QUANTUM ALGORITHM FOR SUPERSINGULAR ISOGENY PROBLEM

Zhengyi Han
School of Electronic Engineering and Computing Science
Peking University
1900013024@pku.edu.cn

May 23, 2022

ABSTRACT

We make an overview of supersingular isogeny cryptography, and exhibit our attempts to make an improvement on existing algorithms along with the possible reasons why we finally failed. This paper also gives a brief tutorial of elliptic curve isogenies and some computational problems. Supersingular isogeny cryptography is attracting attention due to it's quantum-resistant. However, many computational problems involving the structure of isogeny graph have not been sufficiently studied. We summarized some previous results and describe our understanding of this problem. We hope these can help others.

1 Introduction

In 1994, Shor proposed a quantum algorithm which can factor a composite in polynomial time. [1] It means that the RSA protocol [2] is not safe under quantum attack. John and Christof designed a variant Shor's algorithm [3] which can be applied to elliptic curve cryptography. Diffie-Hellman protocol [4] is also unsafe in post-quantum era. In 2011, Luca De Feo, David Jao and Jérôme Plût [5] presented new candidates for quantum-resistant public-key cryptosystems based on the conjectured difficulty of finding isogenies between supersingular elliptic curves. In 2016, National Institute of Standards and Technology (NIST) has begun the call for post-quantum cryptographic algorithms from all over the world. SIKE [6] is the only supersingular isogeny protocol in NIST.

There are many extraordinary textbooks and lecture notes about elliptic curves, such as [7–10]. For quantum algorithm for algebraic problem, [11] is a good survey. Basic knowledge about quantum computing can be referred to [12].

In chapter 2, we make a brief tutorial to elliptic curves. Chapter 3 is about the quantum subexponential time algorithm for constructing isogeny between ordinary curves. In chapter 4,5, we introduce general protocol and two different quantum attacks on it. We also exhibit our attempts on improving the algorithm for isogeny problem, and analyse the essential hardness of this problem along with the reasons why they failed in chapter 6.

2 Mathematical background

2.1 Introduction to elliptic curves

In mathematics, an elliptic curve is a smooth, projective, algebraic curve of genus one, on which there is a specified point O. An elliptic curve is defined over a field k and describes points in$k \times k$.If the field's character$char(k) \neq 2, 3$, then the curve can be described as a plane algebraic curve :

$$E : y^2 = x^3 + ax + b, a, b \in k$$

where $\Delta := 4a^3 + 27b^2 \neq 0$, which means the curve is non-singular, i.e. the curve has no cusps or self-intersections. The set of points of an elliptic curve can be equipped with an additive group law.Actually there is an isomorphism:

$$\begin{aligned}\Phi : \mathbb{C}/L &\to E(\mathbb{C}) \\ z &\mapsto (\wp(z), \wp'(z))\end{aligned}$$

where L is a $\mathbb{C}$-lattice, $\wp(z)$ is Weierstrass elliptic function. So elliptic curves preserves the additive structure. Details about the arithmetic of elliptic curves can be found in many references, such as [7, 10].

Let $\mathbb{F}_q$ be a finite field, where $q = p^n, n \in \mathbb{N}, p > 3$. For simplicity, we assume that $p > 3$ in the following. In projective space, the elliptic curve E over $\mathbb{F}_q$ is the set of points:

$$E(\mathbb{F}_q) = \{(x, y) \in {\mathbb{F}_q}^2 : y^2 = x^3 + ax + b\} \cup \{\mathbf{0}_E\}$$

where $\mathbf{0}_E$ is the point $(x : y : z) = (0 : 1 : 0)$ on the projective curve$y^2z = x^3 + axz^2 + bz^3$, we denote$\mathbf{0}_E$ as $\mathbf{0}$ for simplicity. Sometimes we also consider the set $E(\bar{\mathbb{F}}_q)$ of all the points over the algebraic closure $\bar{\mathbb{F}}_q$ instead of $\mathbb{F}_q$.

There are "nearly q" points on an elliptic curve over $\mathbb{F}_q$. Precisely, $\#E(\mathbb{F}_q) = q - 1 + t$, where t is an integer, and $|t| \leq 2\sqrt{q}$(Hasse Theorem). We call an elliptic curve over $\mathbb{F}_q$, where $q = p^a$, supersingular if $p|t$, and ordinary otherwise. It follows that $\#E(\mathbb{F}_q) \equiv 1 \pmod p$. Maybe it's confusing that why we separate elliptic curves by this, we will later see the motivation.

For $n \in \mathbb{N}, P \in E(\mathbb{F}_q)$, we define $[n]P = P + \cdots + P$,(n times). Define $E[n] = P \in E(\bar{\mathbb{F}}_q)$. If $(p \nmid n)$, it follows that $\#E[n] = n^2$, and actually is a direct product of two cyclic group of order n. If E is supersingular then $E[p] = \{\mathbf{0}\}$, and $\#E[p] = p$ when ordinary.

A morphism between elliptic curves is a group homomorphism and a rational map over the algebraic closure. More generally, morphisms can be defined for abelian varieties. An isomorphism of elliptic curves $f : E \to E'$ is a morphism that satisfies $f(\mathbf{0}_E) = \mathbf{0}_{E'}$, and whose inverse (over the algebraic closure). So an isomorphism is a bijection $E(\bar{\mathbb{F}}_q) \to E(\bar{\mathbb{F}}_{q'})$.

Note: Isomorphisms are over $\bar{\mathbb{F}}_q$, so they are not necessary maps between $E(\mathbb{F}_q)$ and $E'(\mathbb{F}_q)$. Actually if $\#E(\mathbb{F}_q) = q + 1 - t$, there is another elliptic curve E' over $\mathbb{F}_q$ called the **quadrtic twist** of $E(\mathbb{F}_q)$, with $\#E'(\mathbb{F}_q) = q + 1 + t$ and isomorphic to E, because isomorphism is not defined over $\mathbb{F}_q$, but over its quadrtic extension $\mathbb{F}_{q^2}$.

We can separate all elliptic curves over an algebraic closed field$k$(we take $k = \bar{\mathbb{F}}_q$ here) into their isomorphism classes by **j-invariant**.
The j-invariant of an elliptic curve $E : y^2 = x^3 + ax + b$ is:

$$j(E) = 1728 \frac{4a^3}{4a^3 + 27b^2}$$

E and E' is isomorphic if and only if $j(E) = j(E')$.
Inversely, given $j \in \bar{\mathbb{F}}_q$ with $j \neq 0, 1728$, we can compute the corresponding (unique) elliptic curves easily:

$$E : y^2 = x^3 + \frac{3j}{1728 - j}x + \frac{2j}{1728 - j}$$

and $j(E) = j$. But in field which is not algebraic closed, the property doesn't hold, $i.e.$, there two elliptic curves having the same j-invariant but not isomorphic.

