Security Analysis of Isogeny-Based Cryptosystems

by

Christopher Leonardi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics & Optimization

Waterloo, Ontario, Canada, 2020

Author's declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of contributions

I am the sole-author of chapters 1, 2, 5, and 6. Chapters 3, 4, and 7 contain materials from the co-authored papers [18], [6], and [68], respectively.

Abstract

Let E be a supersingular elliptic curve over a finite field. In this document we study public-key encryption schemes which use non-constant rational maps from E. The purpose of this study is to determine if such cryptosystems are secure. Supersingular Isogeny Diffie-Hellman (SIDH) and other supersingular isogeny-based cryptosystems are considered. The content is naturally divided by cryptosystem, and in the case of SIDH, further divided by type of cryptanalysis: SIDH when the endomorphism ring of the base elliptic curve is given (as is done in practice), repeated use of keys in SIDH, and endomorphism ring constructing algorithms. In each case the relevent background material is presented to develop the theory.

In studying the security of SIDH when the endomorphism ring of the base curve E is known, one of the main results is the following. This theorem is then used to reduce the security of such an SIDH instantiation to the problem of finding particular endomorphisms in $\mathrm{End}(E)$.

Theorem 1. *Given*

1. *a supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p = N_1N_2 - 1$ for coprime $N_1 \approx N_2$, where N_2 is $\log p$-smooth,*
2. *an elliptic curve E' that is the codomain of an N_1-isogeny $\phi : E \to E'$,*
3. *the action of ϕ on $E[N_2]$, and*
4. *a k-endomorphism ψ of E, where $\gcd(k, N_1) = 1$, and if g is the greatest integer such that $g \mid N_2^2$ and $g \mid k$, then $k' := \frac{k}{g} < N_1$,*

there exists a classical algorithm with worst case runtime $\tilde{O}(k'^3)$ which decides whether $\psi(\ker\phi) = \ker\phi$ or not, but may give false positives with probability $\approx \frac{1}{\sqrt{p}}$. Further, if k' is $\log p$-smooth, then the runtime is $\tilde{O}(\sqrt{k'})$.

In studying the security of repeated use of SIDH public keys, the main result presented is the following theorem, which proves that performing multiple pairwise instances of SIDH prevents certain active attacks when keys are reused.

Theorem 2. *Assuming that the CSSI problem is intractable, it is computationally infeasible for a malicious adversary, with non-negligible probability, to modify a public key $(E_B, \phi_B(P_A), \phi_B(Q_A))$ to some (E_B, R, S) which is malicious for SIDH.*

It is well known that the problem of computing hidden supersingular isogenies can be reduced to computing the endomorphism rings of the domain and codomain elliptic curves. A novel algorithm for computing an order in the endomorphism ring of a supersingular elliptic curve is presented and analyzed to have runtime $O(p^{1/2}(\log p)^2)$.

In studying non-SIDH cryptosystems, four other isogeny-based cryptosystems are examined. The first three were all proposed by the same authors and use secret endomorphisms. These are each shown to be either totally insecure (private keys can be recovered directly from public keys) or impractical to implement efficiently. The fourth scheme is a novel proposal which attempts to combine isogenies with the learning with errors problem. This proposal is also shown to be totally insecure.

Acknowledgements

First and foremost, I would like to thank my supervisor David Jao for his excellent guidance during my graduate studies. I am grateful for all the assistance from the members of the Combinatorics and Optimization department administration, specifically Melissa Cambridge. I am thankful for the helpful conversations with Alfred Menezes and David Jackson, my department examiners Douglas Stebila and Michele Mosca, and my extenal examiners Yu-Ru Liu and David Kohel. I would also like to thank ISARA Corporation for providing me with excellent resources and allowing me to work on their team of post-quantum researchers during my studies. Finally, thank you to my family for their endless support.

