Elliptic curves and isogenies: The good bits

Week 1 of the Isogeny-based Cryptography School, originally planned in {2020, Bristol}

Lorenz Panny

Academia Sinica, Taipei, Taiwan

July 5, 2021

1 Elliptic curves

Modern cryptography would not be the same without elliptic curves: Compactness and speed make them extremely attractive from an engineering perspective, while at the same time their cryptanalytic hardness is very close to optimal. More recently, they have also been proposed as a foundation for post-quantum cryptography, and in particular they give rise to the only known post-quantum non-interactive key-exchange scheme, which is very cool.

The following material is based on chapter 2 of my thesis [3], but I am taking a slightly more informal and hands-on approach here. I am honestly unsure if it is more helpful to read this or simply read my thesis. I suggest you try switching to the respective other whenever things become confusing. Also note that my thesis is probably the more reliable source when in doubt.

1.1 The bigger picture

Finding an algorithm that's hard to invert is not very difficult (in fact, a sufficiently long sequence of random bit operations is likely to work). Performing mathematically meaningful computations is also not hard: it's the very purpose computers were invented for in the first place.

What's significantly less obvious is how to combine these two features: Efficient algorithms having useful mathematical properties while making some other, related computations practically impossible. These kinds of structures are at the heart of *public-key cryptography*. The traditional main examples:

- Computing a third power modulo an integer of unknown factorization is easy, but taking a cube root seems a lot harder. (RSA)
- Computing exponentiations in a finite field is easy, but taking logarithms seems a lot harder. (DH)

Somewhat amusingly, most of those enrolled in a high-school mathematics curriculum would agree with the sentiment that powers are easier than roots or logarithms. In cryptography, we sprinkle some modular arithmetic on top and this vague sense of one-wayness becomes true in a much stricter sense.

Note to non-cryptographers: I say "*seems* harder" above because we can only very rarely *prove* that a cryptographic construction is secure. "Provable security" can reduce attack surface, but at the bottom of the argument there usually lies an unproven hardness *assumption* whose validity is often supported only by the extent that smart people have tried and failed to break it.

Elliptic curves are another one of several ways of obtaining such structures: Currently, the analogue of exponentiation on an elliptic curve is used as a fundamental building block for securing pretty much everything on the internet. In a future with large-scale quantum computers, these systems will all be broken, but we can build another kind of computationally useful structure from the theory of elliptic curves, and that is precisely what this entire school is about: Isogeny-based cryptography.

Without further ado, let's get started with the math background.

1.2 Weierstraß curves

The traditional, and still extremely useful, way of representing elliptic curves are *Weierstraß equations*. They are bivariate polynomial equations of the form

$$y^2 = x^3 + ax + b\,, \tag{1}$$

where a, b are constants in a field and x, y are symbolic variables. Elliptic curves are typically denoted by the letter E. When a, b are elements of a field k, we say that E is "defined over k" and write "E/k".

Almost all Weierstraß equations are elliptic curves, but there are a few exceptions: When the *discriminant* $\Delta = -16(4a^3+27b^2)$ is zero, the curve is singular, which makes it behave quite differently, hence this case is excluded from the theory of elliptic curves.

Every elliptic curve over a field of characteristic $\notin \{2, 3\}$ can be written as a short Weierstraß curve using coordinate transformations (i.e., isomorphisms).

The equation above defines a *short* Weierstraß curve. Slightly more complicated *long* Weierstraß equations also exist, but they are only required for fields of characteristic 2 and 3, which don't appear to be very interesting for isogeny-based cryptography.

A *point* on an elliptic curve E is a solution (ξ, η) of the defining equation (1), or another, distinct "point at infinity" ∞. For this to make sense, ξ and η must necessarily lie in some extension of the curve's base field.

Example. Points on the curve $y^2 = x^3 + x - 1$ include $(1, -1)$ and $(-1, \sqrt{-3})$ and of course ∞.

The notion of a "point" without further qualification refers to points over the algebraic closure, which is computationally inconvenient. In practice, we thus want to distinguish where the coordinates of a point actually live and what kinds of roots we need to adjoin to be able to write them down.

Example. The point $(1, -1)$ above is defined over any field, while the point $(-1, \sqrt{-3})$ is only defined over fields containing a square root of -3, which in particular includes algebraically closed fields such as $\mathbb{C}$ or prime finite fields $\mathbb{F}_p$ with $p \equiv 1 \pmod 3$.

When a point has coordinates in a field k, we say that it is *k-rational*. The set of all k-rational points on an elliptic curve E is denoted by $E(k)$. Note that by definition, the point at infinity ∞ is always included in $E(k)$. When E is used as a set, for instance in notation like $P \in E$, it refers to the set of points $E(\overline{k})$ over the closure.

In cryptography, we work almost exclusively with elliptic curves defined over finite fields $\mathbb{F}_q$. (Curves over characteristic-zero fields usually only show up in proofs.)