2.2 Isogeny

An **isogeny** is a morphism $\phi : E \to E'$, with $\phi(\mathbf{0}_E) = \mathbf{0}_{E'}$. We call two elliptic curves are isogenous if there is a non-constant isogeny between them. Actually isogenies have a standard form:

$$\phi(x, y) = (\frac{u(x)}{v(x)}, \frac{s(x)}{t(x)}y)$$

The **degree** d of an isogeny is $d = \max\{\deg(u(x)), \deg(v(x))\}$. We call an isogeny is **separable** if the formal derivative of $\frac{u(x)}{v(x)}$ is zero, and **inseparable** otherwise. The degree of an isogeny corresponds with the number of points in the kernel in separable case.

The **dual isogeny** to $\phi: E \to E'$ is an isogeny $\hat{\phi}: E' \to E,\ s.t.\ \hat{\phi} \circ \phi = [\deg(\phi)]: E \to E$. The dual isogeny exists for every isogeny ϕ. And Tate's Theorem says that any two elliptic curves E, E' defined over $\mathbb{F}_q$ are isogenous over $\mathbb{F}_q$ if and only if $\#E(\mathbb{F}_q) = \#E'(\mathbb{F}_q)$ [7, V Ex 5.4]. Maybe it's confusing that an isogeny has a kernel, is it contradictive to that two isogenous elliptic curves have the same number of points? It is due to the kernel of isogeny is a subgroup of $E(\bar{\mathbb{F}}_q)$ rather than $E(\mathbb{F}_q)$.

Every isogeny $\phi: E \to E'$ has a kernel $ker(\phi) \leq E(\bar{\mathbb{F}}_q)$, can we determine an isogeny uniquely only by its kernel? Here a theorem discussing this problem:

Theorem 1 *Let E be an elliptic curve defined over $\mathbb{F}_q$ and G a finite subgroup of $E(\bar{\mathbb{F}}_q)$ that is defined over $\mathbb{F}_q$. Then there is an elliptic curve E' defined over $\mathbb{F}_q$, and a separate isogeny $\phi: E \to E'$ defined over $\mathbb{F}_q$ of degree $\#G$, with $G = ker(\phi)$. If there is another separable isogeny $\psi: E \to E''$ of degree $\#G$, with $ker(\psi) = G$, then $j(E') = j(E'')$, $i, e,$ E' is isomorphic to E''. Hence, the image of $\phi(E)$ is well defined, and we can denote it by E/G*

There is an explicit algorithmic proof named Velu's formulas, whose complexity is $O(n)$ field operations to compute the isogeny ϕ with $\deg(\phi) = n$.

Another important property of isogeny is that isogenys can be factored. For example, $\phi: E \to E'$ is a separable isogeny defined over $\mathbb{F}_q$. If $\phi = \phi_1 \circ \cdots \circ \phi_k \circ [n]$, then $\deg(\phi) = n^2 \cdot \prod_{i=1}^{k} \deg(\phi_i)$. If we take ϕ_i all isogenies with a small prime degree, for example 2. Then the complexity to compute the isogeny ϕ is $O(k)$, instead of $O(2^k)$ by Velu's formulas.

Because of these properties, isogeny has a broad amplication in cryptography, both in encryption and attacking. Many works have been done to speed up the computation of isogenies. Later we will discuss it in more detail. In addition, isogeny-based cryptography usually works on isomorphism classes of elliptic curves. On reason is that Velu's formulas' outputs are the isogeny ϕ and the iamge E'. E' is isomorphic to the desired elliptic curve, but not necessarily the desired curve.

2.3 Endomorphism and complex multiplication

Because any elliptic curve over $\mathbb{C}$ is isomorphic to a complex tori $\mathbb{C}/L$. Actually, elliptic curves over $\mathbb{C}$ and complex torus are categorial equivalence. It follows that the endomorphism ring of elliptic curves is isomorphic to $\mathcal{O}(L) := \{\alpha \in \mathbb{C} : \alpha L \subseteq L\}$. This is the origin of term **complex multiplication**.

The endomorphism algebra of elliptic curve E is $End^0(E) := End(E) \otimes_{\mathbb{Z}} \mathbb{Q}$. Here is the classfication theorem for endomorphism algebras.

Theorem 2 *Let E/k be an elliptic curve. Then $End^0(E)$ is isomorphic to one of:*

- *the field of rational numbers $\mathbb{Q}$*
- *an imaginary quadratic field $\mathbb{Q}(\alpha)$, with $\alpha^2 < 0$*
- *a quaternion algebra $\mathbb{Q}(\alpha, \beta)$, with $\alpha^2, \beta^2 < 0$.*

According to the above thoerem, the endomorphism ring of elliptic curve over $\mathbb{C}$ is either $\mathbb{Q}$ or an imaginary quadratic field $\mathbb{Q}(\alpha)$.

As for elliptic curves over a finite field, there is a theorem about the endomorphism algebra.

Theorem 3 *Let E be an elliptic curve over a finite field of characteristic p, π_E is Frobenius endomorphism. Either E is supersingular, $tr_E \equiv 0 \bmod p$, and $End^0(E_{\bar{\mathbb{F}}_q})$ is a quaternion algebra, or E is ordinary, $tr_E \neq 0$, and $End^0(E_{\bar{\mathbb{F}}_q})$ is an imaginary quadratic field.*

It follows that the endomorphism algebra of elliptic curve over finite field is never equals $\mathbb{Q}$, which is different from the cases over $\mathbb{C}$. And endomorphism of supersingular curves are not commutative.

Furthermore, to show the relationship between the endomorphism ring and isomorphism class, we take elliptic curve over $\mathbb{C}$ as an example, whose endomorphism ring is an order of an imaginary quadratic field.

Due to the relationship between lattices and elliptic curves given by Weierstrass $\wp$ function. Given an order $\mathcal{O}$, there is a bijection:

$$\{L \subseteq \mathbb{C} : \mathcal{O}(L) = \mathcal{O}\}/_{\sim} \longleftrightarrow \{E/\mathbb{C} : \mathrm{End}(E) = \mathcal{O}\}/_{\simeq}$$

Here $\sim$ means the homothety between lattices, and $\simeq$ means the isomorphism betweem elliptic curves. Moreover, the above two sets are both in bijection with the ideal class group $\mathrm{cl}(\mathcal{O})$. We define an $\mathcal{O}$-ideal L is proper if $\mathcal{O}(L) = \mathcal{O}$, the ideal class group can be defined as:

$$\mathrm{cl}(\mathcal{O}) := \{\text{proper } \mathcal{O}\text{-ideals } \mathfrak{a}\}/_{\sim}$$

$$\mathfrak{a} \sim \mathfrak{b} \Longleftrightarrow \alpha\mathfrak{a} = \beta\mathfrak{b} \text{ for nonzero } \alpha, \beta \in \mathcal{O}.$$

Note that the equivalence between $\mathcal{O}$-ideals corresponds to homothety between lattices. We can label the isomorphism classes $\mathrm{Ell}_{\mathcal{O}}(\mathbb{C}) := \{j(E) : E/\mathbb{C} \text{ with } \mathrm{End}(E) = \mathcal{O}\}$. by ideal classes:

$$\begin{aligned} \mathrm{cl}(\mathcal{O}) &\longrightarrow \mathrm{Ell}_{\mathcal{O}}(\mathbb{C}) \\ [\mathfrak{a}] &\longmapsto j(E_{\mathfrak{a}}) = j(\mathfrak{a}) \end{aligned}$$

