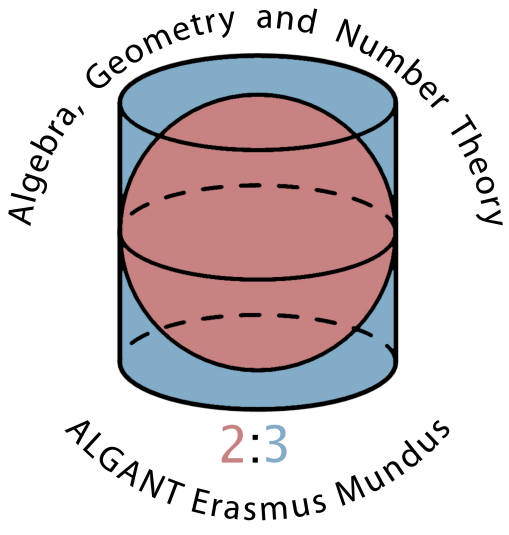


Supersingular isogeny key-exchange

Margarita Pierrakea

Advised by Damien Robert

Universiteit
Leiden

Academic year 2016-2017

Contents

1 Introduction

Some decades now scientists around the world have been trying to build quantum computers. The idea behind a quantum computer is to use the theory of quantum mechanics instead of classical mechanics and base on it the function of computers. In classical computers the data are encoded into *bits*, i.e. binary digits, each of which is either 0 or 1. Instead, in quantum computers *qubits* (quantum bits) are used, which do not have a deterministic state, but they take a spectrum of values in the same time. Using quantum theory horology, they can be in *superpositions* of states.

The algorithms used in the present time for cryptographic purposes, can be broken in polynomial time if the development of a quantum computer becomes reality. In this master thesis, we are studying an algorithm that is believed to be quantum resistant, since no serious attacks against it have been found yet, and therefore can be used in post-quantum era, if this becomes a reality.

This algorithm is called SIDH (Supersingular Isogeny Diffie Hellman) and is proposed by Luca de Feo, David Jao and Jerome Plut[DFJP14]. It is a key exchange method based on Diffie-Hellman key exchange. In SIDH, a supersingular elliptic curve is one of the initial public parameters. In contrast to Elliptic Curve Diffie-Hellman, we do not pick elements in the group of our elliptic curve, but we compute isogenous curves to it. So we pick elements from the so called *isogeny graph*, whose vertices are elliptic curves and its edges (in our case arrows) are isogenies. Each one of Alice and Bob will compute curves isogenous to the public elliptic curve and they will eventually conclude to the same isogenous curve, the j-invariant of which will be used as their common secret shared key.

The correspondence between the two methods is shown in Table 1.

Diffie-Hellman	Supersingular Isogeny Diffie-Hellman
Group G	isogeny graph
elements of G	elliptic curves on the isogeny graph

Table 1: Correspondence.

The first attempt in creating a Diffie-Hellman style key-exchange using isogenies was made by Rostovtsev and Stolbunov[RS06]. In their protocol, ordinary elliptic curves were used. Later, an attack was found against that protocol which was quantum sub-exponential. This lead mathematicians to research more in order to find a quantum exponential protocol. The idea to use supersingular elliptic curves has turned out to be a successful one, since the fastest known attack against SIDH is quantum exponential.

SIDH is based on the conjectured difficulty of finding isogenies between supersingular elliptic curves. The two parties participating in the key-exchange can find such isogenies easily, using Velu's formula, since they both secretly know the kernel of the isogenies they want to compute. A third party, however, who will potentially try to interfere, without knowledge of these kernels, will not be able to compute the isogenies.

A crucial element in Diffie-Hellman protocols that allows Alice and Bob to find the same key, is that the group they work with is commutative. In ordinary isogeny Diffie-Hellman, is the endomorphism rings of ordinary elliptic curves that is commutative. Supersingular curves, however, don't have commutative endomorphism rings and so they couldn't be used in the protocol developed by Rostovtsev and Stolbunov. However, in SIDH, the inventors overcame this obstacle.

A brief overview of cryptography But let's see how cryptography became what it is today. In Roman times, in order to encrypt a message, a simple exchange of letters was used. The person who received the encrypted message, just had to know how the letters where exchanged with each other in order to decrypt it. This information is called the *key*. If an enemy gets to know this key, then he can decrypt the message. As centuries pass, the systems improve and improve. But the situation is the following: Bob wants to send a secret message to Alice. He uses a secret key k to scramble his plain text message m and turn it into a cipher text c. Alice, upon receiving c, uses the secret key k to unscramble c and reconstitute m. If this procedure is to work properly, then both Alice and Bob must possess copies of the secret key k, and if the system is to provide security, then their adversary Eve must not know k, must not be able to guess k, and must not be able to recover m from c without knowing k.

Systems like this have obvious disadvantages. How will Alice and Bob exchange the secret key? How they will ensure that their enemy Eve will not steel it? If they exchange the key once and then use it multiple times to encrypt messages, they avoid the danger hidden in exchanging the key, but then there is the danger that their enemy will get an insight in several messages and will manage to figure out the key they are using. Moreover, in many older cryptosystems, an important part of security was to keep the encryption method secret. These are serious threats.

The breakthrough in cryptography happened with the idea that, instead of the two parties using the same secret key which they have to exchange before exchanging any messages, each of them uses a different secret key that only him/herself knows and, after following a specific procedure, they both individually end up to the same shared key. This solves the problem of finding a secure channel through which Alice and Bob exchange their secret key. This led to the so called *asymmetric cryptography.* The second brilliant idea was to develop a cryptosystem that even if Eve knows it, she cannot decrypt the messages sent through that without knowing the private secret keys that are used.

The first protocol that embodied these principals is the *Diffie-Hellman key exchange*, developed by Whitfield Diffie and Martin Hellman and published in 1977.

The Diffie-Hellman key-exchange We choose an element g and consider the cyclic group G that it generates. These are public knowledge. Alice randomly picks $a \in G$ and computes g^a. Meanwhile, Bob randomly picks $b \in G$ and computes g^b. Alice sends g^a to Bob and Bob sends g^b to Alice. Then, Alice computes $(g^b)^a$ and Bob computes $(g^a)^b$. Their common secret key is

$$(g^b)^a = g^{ba} = g^{ab} = (g^a)^b.$$

The communication channel that Alice and Bob use to exchange information is not considered to be safe, so we assume that Eve is able to learn the values g^a and g^b. However, she is not able knowing these, but not knowing a and b, to compute the key g^{ab}. In particular, the knowledge of g combined with the knowledge of g^a and g^b are not enough to recover a and b. This is the *discrete logarithm problem*, that is exponential on a classic computer and is the "difficult" problem which Eve has to solve in order to interfere in the communication of Alice and Bob. The security of Diffie-Hellman key-exchange is based on the difficulty of solving the discrete logarithm problem.

To be more precise, the discrete logarithm problem is the following:

Discrete logarithm problem: Given a group G and two elements $g, h \in G$ find x such that $g^x = h$.

If g has order n, simply by computing all powers of g up to n one can find x after n multiplications. However, if n is large enough ($n > 2^{80}$) then it is practically impossible, with the computing power available today, to find x . There are other algorithms that solve the discrete logarithm problem and some can do it quickly enough, if the values are not chosen carefully. For example, if $p - 1$ is a product of small primes, then an algorithm called *Pohlig–Hellman algorithm*, gives a quick solution to the discrete logarithm problem in F_p^*.

Another version of Diffie-Hellman key exchange, is Elliptic Curve Diffie-Hellman. In this case, Alice and Bob agree to use a particular elliptic curve $E(\mathbb{F}_p)$, for a prime p, and a particular point $P \in E(\mathbb{F}_p)$. Alice chooses a secret integer n_A and Bob chooses a secret integer n_B. They compute the associated multiples $n_A P$ and $n_B P$ and they exchange these values. Alice then uses her secret multiplier to compute $n_A Q_B$, and Bob similarly computes $n_B Q_A$. They now have the shared secret value which they can use as a key to communicate privately via a symmetric cipher. Elliptic curve Diffie-Hellman key exchange is summarized in the next table.

Public parameter creation	
A trusted party chooses and publishes a large prime p, an elliptic curve E over $\mathbb{F}_p$ and a point $P \in E(\mathbb{F}_p)$.	
Private computations	
Alice	**Bob**
Chooses a secret integer n_A. Computes the point $Q_A = n_A P$.	Chooses a secret integer n_B. Computes the point $Q_B = n_B P$.
Public exchange of values	
Alice sends Q_A to Bob	Bob sends Q_B to Alice
Further private computations	
Alice	**Bob**
Computes the point $n_A Q_B$.	Computes the point $n_B Q_A$.
The shared secret value is $n_A Q_B = n_A(n_B P) = n_B(n_A P) = n_B Q_A$.	

Table 2: Elliptic curve Diffie-Hellman.

As in the generic case of finite multiplicative groups, elliptic curve Diffie-Hellman is based on the difficulty of solving the discrete logarithm problem. On elliptic curves, however, despite the highly structured nature of the group $E(\mathbb{F}_p)$, the discrete logarithm problem appears to be much harder. The fastest known algorithm to solve elliptic curve Diffie-Hellman in $E(\mathbb{F}_p)$ takes approximately $\sqrt{p}$ steps, while in $\mathbb{F}_p^*$ there are algorithms with running time $O(p^e)$ for every $e > 0$.

Computational complexity This scheme achieves performance on the order of one second at the 128-bit security level as measured against the fastest known quantum attacks using desktop PCs.

	Classical attack	**Quantum attack**
$\mathbb{F}_p^*$	sub-exponential	polynomial
$E/\mathbb{F}_p$	exponential	polynomial
ordinary elliptic curves	exponential	sub-exponential
supersingular elliptic curves	exponential	exponential

Table 3: Diffie-Helmann type protocols.

2 Elliptic curves

Elliptic curves are cubic plane projective curves of genus 1 with a rational point. We will mainly describe them and their points in affine coordinates, but we will switch to projective ones when it is necessary.

We define the projective plane over a field k as the set

$$\mathbb{P}^2(k) = \{[X, Y, Z] : X, Y, Z \in k, X, Y, Z \text{ not all } 0\}/\sim,$$

where with $\sim$ we denote the equivalent relation defined as follows:

$[a, b, c] \sim [a', b', c']$ if and only if there exists $\lambda \in k^*$ such that $a' = \lambda a, b' = \lambda b$ and $c' = \lambda c$.

2.1 Weierstrass equations

Definition 2.1.1. A *general Weierstrass equation* over a field k is an equation of the form

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6,$$

with $a_1, a_2, a_3, a_4, a_6 \in k$.

To make it homogeneous we set $x = \frac{X}{Z}$ and $y = \frac{Y}{Z}$ and it takes the projective form

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3.$$

Definition 2.1.2. An elliptic curve is a smooth plane projective curve given by a Weierstrass equation over a field k together with a point O, where O has homogeneous coordinates $(0 : 1 : 0)$.

Remark 2.1.3.

1. The point O is called the *point at infinity*, and it is a rational point, i.e. a point whose coefficients are rational numbers.
2. To pass from a projective curve to an affine curve, we map $[a, b, c]$ to $(a/c, b/c)$, if $c \neq 0$, and to O, if $c = 0$. On the other hand, a point (a, b) of an affine curve corresponds to $[a, b, 1]$ expressed in projective coordinates.
3. If the field k has characteristic different from 2 and 3, then every Weierstrass equation admits a change of coordinates that transforms it into

   $$y^2 = x^3 + ax + b.$$

   We call this a *short Weierstrass equation.*
4. A *singular point* of an elliptic curve is a point of the curve that is also a solution of the derivative of the curve. An elliptic curve is called *smooth*, if it has no singular points.

A quantity that gives important information for an elliptic curve is the discriminant. We define it only for short Weierstrass equations.

