.7)$$

As mentioned above, there exists an isogenous elliptic curve E' such that $j(E') = j_0$. By Proposition 6.1 of [Sch95], there exists a 1-dimensional subspace C of $E[\ell]$ such

that E' and E/C are \mathbb{F}_p^{alg} -isomorphic. Furthermore, the subspace C is an eigenspace of π_E for some eigenvalue λ , which corresponds to a root of $X^2 - tX + p = 0$. Hence, if the eigenvalue λ is known, the value $t_\ell = t \pmod{\ell}$ is easily computed.

Corresponding to the eigenspace C of π_E , there exists a polynomial $F_\ell(X)$ of degree $(\ell - 1)/2$, whose roots correspond to the distinct X -coordinates of the points in C . See [Sch95] for explicit calculation of the polynomial $F_\ell(X)$. Hence, to compute λ , we check which of the relations

$$\pi_E(X, Y) = (X^p, Y^p) = \lambda' \cdot (X, Y) \quad \lambda' = 1, \dots, \ell - 1 \pmod{F_\ell(X)}. \quad (10.8)$$

is satisfied. The polynomials here that we are working with have degrees of size $O(\ell)$ instead of $O(\ell^2)$ as in Schoof's algorithm. The time complexity in computing $t_\ell \pmod{\ell}$ is drastically reduced.

Schoof's algorithm and the observations above provide us a way to remove unwanted curves from the list \mathcal{E}_q that we obtain at the end of Step 16. of Algorithm 2.1. Each element of (E, p, D) is an ordinary curve with possible order f_q over the ring $\mathbb{Z}/p\mathbb{Z}$. The j -invariant of E is not equal to 0 or 1728 and is a root of $H_D(X) \pmod{p}$ by construction. We remove (E, p, D) from the list \mathcal{E}_q if $\Phi_\ell(X, j(E))$ does not have a root for prime ℓ such that $(\frac{D}{\ell}) \neq -1$. If (E, p, D) does pass this test, we verify that a root λ of $X^2 - tX + p = 0$ is an eigenvalue of the Frobenius map π_E .

Algorithm 10.1. (Elkies Primes Verification) Let \mathcal{E}_q be the list of curves obtained from Step 16. of Algorithm 2.1. Let N be the cardinality of \mathcal{E}_q .

1. Let $k = 0$ and let $E(D)$ be an empty list.
2. If $k = N$, then go to the final step, else let $(E, p, D) = \mathcal{E}_q[k]$.
3. Let $\ell = 2$.
4. Compute $r = (\frac{D}{\ell})$.
5. If $r \neq -1$, append ℓ to $E(D)$.
6. Compute $d = \gcd(\Phi_\ell(X, j(E)), X^p - X)$.
7. If $r \neq -1$ and $d = 1$ or $r = -1$ and $d > 1$, then remove (E, p, D) from \mathcal{E}_q and return to Step 2, else go to the next step.
8. Let ℓ be the next prime of ℓ . If $\ell < 2 \log^2(4 \log^2(f_q))$, then return to Step 4, else increase k by 1 and return to Step 2.
9. Output \mathcal{E}_q .

Algorithm 10.2. (Eigenvalue Verification) Let \mathcal{E}_q be the list output by Algorithm 10.1. Let N be the cardinality of \mathcal{E}_q .

1. Let $k = 0$.
2. If $k = N$, then go to the final step, else let $(E, p, D) = \mathcal{E}_q[k]$.
3. Let $\ell = E(D)[k]$.
4. Let λ, μ be the roots of $X^2 - tX + p = 0$.
5. Find polynomial f_ℓ given by

$$F_\ell(X) = \prod_{\pm P \in C} (X - P_x), \quad (10.9)$$

as defined above.

6. Let R_ℓ be the ring defined by

$$R_\ell = \mathbb{F}_p[X, Y]/(F_\ell(X), Y^2 - f(X)). \quad (10.10)$$

7. Verify if either $(X^p, Y^p) = \lambda(X, Y)$ or $(X^p, Y^p) = \mu(X, Y)$ in the ring R_ℓ . If either is satisfied, then increase k by 1 and return to Step 2. If neither is satisfied, then remove (E, p, D) from \mathcal{E}_q and return to Step 2.
8. Output \mathcal{E}_q .