Moreover, $\mathrm{cl}(\mathcal{O})$ has a group action on $\mathrm{Ell}_{\mathcal{O}}(\mathbb{C})$ via: $[\mathfrak{a}]j(E_{\mathfrak{b}}) = j(E_{\mathfrak{a}^{-1}\mathfrak{b}})$. Actually the action of $\mathrm{cl}(\mathcal{O})$ is not only faithful (only the identity fixex every element), and it's free (every stablizer is trivial). In addition, $\#\mathrm{cl}(\mathcal{O}) = \#\mathrm{Ell}_{\mathcal{O}}(\mathbb{C})$ together with the free action imply that the action is transitive. A group action that is free and transitive is said to be regular, which meaning that the action of group G on a set X, for any $x, y \in X$, there is a unique $g,\ s.t.\ gx = y$. In this situation the set X is called G-torsor or principal homogeneous space for G.

Then we state a theorem that shows a important property of endomorphism ring related the isogeny problem over finite field that we shall soon discuss. [13, Theorem 4.5]

Theorem 4 *Let E be the endomorphism algebra of an isogeny class of elliptic curves, $\mathcal{O}$ an order in it which is a possible endomorphism ring. Then every ideal of $\mathcal{O}$ is a kernel ideal for every A with End$(A) = R$.*

- *If E is commutative, the isomorphism classes of curves with endomorphism ring $\mathcal{O}$ form a principal homogeneous space for cl$(\mathcal{O})$*
- *If E is non-commutative, the number of isomophism classes $\#Ell_{\mathcal{O}}(k) = cl(\mathcal{O})$ (the classes of are homogeneous space for Brandt groupoid). Each $\mathcal{O}$ has one or two isomorphism classes of curves with order $\mathcal{O}$, according as the ideal $\mathfrak{p}$ of $\mathcal{O}$ with $\mathfrak{p}^2 = p$ is or is not principal.*

The commutative cases is the key to reduce isogeny problem to hidden abelian shift problem which can be solved in quantum subexponential time.

2.4 Modular polynomial and isogeny graph

Velu's formulas tells us that given kernel points we can compute the isogeny. However, there is another tool for computing isogenies without dealing with the kernel subgroups.

Let l be an integer and $l \geq 2$. The **modular polynomial** $\Phi_z(x, y) \in \mathbb{Z}[x, y]$, with the following remarkable property: A pair $j, j' \in \bar{\mathbb{F}}_q$ satisfies $\Phi_l(j, j') = 0$ if and only if there are two elliptic curves E, E' over $\bar{\mathbb{F}}_q$, with $j(E) = j$ and $j(E') = j'$, and an isogeny $\phi :\ E \to E'$. It can be concluded easily that $\Phi_l(j, j') = 0 \ \Leftrightarrow\ \Phi_l(j', j) = 0$ due to the dual isogeny.

The modular polynomial has very high degree and very large coefficients, so just representing the modular polynomial is actually a hard problem, there are some works related to it. When l is prime then $deg_x(\Phi_l(x, y)) = l + 1$, and indeed $\Phi_l(x, y) = x^{l+1} + y^{l+1} + x^l + y^l + lower\ terms$. It requires $O(l^3 log(l))$ to represent Φ_l.

Hence, given an elliptic curve E over $\mathbb{F}_q$, we can find all j-invariants which are l-isogenous to E by computing the univariate polynomial $\Phi_l(j(E), y) \in \mathbb{F}_q[y]$ and computing its roots in $\bar{\mathbb{F}}_q$. An algorithm due to Elkies allows us to compute the kernel of the corresponding isogeny (in $O(exp(l))$ time) given E and $j' = j(E')$. Along with Velu formulas, we can compute the corresponding isogeny in $O(exp(l))$, that's why we usually take l small in practice.

For elliptic curves over $\mathbb{F}_q$ and l a prime, the **l-isogeny graph** over $\mathbb{F}_q$ is the directed graph $G_l(\mathbb{F}_q)$, whose vertices is the set of j-invariants of elliptic curves over $\mathbb{F}_q$, and whose edges are the pair $(j(E), j(E'))$ with multiplicity equal to the multiplicity of $j(E')$ as a root of $\Phi_l(j(E), Y)$. So the graph is not a simple graph. It can be a multi-graph with more distinct edges between two vertices(and it may have self-loop). For $j \neq 0, 1728$, the multiplicities of the edges $(j(E), j(E'))$ and $j(E'), j(E)$ are the same, which implies we can consider the isogeny graph as an undirected graph without taking the vertices $0, 1728$ into consideration.

Here we give a brief introduction about the isogeny graph, more properties will be refered later.
An elliptic curve which is isogenous to a supersingular curve is also supersingular by Tate' theorem. So the connected component in the l-isogeny graph is either ordinary or supersingular.

The ordinary component in l -isogeny graph is called **l-volcano**, which is a connected undirected graph whose vertices have a level structure meaning that vertices can be divided into $V_0, \cdots, V_d$. V_0 is called the surface or carter, and is a cycle. Each vertex in V_i $(i > 0)$ has exactly one neighbor in V_{i-1}, and all vertices have degree $l + 1$, expect for vertices in V_d with degree 1. l-volcano can be seen as a forest, several exactly the same complete l ary trees are attached to the surface.

The supersingular component has a totally different structure. Every j-invariant of supersingular over $\mathbb{F}_p$ lies in its algebraic closure $\mathbb{F}_{p^2}$, so $\Phi_l(j(E), Y)$ have $l + 1$ roots in $\mathbb{F}_{p^2}$. Hence, the supersingular isogeny graph over $\mathbb{F}_{p^2}$ is a regular graph with degree $l + 1$. There is only one supersingular component in l-isogeny graph. Furthermore, it is an expander graph ,which means it has a good mixing property. Moreover, it's Ramanujan graph, which implies its the optimal expander graph. Such properties led to its widely application in supersingular isogeny-based cryptography, which is one of the post-quantum cryptography in NIST. More detailed properties about supersingular isogeny graph can be refered to.

3 Constructing isogenies between ordinary curves

3.1 Endomorphism ring in ordinary cases

For an ordinary curve over $\mathbb{F}_q$, its endomorphism ring is an imaginary quadratic order $\mathcal{O}_\Delta$ (also a $\mathbb{Z}$-module of rank 2), with $\Delta < 0$. Here Δ is a negative integer, a discriminant of an imaginary quadratic order that uniquely determines $\mathcal{O}$. We can define $\mathrm{Ell}_{\mathbb{F}_q}(\Delta)$, the set of all $\overline{\mathbb{F}}_q$-isomorphism classes over $\mathbb{F}_q$, whose endomorphism ring is Δ. Because two elliptic curves over a finite field are isogenious if and only if they have the same cardinality. So we can define $\mathrm{Ell}_{\mathbb{F}_q,n}$ that is a subset of $\mathrm{Ell}_{\mathbb{F}_q}$ whose elements are all curves with cardinality n. For each isomorphism classes, we can take its j-invariant as its representative.