Definition 2.1.4. The *discriminant* Δ of an elliptic curve in short Weierstrass equation is

$$\Delta = -16(4a^3 + 27b^2)$$

Proposition 2.1.5. *A curve given by a Weierstrass equation is smooth if and only if* $\Delta \neq 0$.

Definition 2.1.6. Let E be an elliptic curve over a field k. The *function field* of E is the set of all rational functions defined on E. The *function field* of E is

$$k(E) = \{\frac{f}{g} | f, g \text{ homogeneous polynomials of the same degree and } g(P) \neq 0 \text{ for every } P \in E\}.$$

We define $E(k)$ to be the set of all the points of k together with O that belong to the elliptic curve E. The following very interesting results holds:

Theorem 2.1.7. *The set* $E(k)$ *is an abelian group with unit* O.

In particular, for an elliptic curve defined over a number field, $E(k)$ has the following property:

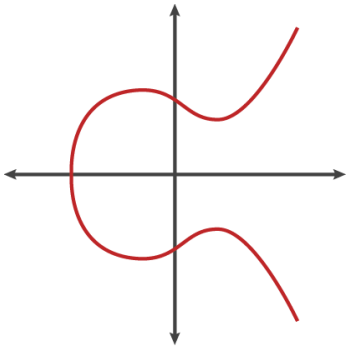
Theorem 2.1.8. (Mordell, Weil) *If* E *is an elliptic curve over a number field* k*, then* $E(k)$ *is a finitely generated abelian group.*

From the above we see that

$$E(\mathbb{Q}) = \mathbb{Z}^r \times E(\mathbb{Q})_{TORS},$$

where r is the rank of $E(\mathbb{Q})$ and $E(\mathbb{Q})_{TORS}$ is its *torsion subgroup*, i.e. the set of points of the group of finite order.

The group law of elliptic curves In order to describe the group law of an elliptic curve, we need a geometric insight into elliptic curves. An elliptic curve is symmetrical about the x-axis and looks like the following picture:



We also need a theorem due to Bezout which determines the number of points of intersection between two curves.

Theorem 2.1.9. (Bezout) *Let C_1, C_2 be two smooth projective curves without common components. Then C_1 and C_2 intersect in $(degC_1)(degC_2)$ number of points counted with multiplicities.*

In the definition of an elliptic curve we require that it has a rational point. The tangent line of the curve on this rational point is a rational line. It also intersects the curve in another point, as a consequence of Bezout's theorem (P is met with multiplicity two and there is a third point of intersection). This point will be rational, since solving the system between the curve and the line implies a 3rd degree equation, whose two solutions we already know and they are rational. In this way we can see that starting with one rational point on our curve, we can find a second.

Let E be an elliptic curve and P and Q be two rational points on E. Consider the line connecting P and Q. Again by Bezout's theorem, this line intersects E in a 3rd point. Let's call this point R. From R we bring a vertical line to the x-axis. This intersects E at another point, which we denote as $-R$, since it will be the inverse of R with respect to the group law we are trying to define. If $R = (x_R, y_R)$ and E is given by a short Weierstrass equation, then $-R = (x_R, -y_R)$. Then $P + Q = R$, where with "+" we will symbolize the addition law on elliptic curves, since the group is commutative.

What happens when we bring the tangent line to a point P? Then it intersects P with multiplicity 2 (or 3 in some cases), and by Bezout's theorem there is one more point on the intersection of this line and the elliptic curve. And, moreover, what happens when we bring the line passing from P and is vertical to the x-axis? This line intersects E at $P, -P$ and O, the point at infinity.

To prove that this is a group action is easy, except proving associativity which is quite complicated.

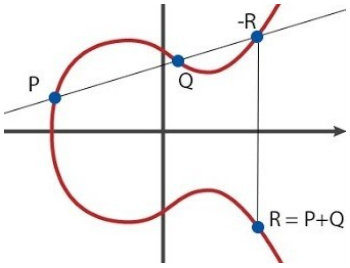
We now give the algebraic formulas that express the addition of two points $P = (x_P, y_P), Q = (x_Q, y_Q)$ on an elliptic curve $E : y^2 = x^3 + ax + b$. Let α be the slope of the line through P and Q.

- If $P \neq \pm Q$, then $\alpha = \frac{y_Q - y_P}{x_Q - x_P}$.
- If $P = Q$, then $\alpha = \frac{3x_P^2 + a}{2y_P}$.

Then $P + Q = (x_{P+Q}, y_{P+Q})$, where

$$\begin{cases} x_{P+Q} = \alpha^2 - x_P - x_Q \\ y_{P+Q} = -y_P - \alpha(x_{P+Q} - x_P) \end{cases}$$

The group law of elliptic curves in the general case is illustrated in the following picture:



Remark 2.1.10. For cryptographic purposes we use finite fields with large characteristic. So, in general, we will assume that the characteristic of the fields over which we are working is larger than 3, since the elliptic curves defined over them admit short Weierstrass equations and consequently the computations are easier. However, we will clarify when we do this assumption, since sometimes we will make statements and proofs for elliptic curves over fields of general characteristic.

2.2 Divisors and differentials

Definition 2.2.1. A *divisor* on C is a finite formal sum

$$D = \sum_{P \in C} n_P P,$$

where $n_P \in \mathbb{Z}$ and $n_P = 0$ for all but finitely many points $P \in C$.

The divisors on C form a free abelian group. We denote it as $Div(C)$.

Definition 2.2.2. The *degree* of a divisor is a map

$$\deg : Div(C) \to \mathbb{Z}$$

$$D = \sum_{P \in C} n_P P \mapsto \sum_{P \in C} n_P.$$

When we refer to the degree of a divisor, we usually mean its image.

Let $f \in k(C)$ be a rational function. Then $f(x) = g(x)/h(x)$, where $g(x), h(x)$ are homogeneous polynomials of the same degree. For a polynomial g and an elliptic curve C we denote as $I(C, g)$ the number of points of intersection of g and C, counted with multiplicity. For $I(C, g)$ the following rules apply:

1. $I(C, g) = 0$ if and only if C and g do not intersect.
2. $I(C, g) \geq 0$ always.

Definition 2.2.3. For every point $P \in C$ we define the order of f at P as

$$ord_P(C, f) = I(C, g) - I(C, h)$$

If $f \in k(C)$ a rational function, then we can define the divisor

$$divf = \sum_{P \in C} ord_P(f) P,$$

where $ord_P(f)$ is the order of P as a root or as a pole of f.

These divisors are called *principal* . The order $ord_P(f)$ is a valuation, so it has all the relative properties.

Remark 2.2.4. It is easy to see, using Bezout's theorem, that for a principal divisor $divf$ holds $\deg(divf) = 0$.

For a divisor D we define the set

$$\mathfrak{L}(D) := \{f \in K : divf + D \geq 0\} \cup \{0\}$$

Note that for a divisor $E = \sum_{P \in C} n_P P$ we say that $E \geq 0$ if $n_p \geq 0$, for every $P \in C$. We call a divisor with the above property *effective*.

Proposition 2.2.5. *$\mathfrak{L}(D)$ is a finite-dimensional k-vector space.*

Proof. Let $f \in \mathfrak{L}(D)$. Let $\lambda \in k^*$ and $P \in C$. Since $ord_P(f)$ is a valuation, we have that $ord_P(\lambda f) = ord_P(\lambda) + ord_P(f) = ord_P(f)$ and so $\lambda f \in \mathfrak{L}(D)$.
Now consider $g \in \mathfrak{L}(D)$. Again by the properties of a valuation we have that $ord_P(f+g) \geq min\{ord_P(f), ord_P(g)\}$ and so $f + g \in \mathfrak{L}(D)$. □

We notate as $l(D)$ the dimension of $\mathfrak{L}(D)$.

Remark 2.2.6. We notice that $\mathfrak{L}(0) = \{f \in K : divf \geq 0\} \cup \{0\}$, so it consists of all rational functions that have no poles and, since $\deg(divf) = 0$, they have no zeros either. Thus, $\mathfrak{L}(0)$ consists of all constant functions and so $l(0) = 1$.

Theorem 2.2.7. *Let E be an elliptic curve and let f and g be nonzero rational functions on E.*

1. *$div(f) = 0$ if and only id $f \in \bar{k}^*$*
2. *If $div(f) = div(g)$, then there is a nonzero constant c such that $f = cg$.*

Proof. 1. If $divf = 0$ then f has no poles, so the associated map

$$f : C \to \mathbb{P}^1, P \mapsto [f(P), 1]$$

is not surjective. Thus this map is constant, so $f \in \bar{k}^*$. The converse is clear.

2. We have that

$$divf = divg \Rightarrow divf - divg = 0 \Rightarrow div(f/g) = 0 \Rightarrow \frac{f}{g} = c \in \bar{k}^* \Rightarrow f = cg$$

□

Definition 2.2.8. The *space of differential forms* on C is the k-vector space Ω, generated by the set of symbols $\{df : f \in K\}$, subject to the relations

1. $d(f+g) = df + dg$
2. $d(fg) = fdg + gdf$
3. $da = 0$, for every $a \in k$

To clarify more what Ω is we consider the following

Proposition 2.2.9. *Ω is an one dimensional k-vector space.*

For $\omega \in \Omega$ and $P \in C$ we define the order $ord_P(\omega)$ as follows:
Let $t \in K$ be a local parameter at P, i.e. a meromorphic function on C that has a simple zero at P, and write $\omega = fdt$ for some other function f. We define

$$ord_P(\omega) = ord_P(f)$$

$$div(\omega) = \sum_{P \in C} ord_P(f)P \in Div(C)$$

On the set of divisors we consider the following equivalence relation:

$$D \sim D' \text{ if and only if } D - D' = divf \text{ for some } f \in K.$$

We notice that if $D \sim D'$ then

1. $\deg(D - D') = \deg(divf) = 0$, where $f \in K$ is such that $D - D' = divf$. Thus, $\deg(D) = \deg(D')$.
2. $l(D) = l(D')$.

Since Ω has dimension one, for every two differentials $\omega_1, \omega_2 \in \Omega$ holds that $\omega_1 = f\omega_2$ for some $f \in K$. Thus, divisors of all differentials lie in the same equivalence class, called the *canonical class*, and any $div\omega$ in it is called a *canonical divisor*.

Theorem 2.2.10. (Riemann-Roch) *Let C be a curve genus g over k and W a canonical divisor. Then, for every divisor D holds*

$$l(D) = \deg D + 1 - g + l(W - D).$$

By the Riemann-Roch theorem we conclude that for elliptic curves, i.e. genus 1 curves, holds

$$l(D) = \begin{cases} \deg D, \text{ if } D \geq 0, \\ 0 \text{ or } 1, \text{ if } \deg D = 0, \\ 0, \text{ if } \deg D < 0. \end{cases}$$

Proposition 2.2.11. *Canonical divisors have degree $2g - 2$, where g is the genus of the curve C.*

Theorem 2.2.12. *Let E be an elliptic curve. Let $D = \sum_{P \in E} n_P[P]$ be a divisor on E. Then D is the divisor of a rational function on E if and only if*

$$\deg(D) = 0 \text{ and } \sum(D) = O.$$

In particular, if a rational function on E has no zeros or no poles, then it is constant.

Example 2.2.13. *Suppose that $P \in E[m]$ is a point of order m. By definition, $mP = O$, so the divisor*

$$m[P] - m[O]$$

satisfies the conditions of theorem 2.2.12. Hence there is a rational function $f_P(X, Y)$ on E satisfying

$$div(f_P) = m[P] - m[O].$$

2.3 Elliptic curves over finite fields

In elliptic curve cryptography, elliptic curves over finite fields are used. For this reason we discuss more precisely about them. For what follows, let E be an elliptic curve over a field of characteristic $p > 0$ and set $q = p^r$ for some integer r.