Note the step in Algorithm 10.1 that has the highest time complexity is in Step 6, which has time complexity $\tilde{O}(\log(f_q))$. Since \mathcal{E}_q is of size $O(\log(f_q))$, the total time complexity of Algorithm 10.1 is $\tilde{O}(\log^2(f_q))$. For Algorithm 10.2, Step 6 requires the heaviest computations, and it has time complexity $\tilde{O}(\log^2(f_q))$. As the list \mathcal{E}_q is of size $O(\log(f_q))$, the total time complexity for Algorithm 10.2 is $\tilde{O}(\log^3(f_q))$.

11. SOME PROPERTIES OF FIBONACCI NUMBERS

Let $(f_n)_{n \geq 0}$ be the Fibonacci sequence given by $f_0 = 0, f_1 = 1$ and $f_{n+2} = f_{n+1} + f_n$ for all $n \geq 0$. If a Fibonacci number f is a rational prime, then we say that f is a Fibonacci prime. As mentioned in the introduction, it is not known whether there is an infinite number of Fibonacci primes (see [Cal17] for current commentary). One of the largest known Fibonacci prime is f_{81839} , which has 17103 digits. However, heuristics regarding elliptic divisibility sequence (EDS) from [EEW01] suggests it may be finite. Even though the sequence of Fibonacci numbers is not an EDS, with appropriate sign they are an EDS (see [SS06]). Unfortunately, we could not go too far in this path as the machinery is far beyond our capacity. We do find it to be interesting that Fibonacci primes can be written as a combination of an elliptic divisibility sequence (see Lemma 11.4). In the following we will discuss some well-known results regarding Fibonacci numbers.

Lemma 11.1. Let K be a field with characteristic not equal to 5. Let α, β be the roots of the polynomial $G(X) = X^2 - X - 1$. Then the n th Fibonacci number is given by

$$f_n = \frac{\alpha^n - \beta^n}{\sqrt{5}}. \quad (11.1)$$

Proof. This is an easy application of generating function. Here we will prove it directly. Let $g_n = (\alpha^n - \beta^n)/\sqrt{5}$. It's clear that $g_0 = 0, g_1 = 1$. Furthermore, since $\alpha^2 = \alpha + 1$, it follows that $\alpha^n = \alpha^{n-1} + \alpha^{n-2}$. Similarly, we have $\beta^n = \beta^{n-1} + \beta^{n-2}$. It is now straightforward to verify that $g_n = g_{n-1} + g_{n-2}$ for $n \geq 2$. Hence, $g_n = f_n$ for $n \geq 0$. \square

Lemma 11.2. (Cassini's Identity) For each integer $k \geq 0$, we have

$$f_{2k+1} = f_k^2 + f_{k+1}^2. \quad (11.2)$$

For Fibonacci prime f_q , the identity from Lemma 11.2 readily provides us a square root of $-1 \pmod{f_q}$. Indeed, if we have $f_q = f_{(q+1)/2}^2 + f_{(q-1)/2}^2$, then a square root of $-1 \pmod{f_q}$ is given by

$$\sqrt{-1} \pmod{f_q} = f_{(q+1)/2}/f_{(q-1)/2} \pmod{f_q}. \quad (11.3)$$

This observation allows us to quickly to test the primality of f_q as seen in Step 4. of Algorithm 2.1.

Theorem 11.3. (See [BGL15]) Let q be a rational prime. There exists integers u, v such that

$$f_q = u^2 + qv^2. \quad (11.4)$$

Lemma 11.4. (See Chapter 1 of [VM02]) For each integer $n \geq 1$, we have

$$\sum_{k=1}^n f_{2k} = f_{2n+1} - 1. \quad (11.5)$$

Lemma 11.5. The sequence $\{f_n \pmod{m}\}$ is periodic for any positive integer m .