An isogeny between too curves is called **horizontal** isogeny if they have the same endomorphism ring. Any separable horizontal isogeny $\phi : E \to E'$ between curves in $\mathrm{Ell}_{\mathbb{F}_q,n}(\mathcal{O}_\Delta)$ can be specified, up to isomorphism, by giving $E,\ ker(\phi)$. [7, III.4.12]. The kernel of ϕ can be represented as an ideal in $\mathcal{O}_\Delta$. So we can identified an isogeny $\phi_{\mathfrak{a}} : E \to E_{\mathfrak{a}}$ by its kernel or by an ideal $\mathfrak{a}$. The ideal's isomorphism corresponds to lattice's homothety, and isomorphism of the image of ϕ. Thus there is a group action:

$$\mathrm{cl}(\mathcal{O}_\Delta) \times \mathrm{Ell}_{\mathbb{F}_q,n}(\mathcal{O}_\Delta) \to \mathrm{Ell}_{\mathbb{F}_q,n}(\mathcal{O}_\Delta)$$
$$[\mathfrak{a}]j(E) = j(E_{\mathfrak{a}})$$

Because the endomorphism ring is commutative, $\mathrm{Ell}_{\mathbb{F}_q,n}(\mathcal{O}_\Delta)$ is principal homogeneous space for $\mathrm{cl}(\mathcal{O}_\Delta)$.

3.2 Reduce to abelian hidden shift problem

abelian hidden shift problem Let A be a known finite abelian group, and $f_0, f_1 :\ A \longrightarrow S$ be a black-box function, where S is a finite set. f_0, f_1 are said to hide a shift $s \in A$ if f_0 is injective and $f_1(x) = f_0(xs),\ i.e.\ f_1$ is a shifted version of f_0. The abelian hidden shift problem is to determine the shift s using queries to the black-box function.

Isogeny construction can be reduced to abelian hidden shift problem via the group action. Given two horizontally isogeneous curves E_0, E_1 with endomorphism ring $\mathcal{O}_\Delta$, we can define two functions $f_0, f_1 : \mathrm{cl}(\mathcal{O}_\Delta) \longrightarrow \mathrm{Ell}_{\mathbb{F}_q,n}(\mathcal{O}_\Delta)$ with a hidden shift $[\mathfrak{s}]$, where $[\mathfrak{s}]$ satisfies that $[\mathfrak{s}]j(E_0) = j(E_1)$. Then we can specify $f_b([\mathfrak{a}]) = [\mathfrak{a}]j(E_b),\ b \in \{0, 1\}$. It follows that we can find the ideal class $[\mathfrak{s}]$ by solving the abelian hidden shift problem, which labels the desired isogeny.

The abelian hidden shift can be reduced to hidden subgroup of the "dihedral" group. If we define $f(x, b) := f_b(x),\ b \in \mathbb{Z}/2\mathbb{Z}$. (We denote $\mathbb{Z}/n\mathbb{Z}$ as $\mathbb{Z}_n$ in the following for simplicity without causing confusion. Sometimes $\mathbb{Z}_p$ means the localization of $\mathbb{Z}$ on p.) Then we can consider the semidirect product $A \rtimes \mathbb{Z}_2$. Because $f_0 \neq f_1$, we take:

$$\psi : \mathbb{Z}_2 \to \mathrm{Aut}(A)$$
$$\psi_b = \begin{cases} \psi_0 :\ x \longmapsto x \\ \psi_1 :\ x \longmapsto x^{-1} \end{cases}$$

Here ψ_b denotes the image of b. We want to find $s,\ s.t.\ f_0(xs) = f_1(x),\ \forall x \in A$. In $A \rtimes_\psi \mathbb{Z}_2,\ (xs, 0)(s^{-1}, 1) = (x, 1)$, so finding s in A equals finding the subgroup $\langle (s, 1) \rangle$ in $A \rtimes_\psi \mathbb{Z}_2$. If A is a cyclic group, the problem is finding the hidden subgroup of a dihedral group, which can be solved in quantum subexponential time. We can call the group $A \rtimes_\psi \mathbb{Z}_2$ A-dihedral group. According to the structure theorem of finitely generated abelian group, we can decompose the group A into direct product of some cyclic groups. Then we can get the algorithm:

Algorithm 1: Isogeny computation between ordinary curves defined over a finite field

---

Input: A discriminant of $\Delta < 0$, and Weierstrass equations of horizontally isogeneous ordinary elliptic curves E_0, E_1 defined over $\mathbb{F}_q$ with characteristic p
Output: $[\mathfrak{s}] \in \mathrm{cl}(\mathcal{O}_\Delta),\ s.t.\ [\mathfrak{s}]j(E_0) = j(E_1)$

1: Decompose $\mathrm{cl}(\mathcal{O}_\Delta) = \langle[\mathfrak{a}_1]\rangle \oplus \cdots \oplus \langle[\mathfrak{a}_k]\rangle$, where $|\langle[\mathfrak{a}_i]\rangle| = n_i$
2: Solve the hidden shift problem defined by $f_0, f_1 : \mathbb{Z}_{n_1} \times \cdots \times \mathbb{Z}_{n_k} \longrightarrow \mathrm{Ell}_{\mathbb{F}_q,n}(\mathcal{O}_\Delta)$ satisfying $f_c(x_1, \cdots, x_k) = ([\mathfrak{a}_1]^{x_1} \cdots [\mathfrak{a}_k]^{x_k})j(E_c)$, with hidden shift $(s_1, \cdots, s_k) \in \mathbb{Z}_{n_1} \times \cdots \times \mathbb{Z}_{n_k}$
3: Output $[\mathfrak{s}] = ([\mathfrak{a}_1]^{x_1} \cdots [\mathfrak{a}_k]^{x_k})$

Let's analyse the complexity. Step 1 is similar to finding hidden subgroup in abelian group, and its time complexity is only polynomial time. Step 2, we need to compute the group action and HSP in dihedral group. Group action can be computed in subexponential time. [14, Prop 4.4]. The latter can be implemented using Kuperberg's algorithm [15] or Regev's algorithm [16] in quantum subexponential time. Kuperberg's sieve requires superpolynomial space. Regev's algorithm is slightly slower than Kuperberg's, but it only requires polynomial space.

4 Computing supersingular isogeny by Grover's algorithm

The main reference for this section is [17].

4.1 More detailed properties about supersingular isogeny graph

For some practical reasons, we take prime $p > 3$ for simplicity. Let $X(\bar{\mathbb{F}}_p, l)$ be the supersingular l-isogeny graph defined over $\bar{\mathbb{F}}_p$. Since the j-invariant of supersingular elliptic curves always lies in $\mathbb{F}_{p^2}$, so we consider $X(\mathbb{F}_{p^2}, l)$ instead. It should be noted that the isogenies are defined over $\mathbb{F}_p$ in general.