Definition 2.3.1. The map

$$\phi_q : E \to E$$
$$(x, y) \mapsto (x^q, y^q)$$

is called the q^{th}-power *Frobenius endomorphism.*

The characteristic polynomial of the Frobenius map over the complex numbers is

$$\pi^2 - t\pi + p = 0,$$

where $t = p - |E(\mathbb{F})_p)|$

Theorem 2.3.2. (Hasse) *For the number of points of an elliptic curve E over a finite field $\mathbb{F}_q$ holds*

$$||E(\mathbb{F}_q)| - q - 1| \leq 2\sqrt{q}.$$

Remark 2.3.3. Hasse's theorem gives a bound for the number of points in $E(\mathbb{F}_q)$, but it does not provide a practical algorithm for computing $|E(\mathbb{F}_q)|$ when q is large.
The *discrete logarithm problem on elliptic curves*, translate into the following:

If $P, Q \in E(\mathbb{F}_q)$ and Q is in the subgroup of $E(\mathbb{F}_q)$ generated by P, find m such that $Q = [m]P$.

If q is small, we can compute $P, [2]P, [3]P, ...$ until we find Q, but for large q, it is difficult to find m. This allows us to use cryptosystems based on the difficulty of solving the discrete logarithm problems on elliptic curves.

Reduction modulo p and torsion subgroups Let p be a prime different from 2 and 3 and let $E : y^2 = x^3 + ax + b$ be an elliptic curve defined over $\mathbb{Q}$. Then,

$$\bar{E} : y^2 = x^3 + \bar{a}x + \bar{b}$$

is the reduction of E modulo p. If p does not divide the discriminant $\Delta = 4a^3 + 27b^2$, then $\bar{E}$ is an elliptic curve and we say that E has a good reduction at p.

Definition 2.3.4. Let E be a smooth elliptic curve over a field k. Consider the multiplication-by-m map $m : E \to E$ defined by $P \mapsto mP$, for every $P \in E$. The *m-torsion* of E is the subgroup $E[m]$ of E , where

$$E[m] = kerm = \{P \in E(\bar{k}) : mP = O\}.$$

Remark 2.3.5. The torsion subgroup $E(k)_{TORS}$ of an elliptic curve E over a field k is equal to

$$E(\mathbb{Q})_{TORS} = \bigcup_{m=1}^{\infty} E[m].$$

As a group, $E[m]$ is a $\mathbb{Z}$-module of rank 2, provided that the characteristic of k does not divide m. Thus,

$$E[m] \simeq \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}.$$

2.4 Isogenies

In almost all branches of mathematics we find structure-preserving maps, which are very useful. In analysis these are continuous functions, in group theory homomorphisms, etc. For elliptic curves, isogenies are the structure-preserving maps.

Let $E_1 = k(x, y)/(y^2 = x^3 + a_1x + b_1)$ and $E_2 = k(x, y)/(y^2 = x^3 + a_2x + b_2)$ be two elliptic curves defined over a finite field F_q of characteristic $p \neq 2, 3$.

Definition 2.4.1. Let $f : E_1 \to E_2$ be a rational map between elliptic curves such that

1. $f(O_{E_1}) = O_{E_2}$.
2. f is not trivial, i.e. there exist $P \in E_1$ such that $f(P) \neq O_{E_2}$.

Then f is called an *isogeny.*

Remark 2.4.2.

- Isogenies are surjective.
- Being isogenous is an equivalence relation and so we have equivalence classes of isogenous curves.
- An isogeny between two elliptic curves $f : E_1 \to E_2$ is a group homomorphism. Thus,

  $$f(P + Q) = f(P) + f(Q),$$

  for every $P, Q \in E_1$. But an even stronger result holds:

  $$Hom(E_1, E_2) = \{\text{isogenies } E_1 \to E_2\}.$$

  We define addition on $Hom(E_1, E_2)$ as

  $$(f + g)(P) = f(P) + g(P),$$

  where $f, g \in Hom(E_1, E_2)$ and $P \in E_1$. With this addition, $Hom(E_1, E_2)$ is a group. If $E_1 = E_2 = E$, we let

  $$End(E) = Hom(E, E)$$

  be the *endomorphism ring* of E, with addition as defined before and multiplication the composition of isogenies.

Example 2.4.3. *1. The multiplication-by-m map is an isogeny.*

2. The Frobenius map $E \to E, (x, y) \mapsto (x^q, y^q)$ is an isogeny.

Theorem 2.4.4. (Tate) *Two curves E_1, E_2 are isogenous over $\mathbb{F}_q$ if and only if $|E_1(\mathbb{F}_q)| = |E_2(\mathbb{F}_q)|$.*

If we look at E_1, E_2 as curves on the projective plane then

$$f(x : y : z) = (P_1(x, y, z), P_2(x, y, z), P_3(x, y, z)),$$

where $P_1(x, y, z)$, $P_2(x, y, z)$, $P_3(x, y, z)$ are homogeneous polynomials of the same degree.

Degree and kernel of an isogeny Let E_1 and E_2 be two elliptic curves defined over a field k and let $f : E_1 \to E_2$ be an isogeny between them.

Consider the fields of rational functions of E_1 and E_2, which are $k(E_1) = k(x, y)/(y^2 = x^3 + a_1x + b_1)$ and $k(E_2) = k(x, y)/(y^2 = x^3 + a_2x + b_2)$, respectively. Then f implies a map between these fields,

$$f^* : k(E_2) \to k(E_1).$$

This map is in fact a homomorphism, which in addition is injective.

Definition 2.4.5. We define the degree of an isogeny as

$$\deg f = [k(E_1) : f^*k(E_2)].$$

For every isogeny $f : E_1 \to E_2$, there exists a unique isogeny $\hat{f} : E_2 \to E_1$ such that $\hat{f} \circ f = [\deg f]$. If $f = [0]$, we set $\hat{f} = [0]$.

Definition 2.4.6. The isogeny $\hat{f}$ is called the *dual* isogeny of f.

The existence of the dual isogeny is in fact what makes the relation "being isogenous" symmetric.

The next proposition contains some very useful properties of the dual isogeny.

Proposition 2.4.7. *Let*

$$f : E_1 \to E_2$$

be an isogeny.

1. $f \circ \hat{f} = [\deg f]$ *on* E_2.
2. *Let* $l : E_2 \to E_3$ *be another isogeny. Then*
$$\hat{l \circ f} = \hat{f} \circ \hat{l}.$$
3. *Let* $g : E_1 \to E_2$ *be another isogeny. Then*
$$\hat{f + g} = \hat{f} \circ \hat{g}.$$
4. *For all* $m \in \mathbb{Z}$,
$$\hat{[m]} = [m].$$
5. $\deg \hat{f} = \deg f$.
6. $\hat{\hat{f}} = f$.

Definition 2.4.8. We say that f is *separable* , *inseparable* or *purely inseparable* if the field extension $k(E_1)/f^*k(E_2)$ has the corresponding property.

Definition 2.4.9. The kernel of an isogeny $f : E_1 \to E_2$ is

$$kerf = \{P \in E_1(\bar{k}) : f(P) = 0\}.$$

Let $f : E_1 \to E_2$ be a nonzero isogeny. Then $kerf = f^{-1}(O)$ is a finite subgroup of E_1 and if f is a separable isogeny, then $|kerf| = degf$.

Example 2.4.10.

1. *The multiplication by* m *map has degree* m^2.
2. *The Frobenius map over* $\mathbb{F}_q$ *is a purely inseparable isogeny of degree* q.
3. *Let* E/K *be an elliptic curve and let* $Q \in E$. *We can define the translation-by-*Q *map as*
$$\tau_Q : E \to E, P \mapsto P + Q.$$
This map has an inverse, τ_{-Q}, *so it is an isomorphism. However it is not an isogeny, unless* $Q = O$.

Theorem 2.4.11. *The set of separable isogenies from E_1 is in one to one correspondence with the finite subgroups of E_1.*

$$\{\text{separable isogenies from } E_1\} \leftrightarrow \{\text{finite subgroups of } E_1\}$$

$$(f : E_1 \to E_2) \mapsto ker f$$

$$(E_1 \to E_1/G) \leftarrow G$$

(For a finite subgroup G of E_1, the quotient E_1/G has the structure of an elliptic curve, as we will see later.)

The question that arises with the previous theorem and is relevant to the algorithm we aim to study is: how can one given the finite subgroup G of an elliptic curve E construct an isogeny $E \to E/G$? This can be done with Velu's formula and will be discussed later in detail, since it plays an important role on the development of the SIDH algorithm.

The following propositions are used in the proof of a very important theorem for our scheme, theorem 2.4.16. What this theorem tells us is that for any given finite subgroup of an elliptic curve, there exists an isogeny having this subgroup as a kernel. This lies in the core of SIDH.

Proposition 2.4.12. *Let E_1 and E_2 be elliptic curves and let $f : E_1 \to E_2$ be a nonzero isogeny. The map*

$$ker f \to Aut(\bar{K}(E_1)/f^*\bar{K}(E_2), T \mapsto \tau_T^*,$$

where τ_T is the translation-by-T map and τ_T^* is the automorphism that τ_T induces on $\bar{K}(E_Q)$, is an isomorphism.

Proof. For the proof the reader can see [Sil09] III.4.10b. □

The next two propositions and the definition refer to curves in general and not only elliptic curves.

Proposition 2.4.13. *Let C_1/K and C_2/K be curves. Let $L \supset K(C_1)$ be a subfield of finite index containing K. Then there exist a smooth curve C'/K, unique up to K-isomorphism, and a non-constant map $f : C_1 \to C'$ defined over K such that $f^*K(C') = L$.*

Proof. For a proof we refer the reader to [Sil09] II.2.4c. □

Definition 2.4.14. Let $f : C_1 \to C_2$ be a non-constant map of smooth curves and let$P \in C_1$. The *ramification index* of f at P is

$$e_f(P) = ord_P(f^*t_{f(P)}),$$

where $t_{f(P)} \in K(C_2)$ is a uniformizer at $f(P)$. Note that $e_f(P) \geq 1$. We say that f is unramified at P if $e_f(P) = 1$, and that f is unramified if it is unramified at every point of C_1.

Proposition 2.4.15. *A map $f : C_1 \to C_2$ is unramified if and only if $|f^{-1}(Q)| = deg(f)$ for all $Q \in C_2$.*

Proof. [Sil09] II.2.7. □

We are now ready to prove the theorem.

Theorem 2.4.16. *Let E be an elliptic curve and F a finite subgroup of E. There exists a unique elliptic curve E' and a separable isogeny $f : E \to E'$ with kernel F.*

Proof. From the proposition 2.4.12 we know that each point $T \in F$ gives rise to an automorphism τ_T^* of $\bar{K}(E)$. Let $\bar{K}(E)^F$ be the subfield of $\bar{K}(E)$ fixed by every element of F. From Galois theory then we have that $\bar{K}(E)$ is a Galois extension of $\bar{K}(E)^F$ with Galois group isomorphic to F.

The field $\bar{K}(E)^F$ has transcendence degree one over $\bar{K}$, so from 2.4.13 there are a unique smooth curve C/K and a finite morphism $f : E \to C$ satisfying $f^*\bar{K}(C) = \bar{K}(E)^F$.

We next show that f is unramified. Let $P \in E$ and $T \in F$. Then for every function $g \in \bar{K}(C)$.

$$g(f(P+T)) = (\tau_T^* \circ f^*)g(P) = (f^*g)(P) = gf(P)),$$

where the middle equality uses the fact that τ_T^* fixes every element of $f^*\bar{K}(C)$. It follows that $f(P+T) = f(P)$. Now let $Q \in C$ and choose any point $P \in E$ with $f(P) = Q$. Then

$$f^{-1}(Q) \supset \{P+T : T \in F\}.$$

However, we also know from 2.4.15 that

$$|f^{-1}(Q)| \leq degf = |F|$$

with equality if and only if f is unramified. Since the points $P+T$ are distinct as T ranges over that elements of F, we conclude that f is unramified at Q. And since Q was arbitrary, the map f is unramified. Finally, we apply 2.2.11 combined with 2.4.15 to f. Since f is unramified, the formula reads

$$2genus(E) - 2 = (degf)(genus(C) - 2).$$

From this we conclude that C also has genus one, and hence C becomes an elliptic curve and f becomes an isogeny if we take $f(O)$ to be the zero point on C. □

3 The SIDH

Luca de Feo, David Jao and Jerome Plut in [DFJP14] proposed a public key cryptosystem that is believed to be quantum resistant. This cryptosystem is analogous to Diffie-Hellman key-exchange and is based on the difficulty of computing isogenies between supersingular elliptic curves. More precisely, the "hard" problem is the following:

Definition 3.0.1. *(Supersingular isogeny problem).* Given a finite field k and two supersingular elliptic curves E_1, E_2 defined over k such that $|E_1| = |E_2|$, compute an isogeny $f : E_1 \to E_2$.