Proof. This is clear by the Pigeonhole Principle. \square

In fact by [Wal60], for each prime ℓ , the period of the sequence $\{f_n \pmod{m}\}$ is the order of λ in the field $\mathbb{F}_\ell[X]/(X^2 - X - 1)$, where λ is a root of the polynomial $B(X) = X^2 - X - 1 \pmod{\ell}$. Using Quadratic Reciprocity, we have the following lemma:

Lemma 11.6. ([Wal60]) Let $\ell \neq 2, 5$ be a prime. Then $\pi(\ell)$ is a divisor of $\ell - 1$ if $\ell \equiv \pm 1 \pmod{10}$ and $\pi(\ell)$ is a quotient of $2(\ell + 1)$ by an odd divisor if $\ell \equiv \pm 3 \pmod{10}$.

Lemma 11.7. If $f_n \neq 3$ is a Fibonacci prime, then n is a prime.

Proof. This is due to the well-known fact that

$$\gcd(f_n, f_m) = f_{\gcd(n, m)}. \quad (11.6)$$

\square

Lemma 11.8. For each prime q , we have the divisibility properties:

$$q \mid f_{q - \left(\frac{5}{q}\right)}, \quad (11.7)$$

$$f_q \equiv \left(\frac{5}{q}\right) \pmod{q}. \quad (11.8)$$

Lemma 11.8 can be seen in [Wil82], but we can prove easily using basic Galois Theory. We will only prove 11.7 as 11.8 follows similarly. The result is clear if $q = 5$ by Lemma 11.5, so assume $q \neq 5$. Let $K = \mathbb{F}_q[X]/(X^2 - X - 1)$. Note that the discriminant of $X^2 - X - 1$ is 5. We will look at the cases $\left(\frac{5}{q}\right) = 1$ and $\left(\frac{5}{q}\right) = -1$ separately, that is, whether or not 5 splits in \mathbb{F}_q .

If 5 splits in \mathbb{F}_q , then K/\mathbb{F}_q is a degree 1 extension. Hence, the Frobenius automorphism $\text{Frob}_p : K \rightarrow K$ defined by $\text{Frob}_q(x) = x^q$ is trivial, which implies that $\text{Frob}_q(\alpha) = \alpha^q = \alpha$ and $\text{Frob}_q(\beta) = \beta^q = \beta$. It follows that $\text{Frob}_q(f_{q-1}) = f_{q-1} = (\alpha^q/\alpha - \beta^q/\beta)/\sqrt{5} = (1 - 1)/\sqrt{5} = 0$.

If 5 does not split in \mathbb{F}_q , then K/\mathbb{F}_q is a degree 2 extension, so the Frobenius automorphism is a conjugation map. It follows that $\text{Frob}_q(\alpha) = \alpha^q = \beta$. Hence, we have $f_{q+1} = (\alpha^q\alpha - \beta^q\beta)/\sqrt{5} = (\beta\alpha - \alpha\beta)/\sqrt{5} = 0$. The proof is complete.

Let $q > 3$ be a rational prime. Henceforth, let $C_q = C(-4f_q)$ and $B_q = B(-4f_q)$, as defined in Equation 8.3. Recall that in Algorithm 8.2, to find generators of the group C_q , we need to find the square roots $-f_q \pmod{\ell}$, for each prime $\ell \leq B_q$ such that $\left(\frac{-f_q}{\ell}\right) = 1$. If $\ell \equiv 3 \pmod{4}$, we find in Section 9 that this is an easy computation. Now for the case $\ell \equiv 1 \pmod{4}$, note that $-f_q \equiv -f_r \pmod{\ell}$, where r is the smallest non-negative integer such that $q \equiv r \pmod{\pi(\ell)}$. Hence, we do not need to compute f_q to find a square root of $-f_q \pmod{\ell}$. Furthermore, as the primes $\ell \leq B_q$ are relatively small, computations of the square roots of $-f_q$

(mod ℓ) are manageable. Noticeably, computing $\left(\frac{f_q}{\ell}\right)$ is easy for primes $\ell < q$ by the periodicity and Lemma 11.8.

Lemma 11.9. If $q > 3$ is prime, then $f_q \equiv 1 \pmod{4}$.