1.3 The j-invariant

Writing down equations for an elliptic curves involves some choices. Even when restricting to Weierstraß curves, the equation is generally not unique. Thus, it is convenient to have a simple means of determining when two curves are really "the same" in different coordinate systems, i.e., when they are *isomorphic*. (Isomorphisms are formally defined in Section 2.1.) The standard way of labelling isomorphism classes of elliptic curves is the *j-invariant*: For (short) Weierstraß curves, it is given by $j = 1728 \cdot 4a^3/(4a^3 + 27b^2)$. As desired, two elliptic curves are isomorphic (over the algebraic closure!) if and only if their j-invariants are the same.

1.4 The group law

The primary reason elliptic curves are interesting is because their set of points carries an algebraic group structure. What this means is that we can add and subtract points, such that these operations behave as expected, and the sum of two points can be written using *rational functions* (i.e., fractions of polynomials) of their coordinates.

This group law has a very intuitive geometric characterization: Any straight line intersects the curve in exactly three points (counted with multiplicities), and the sum of three points on the curve with respect to the group law equals the point at infinity if and only if they lie on such a straight line. (The point at infinity is declared to lie on all vertical lines.) See Figure 1.

Writing down these conditions in terms of polynomials and performing some manipulations yields the following explicit formulas for the group law on a short Weierstraß curve $y^2 = x^3 + ax + b$:

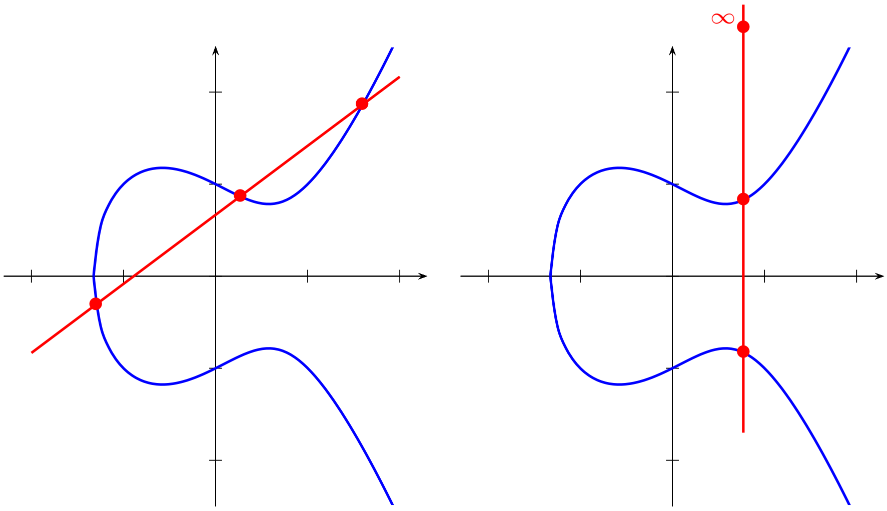

Figure 1: Illustration of the group law on a Weierstraß elliptic curve (pictured over $\mathbb{R}$).

- The neutral element is ∞.
- The inverse of (x, y) is $(x, -y)$.
- The sum of (x_1, y_1) and (x_2, y_2) is
$$\big(\lambda^2 - x_1 - x_2,\ \lambda(2x_1 + x_2 - \lambda^2) - y_1\big)$$
where $\lambda = (y_2 - y_1)/(x_2 - x_1)$ if $x_1 \neq x_2$ and $\lambda = (3x_1^2 + a)/(2y_1)$ otherwise.

It's not hard to see that applying negations or additions to k-rational points yields a result that is again k-rational, hence $E(k)$ forms a subgroup of the group of points of E.

1.4.1 Scalar multiplication

Any integer n defines a group homomorphism $[n]$ from an(y) elliptic curve E to itself. It consists of simply summing up n copies of an input point using the group law defined above:

$$[n]P = \underbrace{P + \cdots + P}_{n \text{ times}}$$

This scalar multiplication is the elliptic-curve analogue of exponentiation in the finite-field setting, and it is the basis of pre-quantum elliptic-curve cryptography: Computing the inverse map, that is, recovering $n \in \mathbb{Z}$ from $[n]P$, is the *elliptic-curve discrete logarithm problem* (ECDLP). For points P of prime order q and not particularly badly chosen curves, the best known classical algorithm for this problem takes time $\Theta(\sqrt{q})$, which is asymptotically optimal in the sense that this complexity is achievable by an attacker for *any* group of order q; in other words, noone has found a way for an attacker to make use of the specific algebraic structure of elliptic curves to accelerate ECDLP solvers.

Also note that we don't *really* perform $n-1$ individual point additions to compute $[n]P$, which would be slower than the attacker. Instead, we use the relations $[2k]P = [k][2]P$ and $[2k+1]P = [k][2]P + P$ to reduce a scalar multiplication to at most a doubling and an addition plus a scalar multiplication by an integer of one bit less. Hence this method (known as "double-and-add") takes time $\Theta(\log n)$, exponentially faster than the naïve method.

The *n-torsion subgroup* of E is the kernel of $[n]$, i.e., the set of points which are mapped to ∞ under multiplication by n. It is denoted by $E[n]$.