The first to propose an algorithm based on isogenies (IDH) is Stolbunov, who uses ordinary elliptic curves. However, an attack has been found against IDH thats takes subexponential time on a quantum computer, as opposed to this algorithm that the fastest known attack requires exponential time.

In the supersingular case, the endomorphism ring is not commutative. This is the main technical difficulty in creating a Diffie-Hellman type protocol using supersingular isogenies. We see how Jao, De Feo and Plut overcame this obstacle by giving some additional information which do not appear to make the problem of finding isogenies (on which their protocol is based) any easier.

3.1 The key exchange

Alice and Bob want to compute a common key in order to communicate secretly with each other. They start with a public supersingular elliptic curve and they both end up to the same isogenous curve to it, following different walks on the isogeny graph. Every supersingular elliptic curve in characteristic p is defined over $\mathbb{F}_p$ or $\mathbb{F}_{p^2}$, so it suffices to fix $\mathbb{F}_q = \mathbb{F}_{p^2}$ as the field of definition for the algorithm that follows.

We fix small primes l_A, l_B different from each other, integers e_A, e_B and we pick a number f such that $p = {l_A}^{e_A} {l_B}^{e_B} f \pm 1$ is prime. Our field of definition is $\mathbb{F}_q = \mathbb{F}_{p^2}$. Next we fix a supersingular curve E defined over $\mathbb{F}_q$.

Alice and Bob pick bases $\{P_A, Q_A\}$ and $\{P_B, Q_B\}$ of $E[{l_A}^{e_A}]$ and $E[{l_B}^{e_B}]$, respectively. Then Alice picks integers $\alpha_1, \alpha_2 \in \mathbb{Z}/{l_A}^{e_A}\mathbb{Z}$, not both divisible by l_A, and computes $K_A =< \alpha_1 P_A + \alpha_2 Q_A >$ and the isogeny $a : E \to E_a$, where $E_a = E/ < K_A >$. Bob acts analogously: he picks integers $\beta_1, \beta_2 \in \mathbb{Z}/{l_B}^{e_B}\mathbb{Z}$, not both divisible by l_B, and computes $K_B =< \beta_1 P_B + \beta_2 Q_B >$ and the isogeny $b : E \to E_b$, where $E_b = E/ < K_B >$.

Alice and Bob want to find the same isogenous curve to E, the j-invariant of which is going to be their shared secret key. So Alice computes

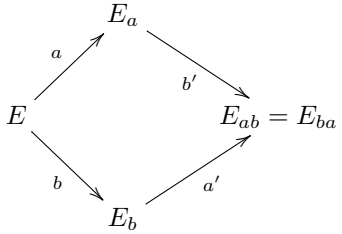
$$E_{ab} = E_b/ < [\alpha_1]b(P_A) + [\alpha_2]b(Q_A) >$$

and Bob computes

$$E_{ba} = E_a/ < [\beta_1]a(P_B) + [\beta_2]a(Q_B) >$$

.

The next graph presents the isogenies computed by Alice and Bob in the process of finding their common shared key.



Proposition 3.1.1. *With the above notation, $E_{ab} = E_{ba}$.*

Proof. We have that

$$E_{ab} = E_b/ < [\alpha_1]b(P_A) + [\alpha_2]b(Q_A) >,$$

where $E_b = E/ < K_B >$ and $K_B =< \beta_1 P_B + \beta_2 Q_B >$.

Since $\{P_B, Q_B\}$ is a basis of $E[{l_B}^{e_B}]$, K_B has order ${l_B}^{e_B}$ and since $\{P_A, Q_A\}$ is a basis of $E[{l_A}^{e_A}]$, K_A has order ${l_A}^{e_A}$.

The image of K_A under the isogeny b is

$$b(K_A) = b([\alpha_1]P_A + [\alpha_2]Q_A) = [\alpha_1]b(P_A) + [\alpha_2]b(Q_A).$$

The isogeny b has degree ${l_B}^{e_B}$ and because l_A, l_B are different primes, $b(K_A)$ has order ${l_A}^{e_A}$. Thus, we see that K_B and $[\alpha_1]b(P_A) + [\alpha_2]b(Q_A)$ are disjoint and so

$$E_{ab} = E/\{K_B, b(K_A)\}.$$

In the same way,

$$E_{ba} = E/\{K_A, a(K_B)\}.$$

Now notice that

- $b(K_A) \simeq K_A$
- $a(K_B) \simeq K_B$

Indeed, by the isomorphism theorems we have that $E/ker b \simeq Im b$. Since all the elements in $ker b$ have order a power of l_B and all the elements in K_A have order a power of l_A, with l_B and l_A different primes, we have that $ker b$ and K_A are two disjoint subgroups of E. Thus, the restriction of b on K_A has trivial kernel and the same isomorphism theorem gives us that $K_A \simeq b(K_A)$. Equivalent arguments are used to prove that $a(K_B) = K_B$.
Hence

$$E_{ab} = E/\{K_B, b(K_A)\} = E/\{K_B, K_A\} = E/\{a(K_B), K_A\} = E_{ba}.$$

□

Public parameter creation	
A trusted party chooses and publishes a large prime p of the form ${l_A}^{e_A}{l_B}^{e_B} f \pm 1$, where l_A, l_B are different primes, and a supersingular elliptic curve E over $\mathbb{F}_p$.	
Private computations	
Alice	**Bob**
Computes a basis $\{P_A, Q_A\}$ of $E[{l_A}^{e_A}]$.	Computes a basis $\{P_B, Q_B\}$ of $E[{l_B}^{e_B}]$.
Chooses secret integers n_A, m_A.	Chooses secret integers n_B, m_B.
Computes an isogeny $a : E \to E_a$ with kernel $K_A =< n_A P_A + m_A Q_A >$.	Computes an isogeny $b : E \to E_b$ with kernel $K_B =< n_B P_B + m_B Q_B >$.
Computes $a(P_B), a(Q_B)$.	Computes $b(P_A), b(Q_A)$.
Public exchange of values	
Sends $E_a, a(P_B), a(Q_B)$ to Bob	Sends $E_b, b(P_A), b(Q_A)$ to Alice
Further private computations	
Alice	**Bob**
Computes an isogeny $a' : E_b \to E_{ab}$ with kernel $< [n_A]b(P_A) + [m_A]b(Q_A) >$.	Computes an isogeny $b' : E_a \to E_{ba}$ with kernel $< [n_B]a(P_B) + [m_B]a(Q_B) >$.
The shared secret value is $E_{ab} = E/\{K_B, b(K_A)\} = E/\{K_B, K_A\} = E/\{a(K_B), K_A\} = E_{ba}$.	

Table 4: The SIDH.

3.2 Computing isogenies and isogenous curves

Let k be a field with characteristic larger than 3, E_1, E_2 be two elliptic curves in short Weierstrass equations and $f : E_1 \to E_2$ be an isogeny. We express f in the following way:

$$f(x, y) = (R_1(x, y), R_2(x, y))$$

where R_1, R_2 are rational functions.

Lemma 3.2.1. *$f(x, y) = (r_1(x), yr_2(x))$, for some rational functions $r_1(x), r_2(x)$.*

Proof. Let $P = (x, y)$ be a point of E_1. Then, since we are on characteristic larger that 3, we have that $-P = -(x, y) = (x, -y)$. Moreover, since f is an isogeny,

$$f(-P) = -f(P). \qquad (*)$$

We express $R_1(x, y)$ and $R_2(x, y)$ as

$$R_1(x, y) = R_{11}(x) + yR_{12}(x)$$

$$R_2(x, y) = R_{21}(x) + yR_{22}(x)$$

We can do that because x and y lie in a relation that is quadratic to y (the equation of the elliptic curve). From $(*)$ we get that

$$\begin{cases} R_1(x, -y) = R_1(x, y) \\ R_2(x, -y) = -R_2(x, y) \end{cases} \Rightarrow \begin{cases} R_{11}(x) - yR_{12}(x) = R_{11}(x) + yR_{12}(x) \\ R_{21}(x) - yR_{22}(x) = -R_{21}(x) - yR_{22}(x) \end{cases} \Rightarrow \begin{cases} 2yR_{12}(x) = 0 \\ 2R_{21}(x) = 0 \end{cases} \Rightarrow$$

$$\begin{cases} y = 0 or R_{12}(x) = 0 \\ R_{21}(x) = 0. \end{cases}$$

Since P was a random element of $E_1(\bar{k})$, we conclude that $R_{12}(x) = 0$.
Hence,

$$f(x, y) = (r_1(x), yr_2(x)),$$

for some rational functions $r_1(x), r_2(x)$. □

Velu's formula In order to exchange messages using SIDH, Alice and Bob should be able, given an elliptic curve, to compute isogenous elliptic curves to it. To do that, they are using Velu's formula.
Before presenting this formula, we discuss a bit more about elliptic curves and how they are defined.

Let $E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ be an elliptic curve over a field k and consider the functions x, y in the function field $k(E)$ of E that satisfy the following conditions:

1. $\upsilon_{0_E}(x) = -2, \upsilon_{0_E}(y) = -3, \frac{y^2}{x^3}(0_E) = 1,$
2. $\upsilon_P(x) \leq 0, \upsilon_P(y) \leq 0,$

where $\upsilon_P(f)$ is the valuation of a function f in the function field of E on a point $P \in E$.

Proposition 3.2.2. *Such functions exist.*

Proof. To prove that these functions exist, one can use Riemann-Roch theorem. Let $D = (0_E)$ be a divisor on E. Then,

$$l(D) = degD + 1 - g + l(W - D),$$

where W is a canonical divisor on E.
Since E is an elliptic curve, it has genus 1 and $l(W - D) = l(W) = 2g - 2 = 0$, by Proposition 2.2.11. So from Riemann-Roch we get

$$l(D) = degD = 1$$

Thus, $\mathfrak{L}(D) = k$.
Now take $D = 2(0_E)$. In this case, $l(D) = degD = 2$ and so $\mathfrak{L}(D) = k \oplus kx$, for some $x \notin k$. Since

$x \in \mathfrak{L}(D)$ we have that $\upsilon_{0_E}(x) \geq -2$. But since $x \notin k = \mathfrak{L}(0_E)$, we have that $\upsilon_{0_E}(x) < -1$. Hence $\upsilon_{0_E}(x) = -2$.
We continue and consider the divisor $3(0_E)$. In the same way we prove that $l(D) = degD = 3$ and so $\mathfrak{L}(D) = k \oplus kx \oplus ky$, for some $y \notin k$ and $y \notin kx$ with $\upsilon_{0_E}(y) = -3$.

□

Moreover, the functions x, y generate the function field of E.