Proof. If $f_q \equiv 3 \pmod{4}$, then $q \equiv 4 \pmod{6}$ by Lemma 11.5, which can not happen as q is a prime. \square

Observation 11.10. We note that given a probable prime f_q , the 2-valuation of $f_q - 1$ is very small. If we write $f_q - 1 = 2^e m$, where m is odd, then $e \leq 6$. Here we have checked all the probable Fibonacci primes given at [onl17].

Let $K = \mathbb{Q}(\sqrt{-f_q})$. By Lemma 11.9, $-f_q \equiv 3 \pmod{4}$, so $D_K = -4f_q$. By Genus Theory (see [Cox13, Theorem 6.1]), the number of elements of $C(\mathcal{O}_K)$ of order 2 is $2^{t-1} - 1$, where t is the number of prime divisors of D_K . Since in our case $D_K = -4f_q$, we have $t - 1 \geq 1$. Hence, $C(\mathcal{O}_k) \cong C_q$ always has even order. It would be of interest to study the 2-sylow subgroup of C_q .

Proposition 11.11. (See also [CH88, Corollary 18.6]) Suppose f_q is a prime for $q > 3$. The 2-sylow subgroup of C_q is cyclic.

Proof. Suppose f_q is a Fibonacci prime. Then the number of prime divisors of $-4f_q$ is 2. Hence, C_q has $2^{2-1} - 1 = 1$ element of order 2. As C_q has exactly one element of order 2, it must be the case that the 2-sylow subgroup of C_q is cyclic. \square

Proposition 11.12. The probability that C_q is cyclic is at least 97%.

Proof. Heuristics from [CL84] by Cohen and Lenstra states (conjecturally) that the odd part of a class group $C(D)$ is cyclic for at least 97% of the time (see also Conjecture 5.10.1 of [Coh93]). Since the 2-sylow subgroup of C_q is always cyclic, we have the desired result. \square

Proposition 11.13. (See [CH88, Corollary 19.6]) Let q be a prime and $E = \mathbb{Q}(\sqrt{-q})$. Then the 2-Sylow subgroup of $C(D_E)$ has order 2 if and only if $q \equiv 5 \pmod{8}$.

Corollary 11.14. The 2-sylow subgroup of C_q has order 2 exactly when $q \equiv 5, 7 \pmod{12}$.

Proof. By Lemma 11.13, the 2-sylow of C_q has order 2 exactly when $f_q \equiv 5 \pmod{8}$. By Lemma 11.6, $f_q \equiv 5 \pmod{8}$ exactly when $p \equiv 5, 7, 8 \pmod{12}$, but $q \equiv 8 \pmod{12}$ can not happen as q is a prime. \square

Having a nontrivial 2-Sylow allows us to factor f_q if it is not prime using the Shank's Class Group Method. All of the elements of order 2 are of the form (a, a, c) , $(a, 0, c)$, and (a, b, a) . For example, if we have the form (a, a, c) , then we have $-4f_q = a^2 - 4ac = a(a - 4c)$, which implies $f_q = (a/2)(a/2 - 2c)$, though the factorization may be trivial. Obtaining an element of order 2 is relatively straightforward if we can easily compute the class number h of C_q . Indeed, factor h as $h = 2^e k$, where k is odd. Compute an arbitrary prime form F_ℓ of C_q using Algorithm 8.2, then the element F_ℓ^k is in the 2-Sylow subgroup of C_q . Of course, the difficulty lies in computing h , which is extremely difficult for large f_q . Since C_q is highly cyclic, we could in theory probabilistically compute its order using Atkin's Variant (see [Coh93, pp. 252–261]); however, its time complexity is sub-exponential.

Assume by some miracle that we are able to compute the class number of C_q , then we can easily obtain an element of order 2. As mentioned above, the class group has a high probability of being cyclic, which implies C_q has exactly one element of order 2 most of the times. Hence, each element of order 2 allows us a way to factor f_q , and even if multiple elements of order 2 yield trivial factorization, the fact that we have more than one element of order 2 hints of the fact that f_q may be composite.