1.5 Projective coordinates

The formulas above involve a division, which is not great from a computational perspective as divisions are usually significantly more expensive than other arithmetic operations. One possible fix for this is to *defer divisions* until the end: Instead of performing a division a/b, we may simply store this fraction as is, i.e., as a tuple (a, b), and remember that a is a numerator and b is a denominator. Additions and multiplications can be performed on this representation using the familiar arithmetic rules for fractions $\frac{a}{b} \cdot \frac{c}{d} = \frac{ac}{bd}$ and $\frac{a}{b} + \frac{c}{d} = \frac{ad+bc}{bd}$, again storing the result as a pair of numerator and denominator instead of evaluating the division.

This reasoning is one way to arrive at *projective coordinates* for elliptic curves from a very concrete perspective: Points are represented projectively as $[x : y : z]$, where the z-coordinate is understood as a common denominator for x and y. Hence, (x, y) turns into $[x:y:1]$, and conversely $[x:y:z]$ corresponds to $(x/z, y/z)$ when z is non-zero. Just how expanding fractions does not change the value, $[x' : y' : z']$ is defined to be equal to $[x:y:z]$ whenever $(x', y', z') = (\lambda x, \lambda y, \lambda z)$ for some non-zero scalar λ.

The other way to get to projective coordinates is that it's really the right way of viewing elliptic curves mathematically: Among other things, it gets rid of the special handling of the point at infinity, which did not have (x, y) coordinates, but in the projective representation simply equals $[0 : 1 : 0]$.

1.6 x-only arithmetic

Notice that there are at most two points for a given x-coordinate. Hence, most of the information about a point is actually contained in the x-coordinate: Just one bit is encoded in the additional y-coordinate, while it *doubles* the representation size of the point. This might leave one wondering if we can get rid of the y-coordinate, and in many cases the answer is yes, using the following two observations:

- Negation changes only the y-coordinate. (Hence, y encodes a "sign bit".)
- Scalar multiplication commutes with negation (because it does in any group).

Combining these two facts implies that there exists an induced map $\texttt{xMUL}_n$ on x-coordinates that has the same effect on an x-coordinate as a scalar multiplication $[n]$ on any of the two points (x, y) with that x-coordinate. (Quotienting E by ± yields the "Kummer variety" or "Kummer line" of E.) Since discarding the y-coordinate saves space on the wire without incurring significant extra computation, we tend throw away y whenever possible and work with x exclusively.

1.7 Point counting and structure

Lots of things in cryptography depend on the number of points on an elliptic curve (e.g., the hardness of ECDLP, as mentioned in Section 1.4.1). Therefore, determining these group orders is very important in practice (it's also interesting to theoreticians for other reasons). Recall that $E(\mathbb{F}_q)$ is the subgroup of $\mathbb{F}_q$-rational points on E. One commonly writes $\#E(\mathbb{F}_q)$ for its cardinality $|E(\mathbb{F}_q)|$.

An elliptic curve E is a 1-dimensional object, hence we would expect that there are about q points defined over $\mathbb{F}_q$ on E. *Hasse's theorem* shows that this is indeed more or less true, with a square-root error term:

Theorem. Let $E/\mathbb{F}_q$. Then $\#E(\mathbb{F}_q) = q + 1 - t$ where $|t| \leq 2\sqrt{q}$.

Of course, this bound doesn't help much in figuring out the exact count: The search space is smaller, but still exponentially-sized in $\log(q)$. Luckily, there is a beautiful algorithm due to Schoof (with later improved, even more beautiful variants by Atkin and Elkies) that counts points in polynomial time:

Theorem. There is an explicit algorithm which, given the coefficients of a Weierstraß curve defined over $\mathbb{F}_q$, computes the number of $\mathbb{F}_q$-rational points on E in time polynomial in $\log(q)$.

For more details on the ideas behind this algorithm, see for example my B.Sc. thesis [4].

Point counting can be used to construct curves with properties that are not too rare. For example, finding curves with a large prime-order subgroup is easy by iterating through a bunch of curves (either deterministically or at random) and running a point-counting algorithm until the order has the desired factorization properties. Other properties cannot realistically be enforced using this method: For example, curves with very smooth[1] order are sparse, hence random sampling is not going to work here. This is a bit sad since isogeny-based cryptography relies heavily on smooth-order curves for efficiency reasons, but luckily, there's a workaround: see Section 1.8.

[1] An integer is *smooth* if it has only small prime factors, for some notion of "small".