Let S_{p^2} denotes the vertices set of $X(\mathbb{F}_{p^2}, l)$. There is a well-known result [7, Theorem V.4.1(c)]:

$$\#S_{p^2} = \lfloor \frac{p}{12} \rfloor + \begin{cases} 0 & \text{if } p \equiv 1 \ \ mod\, 12 \\ 1 & \text{if } p \equiv 5, 7 \ \ mod\, 12 \\ 2 & \text{if } p \equiv 11 \ \ mod\, 12 \end{cases}$$

And denote the set of j-invariants lie in $\mathbb{F}_p$ as S_p. Then:

$$\#S_p = \begin{cases} \frac{h(-4p)}{2} & \text{if } p \equiv 1 \ mod\, 4 \\ h(-p) & \text{if } p \equiv 7 \ mod\, 8 \\ 2h(-p) & \text{if } p \equiv 3 \ mod\, 8 \end{cases}$$

where $h(d)$ is the class number of the imaginary quadratic field $\mathbb{Q}(\sqrt{h})$. [9, Theorem 14.18]. And $h(d) \in \tilde{O}(\sqrt{d})$. It follows that $\#S_p \in \tilde{O}(\sqrt{p})$, and $\#S_{p^2} \in O(p)$.

The algorithm in [17] can be described in the following diagram.

$$\begin{tikzcd}
E/\mathbb{F}_q \arrow[r, dotted, bend left, "\tilde{O}\left(p^{1/2}\right)"] \arrow[d, "\tilde{O}\left(p^{1/4}\right)"'] & E'/\mathbb{F}_q \arrow[d, "\tilde{O}\left(p^{1/4}\right)"] \\
E_1/\mathbb{F}_p \arrow[r, bend right, "\text{subexponential complexity}"'] & E_2/\mathbb{F}_p
\end{tikzcd}$$

Algorithm 2: Isogeny computation between supersingular curves defined over a finite field

Input: Supersingular curves E, E' defined over $\mathbb{F}_q$ with characteristic p
Output: An isogeny between E and E'

1: Find $\phi :\ E \rightarrow E_1$ where E_1 is defined over $\mathbb{F}_p$
2: Find $\psi :\ E' \rightarrow E_2$ where E_2 is defined over $\mathbb{F}_p$
3: Find $\tau :\ E_1 \rightarrow E_2$
4: Return $\hat{\psi} \circ \tau \circ \phi$

The complexity of step 1,2 is $\tilde{O}(p^{1/4})$, and the complexity of step 3 is subexponenial.

4.2 Quantum search for a curve defined over $\mathbb{F}_p$

In this subsection, we will introduce the algorithm of step 1,2 using Grover search [18], which make full use of the expander property of the supersingular isogeny graph.
With a quantum computer, searching an unsorted database of N can be implement in complexity in $O(\sqrt{N})$ using Grover's algorithm, while $O(N)$ with classical computer. Actually $O(\sqrt{N})$ and $O(N)$ are also tight bounds of searching problem in unsorted database with quantum and classical computer [19].
While Grover's algorithm need arbitrary query on elements in the database, maybe in supersingular isogeny graph arbitraty query can not be implemented very easily, but it is not a key point. Here we elaborate the expander property [17, Proposition 1]:

Theorem 5 *Under the Generalized Riemann Hypothesis, there is a probability at least $\frac{\pi}{2^{\gamma}} \frac{1}{p^{1/2}}$ that a random $3-$isogeny path of length*

$$\lambda \geq \frac{log(\frac{2}{\sqrt{6e^{\gamma}}} p^{3/4})}{log(\frac{2}{\sqrt{3}})}$$

passes through a supersingular j-invariant defined over $\mathbb{F}_p$, where γ is the Ruler constant.

This theorem is a direct application of the expander property of the Ramanujan graph.Proof can be referred to [20]. Let's narrate the theorem in short. Note that $\lambda \in O(log(p))$, the probability is $O(p^{-1/2})$, which corresponds with the ratio of $\frac{\#S_p}{\#S_{p^2}}$. This property is called the mixing property, meaning that random walk of length $\lambda \in O(log(p))$ can be almost uniform distribution. Such property of expander graph usually be used for constructing pseudo-random sequence in cryptography and complexity theory.

Then we can enumerate all of the random path with length λ. because of the regularity of the graph, it's easy to label the path. Here we consider all $3-$isogeny paths, then they can be labeled by $\{0, 1, 2\}^{\lambda}$. Since the path length is just $O(log(p))$, so we can construct an oracle with can determine whether the path passes through points in $\mathbb{F}_p$ with very little cost. Then we can implement Grover's algorithm to find out the desired path. (There are many paths meeting our requirement, but one path is enough.) And Grover's algorithm provides a quadratic speedup for the procedure. This is the key point for [17]'s speedup.

4.3 Computing an isogeny between curves defined over $\mathbb{F}_p$

This procedure relies on the complex multiplication theory. The endomorphism ring $\text{End}(E)$ of a supersingular elliptic curve is an order of a quaternion algebra. While as shown in [21, Proposition 2.5], the endomorphism ring $\text{End}_{\mathbb{F}_p}(E)$ defined over $\mathbb{F}_p$ is isomorphic to an order $\mathcal{O}$ in the imaginary quadratic number field $\mathbb{Q}(\sqrt{-p})$.

Theorem 6 *There is an one-to-one correspondence between:*

$$\left\{ \begin{array}{c} \textit{supersingular elliptic} \\ \textit{curves over } \mathbb{F}_p \end{array} \right\} \longleftrightarrow \left\{ \begin{array}{c} \textit{elliptic curves } E \textit{ over } \mathbb{C} \\ \textit{with End}(E) \in \{\mathbb{Z}[\sqrt{-p}], \mathcal{O}_K\} \end{array} \right\}.$$

The theorem implies that properties of supersingular elliptic curves over $\mathbb{F}_p$ are similar to ordinary cases. So there is also a transitive action of $Cl(\mathcal{O})$ on the $\mathbb{F}_p-$isomorphism classes. As in the ordinary case, an ideal class $\mathfrak{a}$ acts via an isogeny of degree $\mathcal{N}(\mathfrak{a})$. Then we get an injective function induced by each supersingular elliptic curve defined over

$\mathbb{F}_p$:

$$\begin{aligned} f_E:\ Cl(\mathcal{O}) &\longrightarrow \quad \mathbb{F}_p - \text{isomorphism classes of curves over } \mathbb{F}_p \\ [\mathfrak{b}] &\longmapsto \quad \text{action of } [\mathfrak{b}] \text{ on the class of } E \end{aligned}$$

Hence, given two supersingular elliptic curves$E_1,\ E_2$ defined over $\mathbb{F}_p$. We can reduce the isogeny finding problem to the hidden abelian shift problem, $i.e.$ finding the ideal class $[\mathfrak{a}]$, such that $f_{E_2}(x) = f_{E_1}([\mathfrak{a}] \cdot x)$ for all x. It equals that E_2 is the image of the action of $[\mathfrak{a}]$ on E_1.