We will now describe a variant of Shanks's Class Group Method, but we will not use it to test probable Fibonacci primes as it is not practical with known machinery. We will avoid this *factoring in the dark ages* (see [Coh93, ch. 8]) and provide much better primality tests in Section 13.

Algorithm 11.15. Variant of Shanks's Class Group Method

1. Find the set of prime generators \mathcal{F} for C_q using Algorithm 8.2.
2. Use Algorithm 8.2 to find prime forms of C_q and obtain elements of order 2. Also let n be the number of trials until the first non-quadratic residue z is found.
3. If $n \geq 50$, then f_q is likely composite.
4. If an element of order 2 is found, use it to factor f_q . If the factorization is nontrivial, then f_q is composite.
5. If two or more distinct elements of order 2 are found, then f_q is definitely not a prime for $q \equiv 5, 7 \pmod{12}$, otherwise f_q is likely composite.
6. Verify that $|\mathcal{F}| \approx B(D)/2$, else f_q is likely not prime.
7. If f_q passes the previous steps, then f_q is a probable prime.

12. EXCEPTIONAL CASES

We have from Lemma 11.2 that

$$4f_q = (2f_{(q+1)/2})^2 + 4f_{(q-1)/2}^2, \tag{12.1}$$

so f_q splits completely in the Hilbert class field of $\mathbb{Q}(\sqrt{-1})$, which is itself since the class number is 1. This is not all surprising since $\left(\frac{-1}{f_q}\right) = 1$. Generalizing this, we have that f_q splits completely in the Hilbert class field H_D of any imaginary quadratic field $\mathbb{Q}(\sqrt{D})$ with class number 1 for which $\left(\frac{D}{f_q}\right) = 1$. It is well-known that the set of all $d < 0$ such that the field $\mathbb{Q}(\sqrt{d})$ has class order 1 is the set of Heegner numbers

$$H = \{-1, -2, -3, -7, -11, -19, -43, -67, -143\}. \tag{12.2}$$

For example, if $f_q \equiv 1 \pmod{8}$, then

$$4f_q = (2x)^2 + 8y^2 \tag{12.3}$$

for some integers x, y . Step 8. skips over $d = -1, -2, -3$ and -7 . We will use these cases to quickly test the primality of f_q .

As mentioned in Section 11, computing $\left(\frac{\ell}{f_q}\right) = \left(\frac{\ell}{f_q}\right)$ can be computed quickly with primes $\ell < q$ due to the fact the sequence $(f_n \pmod{m})_{n \geq 0}$ is periodic. For example, $\left(\frac{3}{f_q}\right) = 1$ exactly when $f_q \equiv 1 \pmod{3}$, which can be quickly determined. Indeed, when $m = 3$, the period is 8 and $f_n \equiv 1$ exactly when $n \equiv 1, 2, 7 \pmod{8}$. However, the index q is prime, so it follows that $f_q \equiv 1 \pmod{3}$ exactly when

$q \equiv \pm 1 \pmod{8}$. Hence, $4f_q = x^2 + 3y^2$ for some positive integers x, y if and only if $q \equiv \pm 1 \pmod{8}$. Hence, for each $d \in H$, $\left(\frac{d}{f_q}\right)$ is sufficient for f_q to be a norm in $\mathbb{Q}(\sqrt{-d})$, and as observed above, $\left(\frac{d}{f_q}\right)$ is easily computed.

Again from Lemma 11.3, we have

$$f_q = x^2 + y^2q \quad (12.4)$$

for some positive integers x, y . Moreover, from Lemma 11.8, it follows that

$$x^2 \equiv \left(\frac{5}{q}\right) \pmod{q}. \quad (12.5)$$

Hence, we can take $x = 1$ or $x = \sqrt{-1} \pmod{q}$, and apply the Cornacchia's Algorithm 9.3 to find positive integers x, y so that $4f_q = (2x)^2 + y^2(4q)$, which implies f_q always split in the Hilbert class field of $\mathbb{Q}(\sqrt{-q})$.

From these simple observations, we have the following algorithm to test the primality of f_q .