Lemma 3.2.3. *For the function x, y as defined above holds that they lie in a relation of degree 3.*

Proof. Set $z = -x/y$. Then

$$\upsilon_0(z) = \upsilon_0(x) - \upsilon_0(y) = -2 - (-3) = 1.$$

Since x, y are rational functions, they are meromorphic, and so each can be expressed by a Laurent series:

$$x = a_0 z^{-2} - az^{-1} - b - cz - dz^2 - ez^3 + ...$$
$$y = -a_0 z^{-3} + az^{-2} + bz^{-1} + c + dz + ez^2 + fz^3 + ...$$

We have that $xz^2 = a_0 - az - bz^2 - ...$ and so if we evaluate this at 0 we have $xz^2(0) = a_0$. But $z = -x/y$ and so $xz^2 = x^3/y^2$ and since $\frac{y^2}{x^3}(0_E) = 1$, we conclude that $a_0 = 1$. Thus, the analytic expansions of x and y are

$$x = z^{-2} - az^{-1} - b - cz - dz^2 - ez^3 - ..$$
$$y = -z^{-3} + az^{-2} + bz^{-1} + c + dz + ez^2 + fz^3 + ...$$

Combining the above relations and doing some computations, we find that x, y lie in the relation

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \tag{1}$$

where

$$a = a_1, b = a_2, c = a_3, d = a_1a_3 + a_4, e = a_2a_3 + a_1^2a_3 + a_1a_4,$$
$$f = a_1^2a_4 + a_1^3a_3 + a_2a_4 + 2a_1a_2a_3 + a_3^2 + a_6.$$

□

One can define the *discriminant* Δ of the above equation in the following way: set

$$b_2 = a_1^2 + 4a_2, b_4 = a_1a_3 + 2a_4, b_6 = a_3^2 + 4a_6, b_8 = a_1^2a_6 - a_1a_3a_4 + 4a_2a_6 + a_2a_3^3 - a_4^2$$

Then, $\Delta = -b_2^3b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6$.

The relation (1) is non-singular, if the discriminant Δ is not 0.

Conversely, if we are given 5 elements a_1, a_2, a_3, a_4, a_5 of k for which $\Delta \neq 0$, the above equation defines an elliptic curve in $\mathbb{P}_2$ and, if we take as O as the point at infinity, the functions x, y satisfies the conditions

1. $\upsilon_{0_E}(x) = -2, \upsilon_{0_E}(y) = -3, \frac{y^2}{x^3}(0_E) = 1,$
2. $\upsilon_P(x) \leq 0, \upsilon_P(y) \leq 0.$

Theorem 3.2.4. (Velu's formula) *Let $E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ be an elliptic curve and set $G(x, y) = x^3 + a_2x^2 + a_4x + a_6 - y^2 - a_1xy - a_3y$. Consider a finite subgroup F of E and the functions X and Y that take the following values for every $P \in E$*

$$\begin{cases} X(P) = x(P) + \sum_{Q \in F-0}[x(P+Q) - x(Q)] \\ Y(P) = y(P) + \sum_{Q \in F-0}[y(P+Q) - y(Q)] \end{cases}$$

Define a set S as follows: Denote as F_2 the 2-torsion points of $F - \{0\}$. Then consider a subset R of $F - \{0\} - F_2$ such that

- $R \cup (-R) = F - \{0\} - F_2$
- $R \cap (-R) = \emptyset$

We define $S := F_2 \cup R$. Using the addition law of elliptic curves, from the above formulas obtain the next more general formulas

$$\begin{cases} X = x + \sum_{Q \in S} [\frac{t_Q}{x - x_Q} + \frac{u_Q}{(x - x_Q)^2}] \\ Y = y - \sum_{Q \in S} [u_Q \frac{2y + a_1 x + a_3}{(x - x_Q)^3} + t_Q \frac{a_1(x - x_Q) + y - y_Q}{(x - x_Q)^2} + \frac{a_1 u_Q - G_x(Q) G_y(Q)}{(x - x_Q)^2}] \end{cases}$$

where

$$\begin{cases} Q = (x_Q, y_Q) \\ G_x(Q) = \frac{\partial G}{\partial x}(Q) \\ G_y(Q) = \frac{\partial G}{\partial y}(Q) \\ t_Q = \begin{cases} G_x(Q), if Q \in F_2 \\ 2G_x(Q) - a_1 G_y(Q), if Q \notin F_2 \end{cases} \\ u_Q = (G_y(Q))^2 \end{cases}$$

Then

1. *The isogeny $f : E \to E/G$ is given by $(x, y) \mapsto (X, Y)$, where x, y are the generators of the function field of E as described above.*
2. *The elliptic curve E/G admits the equation*

$$Y^2 + A_1 XY + A_3 Y = X^3 + A_2 X^2 + A_4 X + A_6$$

where

$$A_1 = a_1, A_2 = a_2, A_3 = a_3,$$
$$A_4 = a_4 - 5t, A_6 = a_6 - (a_1^2 + 4a_2)t - 7w,$$

where $t = \sum_{Q \in S} t_Q$ and $w = \sum_{Q \in S} u_Q + x_Q t_Q$.

Proof. To prove formula (2) by formula (1) we just use the addition law of elliptic curves. We consider 2 cases.

- If Q has order 2, then $x(P + Q) - x(Q) = \frac{t_Q}{x - x_Q}$ and $y(P + Q) - y(Q) = -\frac{a_1(x - x_Q) + y - y_Q}{(x - x_Q)^2}$ and $u_Q = 0$.
- If Q does not have order 2, then $x(P+Q) - x(Q) + x(P-Q) - x(-Q) = \frac{t_Q}{(x - x_Q)^2} + \frac{u_Q}{(x - x_Q)^3}$ and $y(P+Q) - y(Q) + y(P-Q) - y(-Q) = -u_Q \frac{2y + a_1 x + a_3}{(x - x_Q)^3} - t_Q \frac{a_1(x - x_Q) + y - y_Q}{(x - x_Q)^2} - \frac{a_1 u_Q - G_x(Q) G_y(Q)}{(x - x_Q)^2}$

1. i) $X, Y \in k(E')$.

ii) $\upsilon_{0_{E'}}(X) = -2, \upsilon_{0_{E'}}(Y) = -3$. Moreover, one can check that $\frac{Y^2}{X^3}(0_{E'}) = 1$, $\upsilon_{P'}(X), \upsilon_{P'}(Y) \geq 0$ for every $P' \neq 0_{E'}$.

The above indicate that $k(E') \simeq k(X, Y)$. And the isogeny f is given by the transformation $(x, y) \mapsto (X, Y)$.

□

It's important to note here that although the formulas (2) are more general than the formulas (1), it is more efficient computationally to use the formulas (1) when applying our algorithm, i.e. computing the image of every point of E on the isogenous E' is better than computing explicitly the equation of the isogeny between them.

A more refined formula Velu's formula for the isogeny between E and E/G has been very useful and his proof, made entirely without the use of a computer, is very delicate. Velu's formula is also used in Schoof's algorithm, the first deterministic algorithm for counting the points of an elliptic curve over finite fields. In his try to improve Schoof's algorithm, Elkies expressed Velu's formula in a more convenient way. He assumed that the characteristic of the field is larger than 3, which allows us to express elliptic curves in short Weierstrass equations. Since for cryptographic

purposes in practice we will always use fields with large characteristics, we can also assume this here.

Assume that $p \geq 3$. Then, an elliptic curve E over F_{p^n} is isomorphic to the following Weierstrass form

$$E : y^2 = f(x) = x^3 + ax + b$$

We see that since E is in short Weierstrass form, Velu's formula takes the following form

$$X = x + \sum_{Q \in G^*} \frac{f'(x)}{x - x(Q)} + \frac{2f(Q)}{(x - x(Q))^2}$$

$$Y = y + \sum_{Q \in G^*} \frac{yf'(x)}{(x - x(Q))^2} + \frac{4yf(Q)}{(x - x(Q))^3}$$

The reformulation given by Elkies is

$$X = x + \sum_{Q \in G^*} [x - x(Q) - \frac{f'(x)}{x - x(Q)} + \frac{2f(x)}{(x - x(Q))^2}]$$

3.3 The isogenies of Alice and Bob

For the key exchange, Alice and Bob have to compute the isogeny with kernel a certain finite subgroup of an elliptic curve. We will see in more detail how they do that. We will discuss about the computations to which Alice has to proceed, since Bob acts analogously.

As described above, Alice picks $P_A, Q_A \in E[l_A^{e_A}]$, Bob picks $P_B, Q_B \in E[l_B^{e_B}]$ and they publish them. The primes l_A and l_B have to be small primes different from each other. So we can assume that $l_A = 2$ and $l_B = 3$ and that's what we will do from now on. Furthermore, for simplicity of notation, we call $e_A = n$ and $e_B = m$. Next, Alice picks secret numbers α_1, α_2 and computes $K_A = \alpha_1 P_A + \alpha_2 Q_A$. For further simplicity of notation in this paragraph we call $K_A = T$. She wants to compute the isogeny $a : E \to E_a$, where $E_a = E/ < T >$, and the values $a(P_B)$ and $a(Q_B)$.

In order to compute the isogenies a and b Alice and Bob use Velu's formula. The isogeny a is of degree 2^n and requires time $O(2^n)$ to be computed. Since n is big for security reasons, it is difficult to compute a. To overcome this obstacle, Alice splits a into n isogenies a_i, each of degree 2, which are computed in time $nO(2)$, and has $a = a_n \circ a_{n-1} \circ ... \circ a_1$. Velu's formula is used by both of them in order to compute all the intermediate curves on the isogeny graph.

We show a way that the isogenies a_i can be computed.

Step 1
Compute $T_1 = 2^{n-1}T$, an element of E of order 2. The isogeny of kernel T_1 is a_1 (notice that a_1 has indeed order 2).

Lemma 3.3.1. *The element $a_1(T)$ has order 2^{n-1}.*

Proof. Let m be an integer such that $ma_1(T) = 0$. Since for every integer l holds that $la_1(T) = a_1(lT)$, we have:

$$ma_1(T) = 0 \Leftrightarrow mT \in kera_1 \Leftrightarrow mT \in < T_1 > \Leftrightarrow mT \in < 2^{n-1}T > \Leftrightarrow 2^{n-1} | m$$

□

Step 2
Set $T_2 = 2^{n-1}a_1(T)$, which again is of order 2. Compute the isogeny a_2 with kernel $< T_2 >$. Compute also $E_2 = E_1/ < T_2 >, a_2 \circ a_1(Q_1), a_2 \circ a_1(Q_2), a_2 \circ a_1(T)$.

Lemma 3.3.2. *1. $a_2 \circ a_1(T)$ is of order 2^{n-2}.*

2. $ker(a_2 \circ a_1) =< 2^{n-2}T >$.

Proof. $x \in ker(a_2 \circ a_1) \Leftrightarrow a_2 \circ a_1(x) = 0 \Leftrightarrow a_1(x) \in kera_2 =< T_2 > \Leftrightarrow a_1(x) \in < 2^{n-2}a_1(T) > \Leftrightarrow x \in < 2^{n-2}T >$. □

Step 3
$T_3 = 2^{n-3}a_2 \circ a_1(T)$ is of torsion 2. a_3 is the isogeny with kernel $< T_3 >$. Compute $E_3 = E_2/ < T_3 >, a_3 \circ a_2 \circ a_1(Q_1), a_3 \circ a_2 \circ a_1(Q_2), a_3 \circ a_2 \circ a_1(T)$.

Lemma 3.3.3. *1. $a_3 \circ a_2 \circ a_1(T)$ is of order 2^{n-3}.*

2. $ker(a_3 \circ a_2 \circ a_1) =< 2^{n-3}T >$

Proof. The proof of this lemma is similar to the one of the previous lemma. □

. . .

Step n
$T_n = 2^{n-n}a_{n-1} \circ a_{n-2} \circ ... \circ a_1(T) = a_{n-1} \circ a_{n-2} \circ ... \circ a_1(T)$ is of order 2. a_n is the isogeny with kernel $< T_n >$. Compute $a_n \circ a_{n-1} \circ ... \circ a_1(Q_1) = a_(Q_1)$ and $a_n \circ a_{n-1} \circ ... \circ a_1(Q_2) = a_(Q_2)$. (We don't need to compute $a_n \circ a_{n-1} \circ ... \circ a_1(T)$ cause we already know it: it's the kernel.)