Then we can implement the procedure as in ordinary case.

5 SIKE and claw finding

This section mainly refers to [22, 23].

5.1 Supersingular isogeny key encapsulation

This section is mainly referred to [24]. In [6], De Feo, Jao and Plût presented an encryption protocol relying on the difficulty of computing an isogeny between supersingular elliptic curves. Given a secret point S of a supersingular curve E over $\mathbb{F}_{p^2}$, and a public point R, SIKE can be described by the following commutative diagram in general:

$$\begin{tikzcd} E \arrow[r] \arrow[d] & E/\langle S\rangle \arrow[d] \\ E/\langle R\rangle \arrow[r] & E/\langle S, R\rangle \end{tikzcd}$$

Let's take SIDH protocol (supersingular isogeny Diffie-Hellman) for instance. For the Jao and De Feo system [25] ([5] is a more detailed version), there is a special set-up. Choose two distinct small primes l_1, l_2, typically with $l_1 = 2, l_2 = 3$, and choose $e_1, e_2 \in \mathbb{N},\ s.t.\ l_1^{e_1} \approx l_2^{e_2} \approx 2^{\lambda}$, where λ is a security parameter. Then choose a random integer $f \in \mathbb{N}$ satisfying $p := l_1^{e_1} l_2^{e_2} f \pm 1$ is a prime. Constructing an supersingular elliptic curve E over $\mathbb{F}_{p^2}$, which can be done by Broker's alogrithm [26] with time complexity $\tilde{O}((log^3(p)))$ under generalized Riemann Hypothesis. Group theoretically, $E(\mathbb{F}_{p^2})$ is a direct product of two cyclic groups with order $l_1^{e_1} l_2^{e_2} f$. Selecting points $R_1, S_1 \in E[l_1^{e_1}],\ s.t.\ \langle R_1, S_1\rangle = E[l_1^{e_1}]$, and $R_2, S_2 \in E[l_2^{e_2}],\ s.t.\ \langle R_2, S_2\rangle = E[l_2^{e_2}]$. The **SIDH system parameters** are (E, R_1, S_1, R_2, S_2).

The protocal works similarly as Diffie-Hellman. Alice chooses a secret subgroups of $E[l_1^{e_1}]$ by choosing an integer $0 \leq a < l_1^{e_1}$ and setting $T_1 = R_1 + [a]S_1$. Alice computes an isogeny $\phi_A:\ E \to E_A$ with kernel generated by T_1 and publishes $E_A, \phi_A(R_2), \phi_A(S_2)$. Similarly Bob chooses $0 \leq b < l_2^{e_2}$, computes $\phi_B:\ E \to E_B$, with kernel generated by $T_2 := R_2 + [b]S_2$, and publishes $(E_B, \phi_B(R_1), \phi_B(S_1))$. To compute the shared key, Alice computes

$$T_1' = \phi_B(R_1) + [a]\phi_B(S_1) = \phi_B(R_1 + [a]S_1) = \phi_B(T_1).$$

Then computes isogeny $\phi_A' : E_B \to E_{AB}$ with kernel generated by T_1'. The composition $\phi_A' \circ \phi_B$ has kernel $\langle T_1, T_2\rangle$. Similarly Bob computes an isogeny $\phi_B' : E_B \to E_{AB}'$, whose kernel is $\langle \phi_A(R_2) + [b]\phi_A(S_2)\rangle$. E_{AB} and E_{AB}' are not necessarily the same but isomorphic, so $j(E_{AB}) = j(E_{AB}')$. We choose $j(E_{AB})$ as the shared key.

We denote $G_A = \langle T_1\rangle$ is a subgroup of $E[l_1^{e_1}]$, G_B in the same way. Then $E_A = E/G_A,\ E_B = E/G_B$, it follows that $E_{AB} = E/\langle G_A, G_B\rangle$. This process can be shown in the following commutative diagram.

$$\begin{tikzcd}
E \arrow[rr, "/\langle T_1 \rangle"] \arrow[dd, "/\langle T_2 \rangle"'] & & E_A \arrow[dl, "/\langle T_2' \rangle"] \\
& E'_{AB} \arrow[dr, dotted] & \\
E_B \arrow[rr, "/\langle T_1' \rangle"'] & & E_{AB}
\end{tikzcd}$$

Given a hash function family $\mathcal{H} = \{H_k : \ k \in K\}$, where K is a finite set. H_k sends j-invariant to an message $m \in \{0,1\}^M$ Alice encrypt the message$m$ into ciphertext c:

$$h = H_k(j(E_{AB}))$$

$$c = h \oplus m.$$

To break the key exchange protocol is to solve a specific isogeny problem.

5.2 Claw finding and meet in the middle

This subsection is mainly referred to [22, 23].

claw finding: Let $f : X \to S$ and $g : Y \to S$ be two functions. Find $x \in X$ and $y \in Y, \ s.t. \ f(x) = g(y)$, if it exists.

It's closely connected with the isogeny problem. Given two curves $E_0, \ E_1$, there is a secret isogeny $\phi : E_0 \to E_1$ of degree l^e. We take X to be the set of all isogenies $\phi_i : E_0 \to E_1$ of degree $l^{e/2}$ and Y be all isogenies $\psi_j : E_1 \to E_j$ of degree $l^{e/2}$. The functions $f, \ g$ take an isogeny as the input and the j-invariant of the image of the isogeny as output. If there is a claw (x, y), which implies that $\phi_x : E0 \to E_{01}, \ \psi_y : E_1 \to E_{01}$. From the claw, we can get the desired isogeny $\gamma : E_0 \to E_1$ with degree l^e by compositing the claw isogenies $\gamma = \hat{\psi}_y \circ \phi_x$.

For SIDH, S is the set of all j-invariants S_{p^2} of supersingular elliptic curves defined over $\mathbb{F}_{p^2}$. And $|S| \approx p/12$. For X, there are $(l+1)l^{e/2-1} \approx l^{e/2} \approx p^{1/4}$ isogenies of degree $l^{e/2}$ from E_0, so do isogenies form E_1. Hence, $|X| \approx |Y| \approx p^{1/4} \ll |S|$, which implies that the claw is very likely to be the unique.

In classical cases, the state of the art is called **Meet-in-the-middle**. Given a memory parameter R, we construct sorted lists L_x, L_y, L_x consists of random elements $(x, f(x))$, so does L_y. And $|L_x| = |L_y| = R$. We execuate the following procedure, until a claw is found.

- Delete a random element of L_x. Choose a new random element of X, and get the pair $(x, f(x))$. Check if $\exists y \in L_y, \ s.t. \ f(x) = g(y)$, if not insert $x, f(x)$ into L_x, and repeat the procedure.