Algorithm 12.1. (Exceptional Cases Test) This algorithm uses ECPP to test the primality of f_q .

1. Let $d = [-1, -2, -3, -7, -q]$.
2. Let $m = 0$.
3. If $m < 5$, then let $\ell = d[m]$, else go to the final step.
4. Compute $s = \left(\frac{\ell}{f_q}\right)$ and go to the next step.
5. If $s = -1, 0$, then increase m by 1 and return to Step 3, else go to the next step.
6. Let $K_\ell = \mathbb{Q}(\sqrt{\ell})$ and Let D_ℓ be the discriminant of K_ℓ .
7. Find positive integers x, y for which $4f_q = x^2 + y^2|D_\ell|$ if such exist. If no such x, y exist, then increase m by 1 and return to Step 3.
8. Determine if $p = f_q + 1 \pm x$ has a prime divisor $q > (f_q^{1/4} + 1)^2$. If it is too difficult to determine, increase m by 1 and return to Step 3, else go to the next step.
9. Let r be a root of $H_D(X) \pmod{f_q}$ and go to the next step. If no such r exists, then f_q is composite and we end the algorithm.
10. If $r = 0$ or $r = 1728$, take $E/(\mathbb{Z}/f_q\mathbb{Z})$ to be the curve $Y^2 = X^3 + 1$ or $Y^2 = X^3 + X$, respectively. If $r \neq 0, 1728$, take the curve $Y^2 = X^3 + ax - a$, where $a = 27r/(4(1728 - r)) \pmod{f_q}$. Take $P = (-1, 0)$ or $P = (0, 0)$ if $r = 0$ or $r = 1728$, respectively. Take $P = (1, 1)$ if $r \neq 0, 1728$.
11. Apply ECPP (Theorem 13.4) to the curve E with the point P .
12. If f_q is confirmed to be prime by ECPP, then we stop the algorithm, else increase m by 1 and return to Step 3.
13. The integer f_q is a probable prime.

13. WELL-KNOWN PRIMALITY TESTS

In this section we will discuss a number of primality tests. The computations done in these tests are some of the same computations needed in the construction of an elliptic curve of order f_q over some finite field, so there is no loss of computations in performing these tests. This observation is noted in [BS08] as well.

Assuming GRH, Bach [Bac90] has shown that if $p > 1000$, then there exists a quadratic non-residue z modulo p less than $2 \log^2(p)$. Furthermore, by the Chebotarev Density Theorem, half of the primes in the interval $[1, 2 \log^2 p]$ are quadratic residues modulo p . Hence, we have the following crude test for primality:

Theorem 13.1. (Density Test) Let p be a probable prime. Compute $\left(\frac{\ell}{p}\right)$ for all primes $\ell \leq 2 \log^2(p)$. If it takes about 50 trials to find a quadratic non-residue n or if the number of quadratic residues is not approximately $\log^2(p)$, then p is likely composite.

Theorem 13.2. Let p be a probable prime, and write $p - 1 = 2^e d$, where d is odd. If we can find an integer a such that

$$a^d \equiv 1 \pmod{p} \quad (13.1)$$

and

$$a^{2^r d} \not\equiv -1 \pmod{p} \quad (13.2)$$

for all $0 \leq r < e$, then p is not prime.

Notice that Theorem 13.2 is just the contrapositive of Fermat Little Theorem. We will now describe the Rabin-Miller Primality Test following [Coh93].

Algorithm 13.3. (Rabin-Miller Primality Test) Let p be a probable prime, and write $p - 1 = 2^e d$, where d is odd.

1. Choose 20 random integers in the interval $[2, p - 1]$, and store them in a set $W(p)$.
2. If $W(p)$ is empty, go to the final step, else pick $a \in W(p)$ and remove a from $W(p)$.
3. Let $k = 0$.
4. Compute $b = a^d \pmod{p}$.
5. If $b = \pm 1$, then return to Step 2., else increase k by 1 and go to the next step.
6. If $k = e$ and $b \not\equiv -1 \pmod{p}$, then p is composite, else go to the next step.
7. Compute $b^2 \pmod{p}$, and return to Step 5..
8. The prime p is a probable prime.