Lemma 3.3.4. *1. $a(T) = a_n \circ ... \circ a_2 \circ a_1(T)$ has order 1.*

2. $ker(a_n \circ ... \circ a_2 \circ a_1) =< T >$.

Proof. The proof of this lemma is again similar to the previous proofs. □

3.4 Finding bases of torsion subgroups

At the key-exchange, Alice and Bob need to find a basis of $E[l_A^{e_A}]$ and $E[l_B^{e_B}]$, respectively. We give some insight into the algorithmic details of how they do that.

Let E/k be an elliptic curve and $m \geq 2$ an integer prime to the characteristic p of the field k. Then, the torsion subgroup $E[m]$ is a free $\mathbb{Z}$-module of rank two,

$$E[m] \simeq \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$$

We will define a pairing on $E[m]$, the *Weil-pairing*, which Alice and Bob use in order to check the independence of the points they pick in their try to form the bases.

Let $T \in E[m]$. There is a function $f \in \bar{k}(E)$ satisfying

$$div(f) = m(T) - m(O).$$

Next take $T' \in E$ to be a point such that $[m]T' = T$. Then there is similarly a function $g \in \bar{k}(E)$ satisfying

$$div(g) = [m]^*(T) - [m]^*(O) = \sum_{R \in E[m]} (T' + R) - (R).$$

This is indeed a principal divisor, since $|E[m]| = m^2$ and $[m^2]T = O$. It is easy to verify that the functions $f \circ [m]$ and g^m have the same divisor, so multiplying f by an appropriate constant from $\bar{k}^*$, we may assume that

$$f \circ [m] = g^m.$$

Now let $S \in E[m]$ be another m-torsion point. Then, for any point $X \in E$, we have

$$g(X + S)^m = f([m]X + [m]S) = f([m]X) = g(X)^m.$$

Thus, consider as a function on X, the function $g(X+S)/G(X)$ takes on only finitely many values, i.e. for every X it is an m^{th} root of unity. In particular, the morphism

$$E \to \mathbb{P}^1, S \mapsto g(X + S)/g(X)$$

is not surjective, and thus it is constant. So we are ready to define the Weil-pairing.

Definition 3.4.1. *(Weil pairing).* Let μ denote the group of m^{th} roots of unity. The Weil pairing is the map

$$e_m : E[m] \times E[m] \to \mu_m, (S, T) \mapsto g(X + S)/g(X),$$

where $X \in E$ is any point such that $g(X + S)$ and $g(X)$ are both defined and non-zero.

Another way to do it is to try to solve the DLP of Q to the base P, which is easy in a group of order 2^n. If the algorithm fails then the points are independent. In particular, to show that $< P, Q >= E[2^n]$ it suffices to compute $[2^{n-1}]P$ and $[2^{n-1}]Q$ and verify that these points are both different, and neither is the identity.

However the Weil pairing can be used to check a lot more than just independence: It allows Alice to validate that the points provided by Bob are the images of the correct points under an isogeny of the correct degree. She can do this because of the following result:

Proposition 3.4.2. *Let $f : E \to E'$ be an isogeny and $P, Q \in E[N]$ for some integer N. Then*

$$e_N(f(P), f(Q)) = e_N(P, Q)^{deg(f)}$$

To prove this proposition we need to combine the following two lemmas:

Lemma 3.4.3. *The Weil pairing is bilinear, i.e.*

$$e_N(P_1 + P_2, Q) = e_N(P_1, Q)e_N(P_2, Q)$$

$$e_N(P, Q_1 + Q_2) = e_N(P, Q_1)e_N(P, Q_2).$$

Proof. We will only prove the linearity in the first factor, since it is the one we are going to need. We have

$$e_N(P_1 + P_2, Q) = \frac{g(X + P_1 + P_2)}{g(X)} = \frac{g(X + P_1 + P_2)}{g(X + P_1)} \frac{g(X + P_1)}{g(X)} = e_N(P_2, Q)e_N(P_1, Q).$$

□

Lemma 3.4.4. *Let $\phi : E \to E'$ be an isogeny of elliptic curves. Then for all m-torsion points $S \in E(m)$ and $T \in E'(m)$,*

$$e_m(S, \hat{\phi}(T)) = e_m(\phi(S), T)$$

Proof. We have as before that $divf = m(T) - m(O)$ and $f \circ [m] = g^m$. Then

$$e_m(\phi Q, T) = \frac{g(X + \phi S)}{g(Q)}.$$

Choose a function $h \in \bar{k}(E)$ satisfying

$$\phi^*((T)) - \phi^*((O)) = (\hat{\phi}T) - (O) + divh.$$

Now we observe that

$$div(\frac{f \circ \phi}{h^m}) = \phi^* div(f) - mdiv(h) = m(\hat{\phi}T) - m(O)$$

and

$$(\frac{g \circ \phi}{h \circ [m]})^m = \frac{f \circ [m] \circ \phi}{(h \circ [m])^m} = (\frac{f \circ \phi}{h^m}) \circ [m].$$

Then directly from the definition of the Weil pairing we obtain

$$e_m(S, \hat{\phi}T) = \frac{(g \circ \phi / h \circ [m])(X + S)}{(g \circ \phi / h \circ [m])(X)} = \frac{g(\phi X + \phi S)}{g\phi X} \frac{h([m]X)}{h([m]X + [m]S)} = e_m(\phi S, T).$$

□

Now we can give the proof of 3.4.2:

Proof. We apply 3.4.4 with $S = f(P)$, $T = Q$ and $\hat{\phi} = f$ and we have

$$e_m(f(P), f(Q)) = e_m(\hat{f}(f(P)), Q) = e_m([degf]P, Q) = e_m(P, Q)^{[degf]},$$

where the last equality derives by 3.4.3. □

So a validation step for Alice to run is to check whether

$$e_{2^n}(b(P_A), b(Q_A)) = e_{2^n}(P_A, Q_A)^{3^m}$$

and if it's true she will gain some assurance that the points $b(P_A), b(Q_A)$ that Bob sent to her are the images of the correct points under an isogeny of the correct degree.

So we now present the steps that Alice has to follow in order to find a basis for the torsion subgroup $E[2^n]$, with Bob acting accordingly. Alice chooses a random point $P \in E(\mathbb{F}_{p^2})$ and multiply it by $(3^m f)^2$ to obtain a point P' of order dividing 2^n (*Note:* remember that the elliptic curves of the protocol are defined over $\mathbb{F}_{p^2}$, where $p = 2^n 3^m f \pm 1$. This is the "f" we use here.) With high probability P' will have order exactly 2^n. She can check that by simply multiplying P' by powers of 2. If the check succeeds, then set $P_A = P'$. Otherwise try again with another P. A second point Q_A of order 2^n can be obtained in the same way. Then she has to check whether Q_A is independent of P_A. So she calculates the Weil-pairing $e(P_A, Q_A)$ in $E[2^n]$ and check that the result has order 2^n. This will again happen with high probability. If not, she chooses another point Q_A.

A double-and-add algorithm In order to compute the kernel of her isogeny $< \alpha_1 P_A + \alpha_2 Q_A >$, Alice can proceed naively and commute $[\alpha_1]P_A$ and $[\alpha_2]Q_A$. However, computing a multiple of a point on an elliptic curve is of high cost. Thus, it is better that she uses a double-and-add algorithm. A standard double-and-add algorithm for computing nP is the following:

Algorithm 1

Input: A number $n \in \mathbb{Z}$ and a point $P \in E$.
Output: The point $[n]P$.

1: Write the binary expansion of n as

$$n = e_0 + e_1 2 + e_2 2^2 + e_3 2^3 + ...e_t 2^t,$$

with $e_0, ..., e_t \in \{0, 1\}$ and $e_t = 1$.
2: Set $Q = P$ and set $R = O$, if $e_0 = 0$, or $R = P$ if $e_0 = 1$.
3: **for** $i = 1, 2, ...t$ **do**
4: Set $Q = [2]Q$.
5: **if** $e_i = 1$ **then**
6: set $R = R + Q$.
7: **return** R, which is equal to $[n]P$.

3.5 A zero-knowledge proof of identity

A zero-knowledge proof is a procedure that allows Peggy, the prover, to convince Victor, the verifier, that a certain fact is true without giving Victor any information that would let Victor convince other people that the fact is true. It seems at first glance to be impossible to convince someone you know a fact without revealing this fact, but there are ways that this can be done.

An example of a zero-knowledge proof that makes the concept easy to be understood even by children, is a story about the cave of Ali Baba: This cave has 2 passages that they both seem to have a dead end, but they are connected and by a door that opens by saying some secrets words. But how to prove to someone that you know the secret words without revealing them to that someone? And here goes the procedure: The prover enters the cave and then chooses one of the two passages without letting the verifier see which one. Then the verifier asks them to return randomly by the left or the right passage. If the verifier returns by the correct passage, it doesn't prove anything, since they could have entered by this and just be lucky. However, by repeating the procedure many times (in the story there are 40 thieves, so it's repeated 40 times), the probability to always come out by the correct passage is zero. This way the prover can convince the verifier that they know the secret words which open the door connecting the two passages, and thus can return by whichever passage they have been asked.

Algorithmically, a zero-knowledge proof generally involves a number of challenge–response communication rounds between Peggy and Victor. In a typical round, Victor sends Peggy a challenge, Peggy sends back a response, and then Victor evaluates the response and decides whether to accept or reject it. After a certain number of rounds, a good zero-knowledge proof showing that a quantity y has some property P should satisfy the following two conditions:

Completeness If y does have property P, then Victor should always accept Peggy's responses as being valid.

Soundness If y does not have property P, then there should be only a very small probability that Victor accepts all of Peggy's responses as being valid.

In addition to being both sound and complete, a zero-knowledge proof should not convey useful information to Victor, whence the name.

So we present the zero-knowledge proof of identity based on supersingular elliptic curves that Luca de Feo, David Jao and Jerome Plut propose.

Peggy knows a cyclic degree 2^n isogeny $a : E \to E/ < S >$. The curves $E, E/ < S >$ are publicly known. She wants to prove to Victor that she knows a generator for $< S >$, without revealing it.

Secret parameters	
A primitive 2^n-torsion point S defining an isogeny $a : E \to E/ < S >$.	
Public parameter creation	
The supersingular elliptic curves E and $E/ < S >$ over $\mathbb{F}_p^2$, generators P, Q of $E[3^m]$ and their images $a(P), a(Q)$.	
Identification: repeat k times	
Peggy	**Victor**
Chooses a random primitive 3^m-torsion point R Computes $\psi : E \to E_1 = E/ < R >$. Computes $\psi' : E/ < S > \to E_2 = E/ < S, R >$. Computes $a' : E/ < R > \to E_2$.	
Public exchange of values	
Sends E_1 and E_2 to Victor.	Sends a random bit b to Peggy.
If $b = 0$, sends $R, a(R)$ to Victor.	If $R, a(R)$ have order 3^m he accepts them and generates the kernels of isogenies $E \to E_1$ and $E/ < S > \to E_2$.
If $b = 1$, sends $\psi(S)$ to Victor.	If $\psi(S)$ has order 2^n he accepts it and generates the kernel of an isogeny $E_1 \to E_2$.

Table 5: A zero-knowledge proof of identity.

In section 3.8 we present some difficult problems on which the security of SIDH is based. Problems (2) and (5) ensure that this is indeed a zero-knowledge proof.