Let's compute the time complexity. The cost of constructing list is $O(Rlog(R))$ (ignore the cost of evaluate f, g, because the length of each isogeny path is $O(log(p))$). The costs of inserting and searching are $O(log(R))$. The probability of the algorithm detecting a claw (x, y) is no less than $\frac{R^2}{|X||Y|}$. And we need to make the list totally "fresh" because the probability after updating highly depends on the previous cases. Above all, the total cost is :

$$O(Rlog(R) + \frac{|X||Y|}{R^2} R(log(R) + log(R))$$

Taking $R = |X| + |Y|$, then we get the optimal time cost, which can be shown directly by the inequality of arithmetic and geometric means. The optimal time complexity of this method is $\tilde{O}(p^{1/4})$. To get this time complexity, we also need $\tilde{O}(p^{1/4})$ bits of memory.

5.3 Quantum speedup using Grover's search

Can we do better than the classical cases? We make a simple attempt. For searching problem, it's natural to consider Grover's algorithm or quantum walk. But list all pairs (x, y) and search in Brute Force cost too much. For example, if we take X as the set of all paths with length a starting from E_0, then Y is the set of all $e/2 - a$ long path from E_1. $|X| = l^a,\ |Y| = l^{e-a}$. Then the set of pair (x, y) has $l^e = p^{1/2}$ elements. Grover's search can only speed up to $O(p^{1/4})$. It's same as the classical meet in the middle. Because we have not made full use of the structure of this problem, too much redundant information has been searched. We need to find the path in Y whose end is in X, so we can enumerate every element in X and using Grover's algorithm to search the desired element in Y. The time complexity is $O(|X|\sqrt{|Y|})$. However, determining whether a path in Y whose end is in the set of ends of X or not can be done more efficiently. Because we can sort all the elements in X by their j-invariant and just using binary search to determine. Let's analyze this algorithm's complexity. The cost of constructing ordered set X is $O(|X|log(|X|))$, and Grover's search in $|Y|$ costs $O(log(|X|)\sqrt{|Y|})$. So the total time complexity is $O(|X|log(|X|) + log(|X|)\sqrt{|Y|})$, with $|X||Y| = O(p^{1/2})$, taking $|X| = p^{1/6}$ leads to the optimal results $\tilde{O}(p^{1/6})$.

5.4 Other quantum algorithm in $\tilde{O}(p^{1/6})$

Here we show the table in [22], which exhibit the state of the art of quantum claw finding, and I make a revise on it.

Type	Variant	Specialty
Grover	Tiny-Claw [27]	Lowest circuit complexity*
	Parallel Tiny-claw [28]	Offsets a lot of the query cost
Random Walk [29]	Tani [30]	Lowest number of queries*
	Distinguished Points [31]	Lowest gate cost*
Multi-Grover [32]	Distinguished Points [31]	Best parallelism

Note: (In [22]) Under reasonable limits on total runtime, all of them perform worse than classical algorithm by van Oorschot Wiener [33], even without accounting for the overheads of quantum computing. * indicates that the claim only holds with no runtime limit.

Jean-François Biasse and Benjamin Pring [27] made an improvement on the quantum circuit complexity of a specified Grover's algorithm, which has an asymptotic improvement on cryptanalysis of SIKE [23].

In [28], Reza Azarderakhsh , Jean-François Biasse, et al made an optimization and improvement an Tiny Claw which can be executed in parallel and offsets a lot of query cost. Furthermore, it only need several independent small quantum computers and classical connections rather than quantum connectivity. The costs is allowed to be balanced between the relative cost of quantum error correction and classical memory.

Tani's claw finding algorithm [30] has the lowest query complexity, and it is very famous in cryptography.

In [31], Samuel Jaques and André Schrottenloher provide better quantum parallelization methods using distinguished points technique proposed by van Oorschot Wiener [33].

6 Our approaches and hardness of the problem

We have been trying to design a faster quantum algorithm to solve the isogeny problem but finally failed. Here I want to elaborate on our ideas, and make an explanation on why they failed that is highly connected with the structure of the problem.

Having seen Biasse, Jao and Sankar's algorithm in [17], we want to make an improvement on it. Because the Grover's search using in [17, Section 4] seems that it doesn't make full use of the structure of the supersingular isogeny graph. We want to find a path starting from E_0 and passing through j-invariant in $\mathbb{F}_p$ faster than $\tilde{O}(p^{1/4})$.

Inspired by the element distinctness problem solved by Andris Ambainis [34], which has been improved from $O(N^{3/4})$ to $O(N^{2/3})$. He reduced the problem to quantum walk on Johnson Graph which preserves the problem's structure and make it more clear. This algorithm reaches the lower bound of the problem. However, the element distinctness is not a pure searching problem in an unordered dataset, so it can be improved. For searching problem with no structure, the lower bound is $\Omega(\sqrt{N})$ [19]. I have made some attempts to reduce the isogeny finding problem to quantum walk on Johnson Graph, but they all failed. Because the reduction is trivial, we didn't discover more properties. Thus, the reduction won't be helpful.

Quantum snake walk is said to solve a specific searching problem on glued tree [35] with a exponential speedup. Can the snake walk technique be used here? My answer is probably not. The author didn't make a detailed analysis, and he claimed that maybe in some cases the algorithm doesn't work. In addition, or the most importantly, we know little about the supersingular isogeny graph, just constructing the Hamiltonian is not a easy task. So we can't make a convinced analysis. Besides the above, the task in [35] is very different from our problem. The latter is to find the root of the glued tree, the difficulty is to escape from the overlap of two complete binary trees as possible. Once the walker escape from it, it can reach the target easily. But for quantum walk on isogeny graph, the walker doesn't have the sense of direction due to the high expansion of the graph, random walk on it will mix rapidly, which will hinder us from doing better than Grover's search.

Here let's state the difficulties of the problem.

- The problem seems as a path finding problem, but actually a search problem, because once we find the desired point, we can compute the path using meet-in-the-middle (or the quantum version).
- The desired points $|S_p| = O(\sqrt{p})$, and total points on the graph $|S_{p^2}| = O(p)$. So if we pick points randomly, the probability is $|S_p|/|S_{p^2}| = O(p^{-1/2})$. In [17] listing all paths makes use of the expander property, which means the probability the path ends at a desired point equals to $|S_p|/|S_{p^2}|$ due to "mixing". Longer length of the path won't lead to a better distribution. If we consider this problem as a pure search problem, we can not do better than quadratic speedup using Grover's algorithm.

Hence, to make an improvement, we need to explore more about the structure. [36] shows that maybe the desired points are not distributed on the graph evenly, which depends on the choice of p and l. This is quite a amazing result, even though some of their results by experiment have not been proved. Can we make use of some of these properties?

Quantum fast-forwarding (QFF) [37] is a powerful tool which allows to approximately prepare the quantum samples of random walk in the square root of the random walk runtime. And it can be used to speed up some property-testing problem on a graph, for example, expansion testing. At first, QFF will not be useful to estimating the random walk in our problem, because we just care about path with length $O(log(p))$, QFF won't help a lot. Then can expansion testing be helpful?

In [36], authors considered the points S_p that j-invariant is in $\mathbb{F}_p$ along with the graph consisting of vertex set S_p and edge set $\mathbb{F}_p$-isogeny and $\bar{\mathbb{F}}_p$ isogeny. Maybe the topology of these sets contains some information, but these information are local on the graph. While expansion testing is to detect the global property of the graph. This is the essential point that why expansion testing may be not a good choice for the isogeny problem.