It is clear the the Rabin-Miller Primality Test has time complexity $\tilde{O}(\log^2(p))$ since we are computing only a few exponentiations.

Theorem 13.4. (Elliptic Curve Primality Proving) Let $p > 6$ be a probable prime. Let $E/(\mathbb{Z}/p\mathbb{Z})$ be an elliptic curve of order kq , where q is a prime such that $q > (p^{1/4} + 1)^2$. If there exists a point P on E such that $kqP = 0$ and kP is defined and not equal to 0, then p is a prime.

We repeatedly use ECPP in Step 11. of Algorithm 2.1. In Step 11., even when $p = f_q + 1 \pm x$ fails the Rabin-Miller Primality Test, we can still use it to test the primality of f_q using ECPP. If f_q is confirmed to be prime, then Step 16. does indeed confirm the primality of p .

For a discussion of Elliptic Curve Primality Proving see [AM93] and [Cox13, ch. 14].

REFERENCES

- [ACD⁺06] Roberto Avanzi, Henri Cohen, Christophe Doche, Gerhard Frey, Tanja Lange, Kim Nguyen, and Frederik Vercauteren. *Handbook of Elliptic and Hyperelliptic Curve Cryptography*. Chapman & Hall/CRC, Boca Raton, FL, 2006.
- [AM93] A.O.L. Atkin and F. Morain. Elliptic curves and primality proving. *Mathematics of Computation*, 61:29–68, 1993.
- [Bac90] Eric Bach. Explicit bounds for primality testing and related problems. *Mathematics of Computation*, 55(191):355–380, 1990.
- [BGL15] Pedro Berrizbeitia, Juan José Alba Gonzáles, and Florian Luca. On the formula $f_p = u^2 + pv^2$. *International Journal of Number Theory*, 11(3):185–191, 2015.
- [BL07] Christian Ballot and Florian Luca. On the equation $x^2 + dy^2 = f_n$. *Acta Arithmetica*, 127(2):145–155, 2007.
- [BLS12] Reinier Bröker, Kristin Lauter, and Andrew V. Sutherland. Modular polynomials via isogeny volcanoes. *Mathematics of Computation*, 81:1201–1231, 2012.
- [Bro08] Reinier Broker. A p -adic algorithm to compute the hilbert class polynomial. *Mathematics of Computation*, 77:2417–2435, 2008.
- [BS07] Reinier Bröker and Peter Stevenhagen. Efficient cm-constructions of elliptic curves over finite fields. *Mathematics of Computation*, 76:2161–2179, 2007.
- [BS08] Reinier Bröker and Peter Stevenhagen. Construction elliptic curves of prime order. *Contemporary Mathematics*, 463:17–28, 2008.
- [BV07] Johannes Buchmann and Ulrich Vollmer. *Binary quadratic forms: an algorithmic approach*, volume 20 of *Algorithms and Computations in Mathematics*. Springer, 2007.
- [Cal17] Chris Caldwell. The top twenty: Fibonacci number. <http://primes.utm.edu/top20/page.php?id=39>, 2017. Retrieved October 6, 2017.
- [CH88] Pierre Conner and Jurgen Hurrelbrink. *Class number parity*. World Scientific Publishing Co., Singapore, 1988.
- [Che12] Massimo Chenal. Applications of complex multiplication of elliptic curves. Master’s thesis, Università di Padova, Italy, 2012.
- [CL84] Henri Cohen and Hendrik Lenstra. Heuristics on class groups of number fields. 1068:33–62, 1984.
- [Coh93] Henri Cohen. *A course in computation algebraic number theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin Heidelberg, 1993.
- [Cox13] David Cox. *Primes of the form $x^2 + py^2$: Fermat, Class Field Theory, and Complex Multiplication*. Wiley, Hoboken, New Jersey, 2013.
- [DFJP14] Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.
- [DLR15] Harris B. Daniels and Álvaro Lozano-Robledo. On the number of isomorphism class of cm elliptic curves defined over a number field. *Journal of Number Theory*, 157:367–396, 2015.
- [EEW01] Manfred Einsiedler, Graham Everest, and Thomas Ward. Primes in elliptic divisibility sequences. *London Mathematical Society Journal of Computation and Mathematics*, 4:1–13, 2001.
- [Gee99] Alice Gee. Class fields by shimura reciprocity. *Journal de Théorie des Nombres de Bordeaux*, 11:45–72, 1999.
- [KK10] Elisavet Konstaninou and Aristides Kontogeorgis. Ramanujan’s class invariants and their use in elliptic curve cryptography. *Computers and Mathematics with Applications*, 59:2901–2917, 2010.
- [KKSZ09] Elisavet Konstaninou, Aristides Kontogeorgis, Yannis C Stamatiou, and Christos Zaroлиagis. On the efficient generation of prime-order elliptic curves. *Journal of Cryptology*, 23(3):477–503, 2009.
- [Koh96] David Kohel. *Endomorphism rings of elliptic curves over finite fields*. PhD thesis, University of California, Berkeley, 1996.
- [Kon14] Aristides Kontogeorgis. Constructing class invariants. *Mathematics of Computation*, 83:1477–1488, 2014.