1.7.1 The ℓ-torsion structure

Elliptic curves over $\mathbb{C}$ "are" basically the same thing as complex tori. (This fact is entirely non-obvious and relies on the so-called Weierstraß $\wp$ function.) One down-to-earth consequence of this is that the possible group structures of ℓ-torsion subgroups of an elliptic curve are very limited: They are always modules of rank at most two over $\mathbb{Z}/\ell$, and in many cases they are actually just a torus over $\mathbb{Z}/\ell$:

Theorem. Let E/k be an elliptic curve and ℓ a non-zero integer. If $\operatorname{char}(k) = p > 0$, factor ℓ as $m \cdot p^r$ with $m \notin p\mathbb{Z}$; otherwise, let $m = \ell$. Then $E[\ell] \cong \mathbb{Z}/m \times \mathbb{Z}/\ell$ or $E[\ell] \cong \mathbb{Z}/m \times \mathbb{Z}/m$ as groups.

In particular, either $E[p] \cong \mathbb{Z}/p$ or $E[p] \cong \{0\}$, and if $\operatorname{char}(k) \nmid \ell$ then $E[\ell] \cong \mathbb{Z}/\ell \times \mathbb{Z}/\ell$.

Recall that these torsion subgroups are over the algebraic closure. By mere counting, it's clear that not all of these subgroups can be k-rational when k is finite. To gain more insight into the *rational* torsion structure of an elliptic curve than just the point count (which however already implies a lot about the possible structures), we can apply a very useful result of Lenstra [2] which allows us to view the subgroup of rational points as a quotient of the curve's endomorphism ring (see Section 2.3).

1.8 Supersingularity

An elliptic curve E defined over a finite field $\mathbb{F}_q$ of characteristic p is *supersingular* if and only if p divides $\#E(\mathbb{F}_q) - q - 1$. Section 2.3 will show one way in which these curves have a much more complicated underlying theory, but for now, let's observe (by applying Hasse's theorem!) that the defining condition simplifies to $\#E(\mathbb{F}_p) = p + 1$ in the important special case that $q = p$ and $p \geq 5$. What this means that using supersingular curves allows us to pretty much choose exactly what the group order will be assuming there exists a convenient base-field prime, a fact that is tremendously useful when trying to construct efficient cryptosystems. The opposite of supersingular is *ordinary*.

The word "supersingular" is entirely unrelated to singularities (all elliptic curves are nonsingular by definition). It simply means something like "very special".

1.9 Honorable mention: Pairings

Pairings of elliptic curves are another large topic on their own, and they mostly play a support role in post-quantum isogeny-based cryptography, but I suppose I should at least briefly explain what's going on so people have heard of this.

In a nutshell, every elliptic curve admits a non-degenerate, alternating, bilinear map from $E[\ell]$ to the multiplicative group of (the algebraic closure of) the underlying field called the *Weil pairing*. It is usually written $e_\ell\colon E[\ell] \times E[\ell] \to \mu_\ell$.

One way of understanding this pairing is to fix a basis P, Q of $E[\ell]$, see Section 1.7.1, and see how the pairing acts on points expressed in terms of this basis: Using the defining properties of e_ℓ, we get

$$e_\ell([a]P + [b]Q, [c]P + [d]Q) = e_\ell(P, Q)^{ad-bc}.$$

Note that $ad - bc$ is just the determinant of $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$. However, the magic is that the data is "hidden in the exponents" of the groups $E[\ell]$ resp. μ_ℓ, and yet we can evaluate the pairing efficiently without first recovering the exponents "in the clear".[2]

Also note that there are other pairings in use in cryptography, which are however fundamentally still derived from the Weil pairing. Of course, there are important practical differences.

1.10 Alternate curve models

For implementations, we often prefer curve representations other than the Weierstraß form of a curve. The most common choices are the *Montgomery* form $By^2 = x^3 + Ax^2 + x$, which admits very fast x-only arithmetic, and the (twisted) *Edwards* form $ax^2 + y^2 = 1 + dx^2y^2$, which in well-chosen cases has *complete*

[2] The standard algorithm is due to Miller and requires $\Theta(\log \ell)$ operations in the field of definition of the input points. The trouble is that generic high-order points are usually defined over huge extension fields, so even though the complexity is low in terms of base-field operations, it may still blow up exponentially when representing the base field requires huge extensions. "Pairing-friendly curves" are constructed such that the respective groups are defined over relatively small fields, hence they admit fairly efficient pairings.

addition formulas without case distinctions like the special case of doublings in Section 1.4. Perhaps the only drawback of these two curve shapes is that not all elliptic curves can be written in these forms: For example, every Montgomery curve has the rational point $(0, 0)$ of order two, while clearly not all elliptic curves have rational points of order two. In particular, no prime-order curves can be converted into Montgomery or Edwards form, which some people dislike because it incurs extra validation effort (hence potential for human error) in protocols.

2 Isogenies of elliptic curves

Shor's quantum algorithm solves discrete-logarithm problems in any group, including elliptic curves, in polynomial time. Hence for a couple of decades it seemed like elliptic curves in cryptography were a fad that would only last until large-scale quantum computers are constructed. Luckily, the more recent field of isogeny-based cryptography demonstrates that this is (probably) not true, which allows us to continue playing with elliptic curves even after the quantum apocalypse.