3.6 Implementation

We include here some implementations made on Pari GP.

```
formulaVelu7(E,P,Q)={
        if(! ellisoncurve(E,P), error("P is not on the curve"));
        if(! ellisoncurve(E,Q), error("Q is not on the curve"));

        my(n,i,X,Y,iP,QiP);
        n=ellorder(E,P);
```

```
        X=Q[1];
        Y=Q[2];
        for(i=1,n-1,
                iP=ellmul(E,P,i);
                        QiP=elladd(E,Q,iP);
                if(QiP==[0],return([0]));
                X=X+QiP[1]-iP[1];
                Y=Y+QiP[2]-iP[2];
            );
        return([X,Y]);
    }

generatorToEquation(E,P)={
        if(! ellisoncurve(E,P), error("P is not on the curve"));
    my(n,r,i,iP);
    n=ellorder(E,P);
        r=1;
    for(i=1,n-1,
        iP=ellmul(E,P,i);
        r=(x-iP[1])*r;
        );
        return(r);
    }
```

This is an implementation of the improvement of Velu's formula by Luca De Feo:

```
formulaDeFeo(E,h)={
        my(f,deg,p1,r);
        f=x^3+E.a4*x+E.a6;
    deg=poldegree(h)+1;
    p1=-polcoeff(h,poldegree(h)-1);
    r=deg*x-p1-f'*h'/h-2*f*(h'/h)';
    return([r, y*r']);}

evaluate(F,Q)={
        return(substvec(F,[x,y],Q));}
```

We now proceed to an example:

```
E=ellinit([3,4],7)
P=Mod([2,5],7)
PP=elladd(E,P,P)
h=generatorToEquation(E,PP)
F=formulaDeFeo(E,h)
evaluate(F,P)      /*[Mod(4, 7), Mod(0, 7)]*/
formulaVelu7(E,PP,P);
```

3.7 Complexity and security

The security of SIDH is based on some problems that are assumed to be hard and to some of which we have referred earlier. We include them all here.

We are in the same setting as for the SIDH, meaning that p is a prime of the form $p = {l_A}^{e_A}{l_B}^{e_B}f \pm 1$, E is a fixed supersingular curve over $\mathbb{F}_{p^2}$ and $\{P_A, Q_A\}, \{P_B, Q_B\}$ are basis of the torsion subgroups $E[{l_A}^{e_A}], E[{l_B}^{e_B}]$ respectively.

(1) *Decisional Supersingular Isogeny problem.* Let E and E_a be two supersingular curves over $\mathbb{F}_{p^2}$. Decide whether E_a is ${l_A}^{e_A}$-isogenous to E.

(2) *Computational Supersingular Isogeny problem.* Let $a : E \to E_a$ be an isogeny whose kernel is equal to $< [n_A]P_A + [m_A]Q_A >$, where n_A, m_A are random elements of $\mathbb{Z}/{l_A}^{e_A}\mathbb{Z}$ not both divisible by l_A. Given E_a and the values $a(P_B), a(Q_B)$, find a generator of the kernel $< [n_A]P_A + [m_A]Q_A >$.

If the last problem could be easily solved, this would be equivalent to finding the isogenies a, b of Alice and Bob, using Velu's formula, and their common secret key: E_{ab}.

(3) *Supersingular Computational Diffie-Hellman problem.* Let $a : E \to E_a$ be an isogeny whose kernel is equal to $< [n_A]P_A + [m_A]Q_A >$, and let $b : E \to E_b$ be an isogeny whose kernel is equal to $< [n_B]P_B + [m_B]Q_B >$, where n_A, m_A (respectively n_B, m_B) are random elements of $\mathbb{Z}/{l_A}^{e_A}\mathbb{Z}$ (respectively $\mathbb{Z}/{l_B}^{e_B}\mathbb{Z}$) not both divisible by l_A (respectively l_B). Given the curves E_A, E_B and the points $a(P_B), a(Q_B), b(P_A), b(Q_A)$, find the j-invariant of $E/ < [n_A]P_A + [m_A]Q_A, [n_B]P_B + [m_B]Q_B >$.

(4) *Supersingular Decision Diffie-Hellman problem.* Given a tuple sampled with probability $1/2$ from one of the following two distributions:

- $(E_A, E_B, a(P_B), a(Q_B), b(P_A), b(Q_A), E_{ab})$ where $E_{ab} \simeq E/ < [n_A]P_A + [m_A]Q_A, [n_B]P_B + [m_B]Q_B >$,
- $(E_A, E_B, a(P_B), a(Q_B), b(P_A), b(Q_A), E_C)$ where $E_C \simeq E/ < [n'_A]P_A + [m'_A]Q_A, [n'_B]P_B + [m'_B]Q_B >$, where n'_A, m'_A (respectively n'_B, m'_B) are randomly chosen from $\mathbb{Z}/{l_A}^{e_A}\mathbb{Z}$ (respectively $\mathbb{Z}/{l_B}^{e_B}\mathbb{Z}$) not both divisible by l_A (respectively l_B),

determine from which distribution the tuple is sampled.

(5) *Decisional Supersingular Product problem.* Given an isogeny $f : E \to E_3$ of degree ${l_A}^{e_A}$ and a tuple sampled with probability $1/2$ from one of the following two distributions:

- (E_1, E_2, f'), where the product $E_1 \times E_2$ is chosen at random among those ${l_B}^{e_B}$-isogenous to $E \times E_3$, and where $f' : E_1 \to E_2$ is an isogeny of degree ${l_A}^{e_A}$, and
- (E_1, E_2, f'), where the product E_1 is chosen at random among the curves having the same cardinality as E, and where $f' : E_1 \to E_2$ is an isogeny of degree ${l_A}^{e_A}$,

determine from which distribution the tuple is sampled.

4 Attacks

4.1 The endomorphism ring of an elliptic curve

In order to understand the attack discussed below, some knowledge of the endomorphism rings of elliptic curves is necessary, so we include here some basic information about them.

Let E be an elliptic curve over a finite field $\mathbb{F}_q$. The endomorphism ring of E is $End(E) := Hom(E, E)$. It consists of all homomorphisms from E to itself, which in fact are all the isogenies from E to itself and has the structure of a ring, as explained in remark 2.4.2.

Proposition 4.1.1. *1. $End(E)$ is a torsion-free $\mathbb{Z}-$module.*

2. $End(E)$ is a domain of characteristic 0.

3. $End(E)$ has rank at most 4.

Proof. 1. Let $f \in End(E)$ and $m \in \mathbb{Z}$ such that $[m] \circ f = [0]$. This implies that $(deg[m])(degf) = 0$. So either $deg[m] = 0$ or $degf = 0$. If $deg[m] = 0$, since the multiplication-by-m map is not a constant map, the first condition implies that $[m] = 0$. If $m \neq 0$, then $deg[m] \geq 1$ and so $f = 0$.

2. The claim that $End(E)$ has characteristic 0 follows directly from 1. Let $f, g \in End(E)$ such that $f \circ g = 0$. Then $(degf)(degg) = deg(f \circ g) = 0$. Thus, either $f = [0]$ or $g = [0]$ and so $End(E)$ is a domain.

3. To prove this one should define the Tate module. We are not going to include this proof here. We refer the interested reader to [Sil09] III.7.5.

□

Remark 4.1.2. For an elliptic curve E defined over a field k with characteristic 0, the map

$$[\ ] : \mathbb{Z} \to End(E)$$

is usually an isomorphism. However, if k is a finite field, then $End(E)$ is always larger than $\mathbb{Z}$.

Definition 4.1.3. We say that an elliptic curve E has *complex multiplication* if $End(E)$ is strictly larger than $\mathbb{Z}$.

Definition 4.1.4. A (definite) *quaternion algebra* (over $\mathbb{Q}$) is an algebra of the form

$$K = \mathbb{Q} + \mathbb{Q}\alpha + \mathbb{Q}\beta + \mathbb{Q}\alpha\beta$$

with α, β satisfying $\alpha^2, \beta^2 \in \mathbb{Q}, \alpha^2 < 0, \beta^2 < 0, \beta\alpha = -\alpha\beta$.

Definition 4.1.5. Let K be a $\mathbb{Q}$-algebra that is finitely generated over $\mathbb{Q}$. An *order* R of K is a subring of K that is finitely generated as a $\mathbb{Z}$-module and satisfies $R \otimes \mathbb{Q} = K$.

Example 4.1.6. *Let K be an imaginary quadratic field and let O be its ring of integers. Then for each integer $f \geq 1$, the ring $\mathbb{Z} + fO$ is an order of K, and more precisely, these are all the orders of K.*

Using the proposition 4.1.1, we can see that the endomorphism ring of an elliptic curve may take one of the following forms:

Theorem 4.1.7. *The endomorphism ring R of an elliptic curve E over a field K is either $\mathbb{Z}$, an order in an imaginary quadratic field or an order in a quaternion algebra. If $char(K) = 0$, only the two first cases occur.*

Proof. Let $K = R \times \mathbb{Q}$. Since R is finitely generated as a $\mathbb{Z}$-module, it suffices to prove that K is either $\mathbb{Q}$, an imaginary quadratic field or a quaternion algebra. We extend the anti-involution to K and define a reduced norm and trace from K to $\mathbb{Q}$ by

$$N\alpha = \alpha\hat{\alpha}, T\alpha = \alpha + \alpha$$

We observe that:

- $T\alpha = 1 + N\alpha - N(\alpha - 1)$. Thus $T\alpha \in \mathbb{Q}$.
- The trace is $\mathbb{Q}$-linear, since the involution fixes $\mathbb{Q}$.
- If $\alpha \in \mathbb{Q}$, then$T\alpha = 2\alpha$.
- If $\alpha \in K$ satisfies $T\alpha = 0$, then $0 = (\alpha - \alpha)(\alpha - \hat{\alpha}) = \alpha^2 - (T\alpha)\alpha + N\alpha = \alpha^2 + N\alpha$. So $\alpha^2 = -N\alpha$. Thus $a \neq 0$ and $T\alpha = 0$ and so $\alpha^2 \in \mathbb{Q}$ and $\alpha^2 < 0$.

If $K = \mathbb{Q}$, there is nothing to prove. Otherwise we can find some $\alpha \in K$ with $a \notin \mathbb{Q}$. Replacing α by $\alpha - \frac{1}{2}T\alpha$, we may assume that $T\alpha = 0$. Then $\alpha^2 \in \mathbb{Q}$ and $\alpha^2 < 0$, so $\mathbb{Q}(a)$ is a quadratic imaginary field. If $K = \mathbb{Q}(\alpha)$ we are again done. Suppose now that $K \neq \mathbb{Q}(\alpha)$ and choose some $\beta \in K$ with $\beta \notin \mathbb{Q}(\alpha)$. We may replace β with

$$\beta - \frac{1}{2}T\beta - \frac{T(\alpha\beta)}{2\alpha^2}\alpha.$$

We know that $T\alpha = 0$ and $\alpha^2 \in \mathbb{Q}^*$, so an easy calculation shows that

$$T\beta = T(\alpha\beta) = 0.$$

In particular, $\beta^2 \in \mathbb{Q}$ and $\beta^2 < 0$. Next we write

$$T\alpha = 0, T\beta = 0, T(\alpha\beta) = 0$$

as

$$\alpha = -\hat{\alpha}, \beta = -\hat{\beta}, \alpha\beta = -\hat{\beta}\hat{\alpha}$$

and substitute the first two equalities into the third to obtain

$$\alpha\beta = -\alpha\beta.$$

Hence

$$\mathbb{Q}[\alpha, \beta] = \mathbb{Q} + \mathbb{Q}\alpha + \mathbb{Q}\beta + \mathbb{Q}\alpha\beta$$

is a quaternion algebra. It remains to prove that $\mathbb{Q}[\alpha, \beta] = K$, and to do this, it suffices to show that $1, \alpha, \beta, \alpha\beta$ are $\mathbb{Q}$-linearly independent, since then $\mathbb{Q}[\alpha, \beta]$ and K both have dimension 4 over $\mathbb{Q}$.
Suppose that

$$w + x\alpha + y\beta + z\alpha\beta = 0$$

with $w, x, y, z \in \mathbb{Q}$. Taking the trace yields $2w = 0$, so $w = 0$. Next we multiply by α on the left and by β on the right to obtain

$$(x\alpha^2)\beta + (y\beta^2)\alpha + z\alpha^2\beta^2 = 0.$$

We know that $1, \alpha$ and β are $\mathbb{Q}$-linearly independent, since $\alpha \notin \mathbb{Q}$ and $\beta \notin \mathbb{Q}(\alpha)$.Hence this equation implies that

$$x\alpha^2 = y\beta^2 = z\alpha^2\beta^2 = 0,$$

and so $x = y = z = 0$, which completes the proof that $1, \alpha, \beta$ and $\alpha\beta$ are $\mathbb{Q}$-linearly independent. □

Elliptic curves are either ordinary or supersingular, depending on the structure of their endomorphism ring. More concretely,

Definition 4.1.8. An elliptic curve E is called *supersingular* if $End(E)$ is an order in a quaternion algebra. Otherwise, we say that E is ordinary.