- [LR11] Álvaro Lozano-Robledo. *Elliptic Curves, Modular Forms, and Their L-functions*, volume 58 of *Student Mathematics Library*. American Mathematical Society, Providence, Rhode Island, 2011.
- [onl17] The on-line encyclopedia of integer sequences. <https://oeis.org/A001605>, 2017. Retrieved October 6, 2017.
- [Sch85] René Schoof. Elliptic curves over finite fields and the computation of square roots mod p . *Mathematics of Computation*, 44(170):483–494, 1985.
- [Sch95] René Schoof. Counting points on elliptic curves over finite fields. *Journal de Théorie Des Nombres de Bordeaux*, 7(1):219–254, 1995.
- [Sil94] Joseph H. Silverman. *Advanced Topics in the Arithmetic of Elliptic Curves*. Springer New York, New York, 1994.
- [Sil09] Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2009.
- [SS71] A. Schnhage and V. Strassen. Schnelle multiplikation großer zahlen. *Computing*, 7:281–292, 1971.
- [SS06] Joseph H. Silverman and Nelson Stephens. The sign of elliptic divisibility sequence. *Journal of the Ramanujan Mathematical Society*, 21(1):1–17, 2006.
- [SS14] Igor E. Shparlinski and Andrew V. Sutherland. On the distribution of atkin and elkies primes. *Foundations of Computational Mathematics*, 14(2):285–297, 2014.
- [ST92] Joseph H. Silverman and John Tate. *Rational Points on Elliptic Curves*. Undergraduate Texts in Mathematics. Springer New York, New York, 1992.
- [Sut11] Andrew V. Sutherland. Computing hilbert class polynomials with the chinese remainder theorem. *Mathematics of Computation*, 80:501–538, 2011.
- [Sut12a] Andrew V. Sutherland. Accelerating the cm method. *LMS Journal of Computation and Mathematics*, 15:172–204, 2012.
- [Sut12b] Andrew V. Sutherland. Constructing elliptic curves over finite fields with prescribed torsion. *Mathematics of Computation*, 81:1131–1147, 2012.
- [VM02] Nikola Nikolaevich Vorobiev and Mircea Martin. *Fibonacci numbers*. Birkhäuser Verlag, Basel - Boston - Berlin, 2002.
- [Wal60] Donald Wall. Fibonacci series modulo m . *American Mathematical Monthly*, 67(6):525–532, 1960.
- [Wil82] H. C. Williams. A note on the fibonacci quotient $f_{p-\epsilon}/f_p$. *Canadian Mathematical Bulletin*, 25(3):366–370, 1982.
- [Wil87] H. C. Williams. Effective primality tests for some integers of the form $a5^n - 1$ and $a7^n - 1$. *Mathematics of Computations*, 48:285–306, 1987.
- [Zag84] Don Zagier. L-series of elliptic curves, the birch-swinnerton-dyer conjecture, and the class number problem of gauss. *Notices of the AMS*, 31:739–743, 1984.