2.1 Definition & examples

"Isogeny" is a fancy word, but they might just as well have been called "nice maps". As explained above, the two main characteristics of elliptic curves is that they are *algebraic* curves carrying an algebraic *group structure*. Now an *isogeny* is simply a non-zero[3] map between two elliptic curves that respects both of these structural properties, i.e., the map is given by *rational functions* and it is a *group homomorphism*.

Example. The map $(x, y) \mapsto \left(\frac{x^3-4x^2+30x-12}{(x-2)^2}, y \cdot \frac{x^3-6x^2-14x+35}{(x-2)^3}\right)$ is an isogeny from $y^2 = x^3 + x$ to $y^2 = x^3 - 3x + 3$ over $\mathbb{F}_{71}$.

More generally, an isogeny $\varphi\colon E \to E'$ between two Weierstraß curves E, E' defined over k can always be written in the form

$$\varphi((x, y)) = \left(\frac{f}{h^2}(x),\, y \cdot \frac{g}{h^3}(x)\right),$$

where f, g, h are polynomials in $k[x]$ and poles of f/h^2 and g/h^3 signify that the result is ∞. The *degree* of the isogeny equals the smallest possible degree of a rational function expressing the isogeny; it is therefore a measure for the algebraic (and, as discussed below, computational) complexity of an isogeny. Degrees are multiplicative: $\deg(\varphi \circ \psi) = \deg(\varphi) \cdot \deg(\psi)$.

Example. The isogeny from the previous example has degree 3.

Example. Scalar multiplication $[m]$ has degree m^2. (This follows from the structure of the m-torsion given in Section 1.7.1.)

Special names are given to special isogenies:

- Isogenies of degree 1 are *isomorphisms*. (This matches the intuitive notion of a coordinate change.)
- Isogenies from a curve to itself, together with the zero map [0], are *endomorphisms*. (See Section 2.3.)
- Endomorphisms that are also isomorphisms are *automorphisms*.

Just like for points, an isogeny is *defined over k* whenever the coefficients in its formula all lie in k.

2.1.1 Dual isogenies

Every isogeny $\varphi\colon E \to E'$ comes with a unique isogeny $\widehat{\varphi}\colon E' \to E$ in the opposite direction, defined by the property that $\widehat{\varphi} \circ \varphi = [\deg(\varphi)]$ and $\varphi \circ \widehat{\varphi} = [\deg(\varphi)]$. In this sense, the dual is *almost* an inverse, except that one would need to divide by the degree to get there.

This shows that being isogenous is an equivalence relation. Isogenous curves over a finite field have the same number of points, and remarkably, the converse holds as well (this is *Tate's isogeny theorem*):

Theorem. Two elliptic curves defined over a finite field k are isogenous over k if and only if they have the same number of k-rational points.

[3] The zero map $P \mapsto \infty$ is usually excluded because it is an annoying outlier that would be an exception for most theorems dealing with isogenies. (Some authors do include the zero map, but let's not.)

Remembering that point counting is efficient, this shows that the property of being isogenous can easily be decided. *Finding* an isogeny is dramatically harder for all we know (as will be discussed in weeks 6 and 8).

2.1.2 Frobenius isogenies

For any elliptic curve E defined over a field of characteristic p, let $E^{(p^r)}$ denote the curve obtained by raising all coefficients of the defining equation to the p^rth power. The isogeny

$$\pi\colon E \to E^{(p^r)},\ (x,y)\mapsto (x^{p^r}, y^{p^r})$$

is the *p^r-power Frobenius isogeny* of E. It has degree p^r. In the important special case $E/\mathbb{F}_{p^r}$, we have $E^{(p^r)} = E$ and π is the *Frobenius endomorphism.*

Due to its intimate connection with the Galois group of the underlying field, the Frobenius endomorphism is related to the structure of rational points of the curve. In particular, what Schoof's algorithm actually does to count points is computing the characteristic polynomial of the Frobenius endomorphism: It equals $X^2 - tX + p^r$ where $\#E(\mathbb{F}_{p^r}) = p^r + 1 - t$.

Frobenius isogenies are very special (as will become clear for instance in Section 2.2), hence their appearance is again emphasized by a name: An isogeny $\varphi\colon E \to E'$ is called *inseparable* if it factors through a Frobenius isogeny, otherwise it is *separable.* Moreover, φ is *purely inseparable* if it equals a Frobenius isogeny composed with an isomorphism.

(All this terminology, including the degree, is borrowed from the extension of function fields corresponding to the isogeny.)

2.2 Isogenies from kernels

Perhaps *the* most important object when doing isogeny-based cryptography is the *kernel* of an isogeny, i.e., the set of points mapped to ∞. It is not hard to see from the general form of a Weierstraß isogeny that the kernel is a finite set. It is also a subgroup, because it is for any group homomorphism.

(Sometimes, in particular for higher-dimensional abelian varieties, finite kernels are part of the definition of an isogeny.)

Example. The kernel of the isogeny over $\mathbb{F}_{71}$ from the example in Section 2.1 is $\{\infty, (2,9), (2,62)\}$ since these are the points where the denominator vanishes.