An equivalent definition is that E over $\mathbb{F}_q$ is

- supersingular, if $|E[p]| = 0$, which implies that $|E[p^n]| = 0$, for every n,
- ordinary, if $|E[p^n]| = p^n$, for every n.

Two isogenous curves are either both supersingular or both ordinary.

Most elliptic curves are ordinary. In particular, supersingular curves have density zero.

If an elliptic curve over the rational numbers has complex multiplication, then the set of primes for which it is supersingular has density $1/2$. If it does not have complex multiplication then Serre showed that the set of primes for which it is supersingular has density zero. Elkies (1987) showed that any elliptic curve defined over the rationals is supersingular for an infinite number of primes. We call a prime number p, for which the reduction over $\mathbb{F}_p$ of an elliptic curve $E/\mathbb{Q}$ is supersingular, a *supersingular prime*.

Theorem 4.1.9. *Let E/Q be an elliptic curve without complex multiplication. Then the set of supersingular primes has density 0.*

Theorem 4.1.10. *Let $E/\mathbb{Q}$ be an elliptic curve without complex multiplication. Then there are infinitely many primes p for which $E/\mathbb{F}_p$ is supersingular.*

The endomorphism rings of supersingular elliptic curves have the largest possible rank: 4.

4.2 Why supersingular instead of ordinary curves

As we have already mentioned, the first proposed algorithm based on isogenies was using ordinary elliptic curves. However, after exploring possible attacks, it appears that there are attacks against this protocol that are subexponential using quantum computers. A subexponential attack against it was found by A. Childs, D. Jao and V. Soukharev[CJS14]. In this section we will see why this happens with ordinary curves and not with supersingular.

At their article, Childs, Jao and Soukharev, expose a subexponential-time quantum algorithm for constructing isogenies, assuming only the Generalized Riemann Hypothesis. Their algorithm is based on a reduction to the *abelian hidden shift problem*.

Definition 4.2.1. *(Hidden shift problem)* Let X be a set and G a group that acts on X. Let $g \in G$ be a secret element of G. A *hidden shift problem* is the following: If $f_1, f_2 : X \to X$ are functions such that, for every $x \in X$, $f_2(x) = f_1(g \star x)$, find g. (Note: With $\star$ with symbolize the group action.)

A special case of a hidden shift problem, is the hidden subgroup problem. The hidden subgroup problem is especially important in the theory of quantum computing, since, among others, Shor's quantum algorithm for factoring and discrete logarithm is essentially equivalent to the hidden subgroup problem for finite abelian groups. And indeed, there is a polynomial time quantum algorithm for solving the hidden subgroup problem over finite abelian groups.

But let's see how the ordinary isogeny Diffie-Hellman algorithm, proposed by Rostovtsev and Stolbunov in [RS06], works:

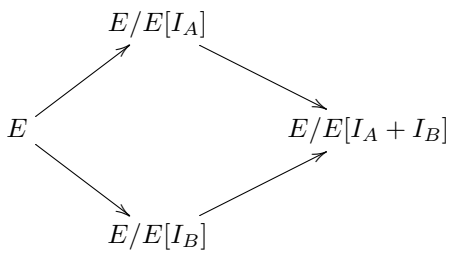
Alice and Bob begin with a public elliptic curve E . Each of them computes an ideal I_A, I_B of $End(E)$, respectively. Then, Alice computes the isogeny from E with kernel $E[I_A]$, called E_a, and Bob the one with kernel $E[I_B]$, called E_b. Note that for an ideal I of a curve E we symbolize with $E[I]$ the finite subgroup of E

$$E[I] = \{P \in E(\bar{\mathbb{F}}_q) : f(P) = 0 \text{ for all } f \in I\}.$$

Then they exchange the curves E_a and E_b. Next, Alice computes $E_a/E_a[I_B]$ and Bob $E_b/E_b[I_A]$, which are equal, since

$$E_a/E_a[I_B] = E/E[I_A + I_B] = E_b/E_b[I_A].$$

The above procedure is described in the next diagram:



We notice that in the ordinary case Alice and Bob don't need to exchange any information about their isogenies, they just exchange their isogenous curves.

This algorithm works with ordinary curves and not supersingular ones, because the endomorphism ring of ordinary curves is commutative, while for supersingular it is not. And without the commutativity of the endomorphism ring, Alice and Bob cannot conclude to the same elliptic curve which they use as their common key. And that's because I_A is a left ideal of E and a right ideal of E_a (respectively, I_B is a left ideal of E and a right ideal of E_b) and although left and right ideals coincide in commutative rings, this isn't the case in non-commutative ones.

We consider the problem of constructing an isogeny between two given isogenous ordinary elliptic curves defined over a finite field F_q and having the same endomorphism ring. We recall from the previous section that the endomorphism ring of an ordinary elliptic curve is either $\mathbb{Z}$ or an order in an imaginary quadratic field. In any case it is commutative. The fastest known probabilistic algorithm for solving this problem is the algorithm of Galbraith and Stolbunov. Their algorithm is exponential.

However, an attack to the problem of isogeny construction is a reduction to the abelian hidden shift problem. While a connection between isogenies and hidden subgroups was noted previously by Stolbunov, in their article, Childs, Jao and Soukharev observe that the reduction gives an injective hidden shift problem. This allows them to apply an algorithm of Kuperberg to solve the hidden shift problem using a subexponential number of queries to certain functions.

Over a finite field $\mathbb{F}_q$, two elliptic curves E and E' are isogenous if and only if $|E(\mathbb{F}_q)| = |E'(\mathbb{F}_q)|$, as stated in theorem 2.4.4. The endomorphism ring of an ordinary elliptic curve over a finite field is an imaginary quadratic order O_Δ of discriminant $\Delta < 0$. The set of all isomorphism classes (over $\bar{F}_q$) of isogenous curves with endomorphism ring O_Δ is denoted $Ell_{q,n}(O_\Delta)$, where n is the cardinality of any such curve. We represent elements of $Ell_{q,n}(O\Delta)$ by taking the j-invariant of any representative curve in the isomorphism class. Any separable isogeny $\phi : E \to E'$ between curves in $Ell_{q,n}(O_\Delta)$ with the same endomorphism rings can be specified, up to isomorphism, by giving E and $ker\phi$, as seen in theorem 2.4.16. The kernel of an isogeny, in turn, can be represented as an ideal in O_Δ. Denote by $\phi_b : E \to E_b$ the isogeny corresponding to an ideal b (keeping in mind that ϕ_b is only defined up to isomorphism of E_b). Principal ideals correspond to isomorphisms, so any other ideal equivalent to b in the ideal class group $Cl(O_\Delta)$ of $O_\Delta$induces the same isogeny, up to isomorphism. Hence one obtains a well-defined group action

$$* : Cl(O_\Delta) \times Ell_{q,n}(O_\Delta) \to Ell_{q,n}(O_\Delta)$$

$$[b] * j(E) = j(E_b),$$

where $[b]$ denotes the ideal class of b. This group action, which we call the *isogeny star operator*, is free and transitive, and thus $Ell_{q,n}(O_\Delta)$ forms a principal homogeneous space over $Cl(O_\Delta)$.

The following appears as Theorem 7.1 of [Kup05]:

Theorem 4.2.2. *The abelian hidden shift problem has a [quantum] algorithm with time and query complexity $2^{O(\sqrt{n})}$, where n is the length of the output, uniformly for all finitely generated abelian groups.*

We now return to the original problem of constructing isogenies. Note that to use the hidden shift approach, the group structure of $Cl(O_\Delta)$ must be known. Given Δ, it is straightforward to compute $Cl(O_\Delta)$ using existing quantum algorithms. Thus, we assume for simplicity that the discriminant Δ is given as part of the input. This requirement poses no difficulty, since O_Δ is a maximal order, in which case its discriminant can be computed easily: simply calculate the trace $t(E)$ of the curve using Schoof's algorithm, and factor $t(E)^2 - 4q$ to obtain the fundamental discriminant Δ. Factoring is easy on a quantum computer since it can be done in polynomial time using Shor's algorithm.

Assuming Δ is known, we decompose $Cl(O_\Delta)$ as a direct sum of cyclic groups, with a known generator for each, and then solve the hidden shift problem. The overall procedure is described in Algorithm 3 of [CJS14].

4.3 How useful to an attacker is the public information about the isogenies of Alice and Bob?

In SIDH Alice and Bob give some information about their secret isogenies, in contrast to the ordinary isogeny protocol, in order to overcome the non-commutativity of the endomorphism ring. This happens when they publish the image of the basis of $E[{l_B}^{e_B}]$ and $E[{l_A}^{e_A}]$ under their isogenies, respectively. An attack on Jao-De Feo-Plut protocol is the following: compute an isogeny $a : E \to E_a$ of degree ${l_A}^{e_A}$ given an action of a on $E[{l_B}^{e_B}]$. But does this attack represent an important threat?

If $gcd({l_A}^{e_A}, {l_B}^{e_B}) \neq 1$, one can recover part of a, since if a is a group homomorphism. However, we always pick l_A, l_B such that $gcd({l_A}^{e_A}, {l_B}^{e_B}) = 1$, since l_A, l_B are different primes, and so we overcome this danger. As previously, we choose $l_A = 2$ and $l_B = 3$ and continue to explore possible attacks.

An attacks was published by Galbraith, Petit, Shani and Ti[GPST16]. This attack can be prevented, but this adds significant cost to the running time of the system.

5 Conclusion

The SIDH is considered by now a good applicant for post-quantum cryptography. Except for the key-exchange, a zero-knowledge proof of identity has also been developed based on computing isogenies between supersingular elliptic curves. An open problem is to find a digital signature based on it, that will be quantum resistant.

References

[BJS14] Jean-François Biasse, David Jao, and Anirudh Sankar. A quantum algorithm for computing isogenies between supersingular elliptic curves. In *INDOCRYPT*, volume 8885, pages 428–442, 2014.

[CJS14] Andrew Childs, David Jao, and Vladimir Soukharev. Constructing elliptic curve isogenies in quantum subexponential time. *Journal of Mathematical Cryptology*, 8(1):1–29, 2014.

[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.

[GPST16] Steven D Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. 2016.

[HPS14] Jeffrey Hoffstein, Jill Pipher, and Joseph H Silverman. *An introduction to mathematical cryptography*. springer, 2014.

[JDF11] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *PQCrypto*, 7071:19–34, 2011.

[Kup05] Greg Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM Journal on Computing*, 35(1):170–188, 2005.

[RS06] Alexander Rostovtsev and Anton Stolbunov. Public-key cryptosystem based on isogenies. *IACR Cryptology ePrint Archive*, 2006:145, 2006.

[Sil09] Joseph H Silverman. *The arithmetic of elliptic curves*, volume 106. Springer Science & Business Media, 2009.

[ST15] Joseph H Silverman and John T Tate. *Rational points on elliptic curves*. Springer, 2015.

[Vél71] Jacques Vélu. Isogénies entre courbes elliptiques. *CR Acad. Sci. Paris Sér. AB*, 273(A238-A241):5, 1971.

[DFJP14] [BJS14] [JDF11] [Sil09] [Vél71] [GPST16] [CJS14] [HPS14] [ST15] [RS06] [Kup05]