We also have made a few algebraic approaches, but we think there is no hope to make a improvement using algebraic method. There are no algebraic algorithm performs better than graph algorithm in the supersingular isogeny problem. And we are not experts in this field.

Another interesting results about supersingular isogeny graph is [38], which shows an explicit connection between the graph and Bruhat-Tits trees. Can we get some global information about the isogeny graph from the Bruhat-Tits trees? Bruhat-Tits trees is related to p-adic lattice, maybe some technique on p-adic analysis will help? I think it's a interest problem.

7 Acknowledgement

This work is under my undergraduate advisor Tongyang Li's guidance. Thanks for his patience. Thanks to my partner Boyang Chen and Han Luo, they contributed a lot for the discussion during the researching. And thank Qi Liu, she helped me correct language mistakes and polish this paper.

References

[1] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[2] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[3] John Proos and Christof Zalka. Shor's discrete logarithm quantum algorithm for elliptic curves. *Quantum Info. Comput.*, 3(4):317–344, jul 2003.

[4] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[5] Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. Cryptology ePrint Archive, Report 2011/506, 2011. https://ia.cr/2011/506.

[6] C. Costello L. D. Feo B. Hess A. Jalali D. Jao B. Koziel B. LaMacchia P. Longa M. Naehrig J. Renes V. Soukharev R. Azarderakhsh, M. Campagna and D. Urbanik. Supersingular isogeny key encapsulation – submission to the nist's post-quantum cryptography standardization process, 2017. Available at https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/ submissions/SIKE.zip.

[7] Joseph H. Silverman. *The Arithmetic of Elliptic Curves*. Springer New York, New York, NY, 2009.

[8] Joseph H. Silverman. *Advanced Topics in the Arithmetic of Elliptic Curves*. Springer New York, New York, NY, 1994.

[9] David A.Cox. *Complex Multiplication*. John Wiley Sons, Ltd, 2013.

[10] Andrew Sutherland. Elliptic curves. https://math.mit.edu/classes/18.783/2019/lectures.html, 2019. Lecture notes from a course(18.783) at MIT.

[11] Andrew M. Childs and Wim van Dam. Quantum algorithms for algebraic problems. *Reviews of Modern Physics*, 82(1):1–52, jan 2010.

[12] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[13] William C. Waterhouse. Abelian varieties over finite fields. *Annales scientifiques de l'École Normale Supérieure*, Ser. 4, 2(4):521–560, 1969.

[14] Gaetan Bisson. Computing endomorphism rings of elliptic curves under the grh. *Journal of Mathematical Cryptology*, 5(2):101–114, 2012.

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

[16] Oded Regev. A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space. *arXiv*, June 2004. 7 pages, 1 figure.

[17] Jean-François Biasse, David Jao, and Anirudh Sankar. A quantum algorithm for computing isogenies between supersingular elliptic curves. In Willi Meier and Debdeep Mukhopadhyay, editors, *Progress in Cryptology – INDOCRYPT 2014*, pages 428–442, Cham, 2014. Springer International Publishing.

[18] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery.

[19] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum search. In *Proceedings of the Workshop on Physics of Computation: PhysComp'96*, pages 36–43, 1996.

[20] David Jao, Stephen D. Miller, and Ramarathnam Venkatesan. Expander graphs based on grh with an application to elliptic curve cryptography. *Journal of Number Theory*, 129(6):1491–1504, 2009.

[21] Christina Delfs and Steven D. Galbraith. Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. *Des. Codes Cryptography*, 78(2):425–440, feb 2016.

[22] Samuel Jaques. Quantum claw finding. https://www.isogenyschool2020.co.uk/schedule/8-Quantum-Claw-Finding.pdf, 2020. Lecture notes from isogeny-based cryptography school online.

[23] Samuel Jaques and John M. Schanck. Quantum cryptanalysis in the ram model: Claw-finding attacks on sike. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 32–61, Cham, 2019. Springer International Publishing.

[24] Steven D Galbraith and Frederik Vercauteren. Computational problems in supersingular elliptic curve isogenies. *Quantum Information Processing*, 08 2018.

[25] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Bo-Yin Yang, editor, *Post-Quantum Cryptography*, pages 19–34, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[26] Reinier Broker. Constructing supersingular elliptic curves. *Frontiers of Combinatorics and Number Theory*, 01 2009.

[27] Jean-François Biasse and Benjamin Pring. A framework for reducing the overhead of the quantum oracle for use with grover's algorithm with applications to cryptanalysis of sike. *Journal of Mathematical Cryptology*, 15(1):143–156, 2021.

[28] Reza Azarderakhsh, Jean-François Biasse, Rami El Khatib, Brandon Langenberg, and Benjamin Pring. Parallelism strategies for the tuneable golden-claw finding problem. *International Journal of Computer Mathematics: Computer Systems Theory*, 6(4):337–363, 2021.

[29] Frederic Magniez, Ashwin Nayak, Jeremie Roland, and Miklos Santha. Search via quantum walk. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing*, STOC '07, page 575–584, New York, NY, USA, 2007. Association for Computing Machinery.

[30] Seiichiro Tani. An improved claw finding algorithm using quantum walk. In Luděk Kučera and Antonín Kučera, editors, *Mathematical Foundations of Computer Science 2007*, pages 536–547, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[31] Samuel Jaques and André Schrottenloher. Low-gate quantum golden collisionnbsp;finding. In *Selected Areas in Cryptography: 27th International Conference, Halifax, NS, Canada (Virtual Event), October 21-23, 2020, Revised Selected Papers*, page 329–359, Berlin, Heidelberg, 2020. Springer-Verlag.

[32] Gray Oliver Harrow Aram W. Kutin Samuel Linden Noah Shepherd Dan Beals Robert, Brierley Stephen and Stather Mark. Efficient distributed quantum computing. *Proceedings of the Royal Society A*, 2013.

[33] Paul C. van Oorschot and Michael J. Wiener. Parallel collision search with cryptanalytic applications. *J. Cryptology*, 12:1–28, 1999.

[34] A. Ambainis. Quantum walk algorithm for element distinctness. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 22–31, 2004.

[35] Ansis Rosmanis. Quantum snake walk on graphs. *Physical Review A*, 83(2), feb 2011.

[36] Sarah Arpin, Catalina Camacho-Navarro, Kristin Lauter, Joelle Lim, Kristina Nelson, Travis Scholl, and Jana Sotáková. Adventures in supersingularland. *Experimental Mathematics*, pages 1–28, 2021.

[37] Simon Apers and Alain Sarlette. Quantum fast-forwarding: Markov chains and graph property testing. *arXiv preprint arXiv:1804.02321*, 2018.

[38] Laia Amorós, Annamaria Iezzi, Kristin Lauter, Chloe Martindale, and Jana Sotáková. Explicit connections between supersingular isogeny graphs and bruhat–tits trees. *Cryptology ePrint Archive*, 2021.