Example. Purely inseparable isogenies have trivial kernel. (To see this, observe that $x \mapsto x^p$ is injective in any field of characteristic p.)

The significance of the kernel is that it defines an isogeny almost uniquely. The only disturbance is inseparability, since purely inseparable isogenies have trivial kernel despite being more than just an isomorphism: If $\varphi\colon E \to E'$ and $\psi\colon E \to E''$ have the same kernel, then $\psi = \alpha \circ \varphi$ for some purely inseparable $\alpha\colon E' \to E''$. (Recall that purely inseparable isogenies are a composition of Frobenius isogenies and isomorphisms. The Frobenius part is often trivial, so that α is just an isomorphism.)

Morever, *every* finite subgroup is the kernel of an isogeny![4] This can be seen, for instance, by simply writing down formulas that construct an isogeny with prescribed kernel (see Section 2.2.1). Hence, and I like to think of this as the main theorem underlying isogeny-based cryptography:

Theorem. There is a one-to-one correspondence from finite subgroups of an elliptic curve to separable isogenies from said curve, up to post-composition with isomorphisms.

This (almost unique) isogeny with kernel H is often denoted by φ_H or similar, and the codomain (or rather, one particular representative of the unique isomorphism class) is denoted by E/H in analogy to quotients of groups.

We always have $\deg \varphi_H = |H|$. (More generally, the cardinality of the kernel of an isogeny equals the degree of its separable part.)

2.2.1 Vélu's formulas

Let E/k. In 1971, Vélu found a nice trick to write down an isogeny with a given kernel $H \leq E$: The basic idea is to exploiting the existing Weierstraß coordinate projections $x, y\colon E \setminus \{\infty\} \to k$ to construct

[4] This is reminiscent of how every subgroup of an abelian group is the kernel of a homomorphism. The result here is stronger in that the codomain can be chosen to be another elliptic curve, and the homomorphism can be realized as an *algebraic* map.

algebraic projection maps $f_x, f_y\colon E \setminus H \to k$ which are invariant under translations by elements of H. These functions are then well-defined on the curve E/H (which thus far exists only abstractly) and can be used to set up an embedding of E/H into the plane, which finally yields an equation.

Concretely, for $\gamma \in \{x, y\}$ (the Weierstraß coordinate projections) and a point $P \in E$, Vélu defines

$$f_\gamma(P) := \gamma(P) + \sum_{\substack{Q \in H \\ Q \neq \infty}} \big(\gamma(P+Q) - \gamma(Q)\big) \,.$$

Then the map

$$\varphi\colon E \to E/H,\ P \mapsto \big(f_x(P), f_y(P)\big)\,,$$

where poles of f_x, f_y get mapped to the point at infinity, is a separable isogeny with kernel H. The codomain is again a Weierstraß curve whose equation can be recovered using some extra steps.

2.2.2 x-only isogenies

Noting that groups are closed under inversion, we observe that the y-coordinate of the points in a kernel subgroup should be irrelevant (just like for scalar multiplications). This is indeed true, and we can compute isogenies using x-only arithmetic.

2.2.3 Smooth-degree isogenies

The complexity of Vélu grows linearly with the subgroup size. However, for cryptographic purposes, we need to do better, similar to how computing a scalar multiplication naïvely is not good enough to beat attackers trying to solve ECDLP. Hence, we use subgroups H of smooth order (i.e., only small prime factors, often large powers of a small prime) and decompose the isogeny φ_H into a sequence of many small-prime-degree isogenies.

For example, for $H \cong \mathbb{Z}/\ell^k$, set $\ker\psi_i := [\ell^{k-i}](\psi_{i-1} \circ \cdots \circ \psi_1)(H)$; then each ψ_i has degree ℓ.

$$\begin{tikzcd} E \arrow[r, "\psi_1"] \arrow[rrrr, bend right, "\varphi_H"'] & E_1 \arrow[r, "\psi_2"] & \ldots \arrow[r, "\psi_{k-1}"] & E_{k-1} \arrow[r, "\psi_k"] & E/H \end{tikzcd}$$

This reduces the complexity from $\Theta(\ell^k)$ down to $\Theta(k^2 \cdot \ell)$ base-field operations when the ℓ-isogenies ψ_i are computed using Vélu's original formulas[5], representing an exponential speedup.

2.3 The endomorphism ring

The set of endomorphisms of an elliptic curve carries an algebraic structure itself: We can add endomorphisms pointwise (in fact, addition works for isogenies between a fixed pair of curves in general), and we can "multiply" them by composing them. It is not very hard to check that these two operations turn the set of endomorphisms into a well-defined (not necessarily commutative) *ring*, which is denoted by $\mathrm{End}(E)$. (For every field k the curve is defined over, the endomorphism ring has a subring $\mathrm{End}_k(E)$ consisting of just the k-rational endomorphisms. We'll focus on $\mathrm{End}(E)$ for now.)

In general, for elliptic curves over finite fields $\mathbb{F}_q$, the endomorphism ring contains at least:

- All scalar multiplications. In other words, $\mathbb{Z}$ is a subring of $\mathrm{End}(E)$.
- The q-power Frobenius endomorphism π.

For ordinary elliptic curves, one can show that this is already very close to the entirety of the ring: Their endomorphism ring is of the form $\mathbb{Z}[(\pi + m)/f]$, where m and f are non-zero integers.[6] In more complicated words: The endomorphism ring is an *order* containing $\mathbb{Z}[\pi]$ inside the imaginary quadratic field $\mathbb{Q}(\pi)$.

[5] Since 2020, there exists the $\sqrt{\text{é}}$lu ("square-root Vélu") algorithm [1] to compute isogenies of prime degree ℓ in time $\tilde{O}(\sqrt{\ell})$ rather than $\Theta(\ell)$.

[6] Those wondering what dividing an endomorphism by an integer is even supposed to mean have a point: This is only possible in special cases, namely (here) when $\pi + m$ kills the entire f-torsion of the curve, and then the kernel of $(\pi + m)/f$ can be recovered as the image of the kernel of $\pi + m$ under $[f]$.

For supersingular elliptic curves, the story is much more complicated: Their endomorphism rings are *maximal orders in a quaternion algebra*. That quaternion algebra is $B_{p,\infty}$, which has a technical meaning, but the most important thing to know is that $B_{p,\infty}$ basically consists of two quadratic fields "glued together" in a non-commuting way: There exist elements $\mathbf{i}, \mathbf{j} \in B_{p,\infty}$ such that

$$B_{p,\infty} = \mathbb{Q} + \mathbb{Q}\mathbf{i} + \mathbb{Q}\mathbf{j} + \mathbb{Q}\mathbf{ij}$$

obeying the multiplication rules

$$\mathbf{i}^2 = -q\,, \quad \mathbf{j}^2 = -p\,, \text{and} \quad \mathbf{ji} = -\mathbf{ij}\,,$$

where q is a positive integer with some specific properties depending on p. An *order* in $B_{p,\infty}$ is a subring that has full rank (i.e., rank 4) as an abelian group, and a *maximal order* is an order that is not properly contained in any other order.

Example. Let $p \equiv 3 \pmod 4$. The elliptic curve $E\colon y^2 = x^3 + x$ over $\mathbb{F}_{p^2}$ is supersingular. Its endomorphism ring contains both the p-power Frobenius endomorphism $\pi\colon (x,y) \mapsto (x^p, y^p)$ and the automorphism $\iota\colon (x,y) \mapsto (-x, \sqrt{-1} \cdot y)$. By the congruence condition on p, the square root of -1 lies in $\mathbb{F}_{p^2} \setminus \mathbb{F}_p$, hence $\pi\iota = -\iota\pi$. Thus, an embedding of $\mathrm{End}(E)$ into $B_{p,\infty}$ is given by $\iota \mapsto \mathbf{i}$ and $\pi \mapsto \mathbf{j}$. In fact, it is not extremely hard to show starting from this point that $\mathrm{End}(E) = \mathbb{Z} + \mathbb{Z}\iota + \mathbb{Z}\frac{\iota+\pi}{2} + \mathbb{Z}\frac{1+\iota\pi}{2}$.

Warning. The identification of $\mathrm{End}(E)$ with a subring of $B_{p,\infty}$ is in general highly non-unique. Do not assume any kind of compatibility unless there is a good reason to do so.

2.4 Kernel ideals

In this final section, we briefly discuss an important connection between the set of isogenies attached to a curve and the structure of its endomorphism ring.

Let $\mathcal{I}$ be a left ideal[7] of $\mathrm{End}(E)$. Any such ideal defines a finite subgroup, namely the intersection of the kernels of all endomorphisms in the ideal (and note that it suffices to use a set of generators of the ideal). Hence, we get a mapping from left ideals of $\mathrm{End}(E)$ to isogenies emanating from E.

If $\mathcal{I}$ is principal, the kernel of the corresponding isogeny is just the kernel of the endomorphism generating the ideal. Hence, the isogeny equals that endomorphism (up to isomorphism), and in particular multiplying an ideal by a principal ideal does not change the codomain of the corresponding isogeny.

The bottom line of these observations is a famous theorem for elliptic curves with imaginary quadratic endomorphism ring, which lies at the heart of the CSIDH cryptosystem (week 3): It yields a well-behaved group action of the ideal-class group of $\mathrm{End}_k(E)$ on a certain set of elliptic curves. This result is one of the main reasons for discussing class groups in this context (second half of week 1).

References

[1] Daniel J. Bernstein, Luca De Feo, Antonin Leroux and Benjamin Smith. "Faster computation of isogenies of large prime degree". In: *ANTS XIV: Proceedings of the fourteenth algorithmic number theory symposium*. Ed. by Steven Galbraith. Auckland, 2020. URL: `https://iac.r/2020/341`.

[2] Hendrik W. Lenstra, Jr. "Complex Multiplication Structure of Elliptic Curves". In: *Journal of Number Theory* 56.2 (1996), pp. 227–241. ISSN: 0022-314X.

[3] Lorenz Panny. "Cryptography on Isogeny Graphs". PhD thesis. Technische Universiteit Eindhoven, 2021. URL: `https://yx7.cc/docs/phd/thesis.pdf`.

[4] Lorenz Panny. "Schoof's algorithm for elliptic curves". B.Sc. thesis. Technische Universität München, 2015. URL: `https://yx7.cc/docs/tum/thesis_schoof.pdf`.

[5] The Sage Developers. *SageMath, the Sage Mathematics Software System*.

[7] This choice of *left* ideals is arbitrary. Of course, when $\mathrm{End}(E)$ is commutative, there is no difference.

3 Exercises

All of these are primarily intended to be solved with the assistance of a computer-algebra system such as SageMath [5]. Crash course: The elliptic curve $y^2 = x^3 + ax + b$ over $\mathbb{F}_q$ can be defined using `E = EllipticCurve(GF(q), [a,b])`, and a point $P = (x, y)$ on E can be constructed by writing `P = E(x,y)`. An isogeny $E \to E'$ with kernel $\langle K \rangle$ can be constructed by writing `f = E.isogeny(K)`, but note that this may take effectively forever when the order of K is big. Other functionality can usually be found by typing the right words into a search engine or by asking people who know stuff.

3.1 The group law

Let E denote the elliptic curve $y^2 = x^3 - 7x + 10$ over $\mathbb{Q}$. Let g be the straight line defined by $y = 2x - 2$. Compute all points of intersection between E and g and verify that there are indeed three points as claimed and that their sum with respect to the elliptic-curve group law is ∞.

3.2 A basis of the ℓ-torsion

Let $p = 18446744073709551667$. Define the elliptic curve $E\colon y^2 = x^3 + x$ over $\mathbb{F}_{p^2}$. Compute a basis of the 4999-torsion subgroup $E[4999]$, i.e., find two points $P, Q \in E$ such that any point in $E[4999]$ is a unique $\mathbb{Z}/4999$-linear combination of P and Q.

As 4999 is prime, $\mathbb{Z}/4999$ is a field, hence things like linear (in)dependence work as you would expect from a two-dimensional vector space. In particular, P and Q are dependent if and only if they are scalar multiples of one another.

3.2.1 Two-dimensional discrete logarithms

Devise and implement an algorithm that computes an isomorphism $E[4999] \to \mathbb{Z}/4999 \times \mathbb{Z}/4999$, i.e., takes any point $R \in E[4999]$ and returns $a, b \in \mathbb{Z}$ such that $R = [a]P + [b]Q$, where P, Q is your basis of the 4999-torsion from before. What's the complexity of your method?

3.3 Verifying supersingularity

Let again $p = 18446744073709551667$. Check that the curve $y^2 = x^3 + 6120164818944x + 9660707028$ defined over $\mathbb{F}_p$ has $p + 1$ rational points. Can you do it without using a point-counting algorithm?

3.4 Decomposing isogenies

Consider $p = 2^{127} - 1$ and $E\colon y^2 = x^3 + x$ over $\mathbb{F}_{p^2}$. Pick a random point $P \in E$ of order 2^{127}. Implement the decomposition technique from Section 2.2.3 to compute the isogeny with kernel $\langle P \rangle$. (The output of your computation should be a list of 127 degree-2 isogenies whose composition has kernel $\langle P \rangle$.)

(Note that SageMath does *not* perform this decomposition on its own and resorts to naïvely enumerating the 2^{127} points in the subgroup instead. Someone who hopefully isn't me should probably improve this behaviour in SageMath at some point.)

3.5 The kernel of the dual

Let E be an elliptic curve and P, Q be a basis of the ℓ-torsion subgroup for some integer ℓ such that this makes sense (i.e., excluding the few cases where $E[\ell] \ncong \mathbb{Z}/\ell \times \mathbb{Z}/\ell$). Let $K = [a]P + [b]Q$ for some $a, b \in \mathbb{Z}$ and let $\varphi\colon E \to E'$ be an isogeny with kernel $\langle K \rangle$. What is the kernel of the dual isogeny $\widehat{\varphi}$?

Verify your answer experimentally by choosing a suitable curve and trying it out on a few examples.

3.6 Quaternions!

Let $p = 2^{32} - 5$ and $E\colon y^2 = x^3 + 1$ over $\mathbb{F}_{p^2}$. Let $\zeta \in \mathbb{F}_{p^2}$ be a nontrivial cube root of unity. Check that $\pi\colon (x, y) \mapsto (x^p, y^p)$ and $\omega\colon (x, y) \mapsto (\zeta \cdot x, y)$ are non-commuting endomorphisms of E. Verify that $\vartheta := (1 - \omega + \pi - \omega\pi)/3$ is a well-defined endomorphism of E.

To my knowledge, SageMath does not have any meaningful support for endomorphism rings of elliptic curves. You'll need to get quite a few things to work on your own. An alternative is to do the computations literally by hand like in the dark ages.