Table of contents

List of Figures

List of Tables

Chapter 1

Introduction

The goal of this work is an in-depth study of the security of different supersingular isogeny-based public-key cryptosystems. Let E be a supersingular elliptic curve over a finite field $\overline{\mathbb{F}}_p$. An isogeny is a non-constant rational map from $E/\overline{\mathbb{F}}_p$ to another elliptic curve $E'/\overline{\mathbb{F}}_p$ which is also a group homomorphism between $E(\overline{\mathbb{F}}_p)$ and $E'(\overline{\mathbb{F}}_p)$. If two parties wish to establish a shared secret, they may use, among other things, a random secret isogeny with domain E as their secret key, and the codomain of their isogeny as their public key. In this generic framework, many distinct cryptosystems have been proposed. The main appeal of isogeny-based cryptosystems, as opposed to the classical discrete logarithm based elliptic curve cryptosystems, is their potential post-quantum security. This work examines some of these proposals, and in each case, describes novel attacks, reveals insecurities, or provides proofs of security.

Constructive applications of supersingular elliptic curve isogenies to cryptography were first introduced by Charles, Goren, and Lauter, who proposed a cryptographic hash function [26]. The hash function takes a walk in the supersingular isogeny graph, determined by the input, taking advantage of the random mixing property of the graph. The first work on isogeny-based key establishment is a note published by Couveignes [36] in 2006 (written in 1997) using ordinary elliptic curves, and independently by Rostovtsev and Stolbunov [82], frequently referred to as the CRS scheme. By computing a multitude of prime degree isogenies from a public ordinary elliptic curve E over a field with cryptographically large characteristic, it is believed to be computationally difficult to determine this isogeny from the domain, codomain, and degree. This is the basis of the CRS key establishment scheme. The appeal of this system was that it had exponential security against quantum adversaries, and small public-key sizes, but the downside is the computational inefficiency of computing ordinary isogenies.

Around 2010, Childs, Jao and Soukharev discovered a quantum subexponential-time attack on these two schemes [28], utilizing the commutativity of the ordinary elliptic curves' endomorphism rings. Close to this time Jao and De Feo proposed a new key-exchange scheme, later called supersingular isogeny Diffie-Hellman (SIDH), using supersingular elliptic curves [58]. The benefits of shifting to supersingular elliptic curves is that the computation cost is drastically less than for ordinary elliptic curves, small public-keys are retained, and the subexponential attack of [28] does not apply (supersingular endomorphism rings are non-commutative) making SIDH a good candidate for post-quantum cryptography. The downside is that part of the action of the secret isogeny needs to be published for key-exchange, meaning its security is dependent on a stronger computational assumption than the usual isogeny problem.

Computational improvements have been made to the CRS scheme, resulting in a more efficiently performable key-exchange called CSIDH [23]. Many signature schemes, and other cryptographic utilities have been proposed in recent years [54, 40, 50, 4].

In 2016, NIST announced a competition (but not a competition) to standardize post-quantum encryption and signature schemes [27]. The only isogeny-based entry was Supersingular Isogeny Key Exchange (SIKE) obtained by applying a Fujisaki-Okamoto-like transformation [57] to SIDH. The initial entry consisted of three parameter sets and implementation code, and is in the second round of the contest at the time of this writing.

The work in this thesis begins with Chapter 3, which introduces a novel area of cryptanalysis by studying the relationship between the ℓ-isogeny problem and knowledge of the involved elliptic curves' endomorphism rings. It has been shown that the ℓ-isogeny problem can be solved efficiently if the endomorphism rings of *both* the domain and codomain elliptic curves are known [65]. It has also been argued heuristically that when neither of the endomorphism rings are known, the ℓ-isogeny problem is as difficult as computing the endomorphism rings [77]. The work in this chapter discusses the tractability of the ℓ-isogeny problem in the case where only the endomorphism ring of the domain elliptic curve is known. The proposed algorithms for solving the ℓ-isogeny problem in this chapter also use the revealed torsion information inherent in SIDH to increase their probability of success, making these algorithms non-generic.

The first result of Chapter 3 discusses when an endomorphism of E implies the existence of an endomorphism on some isogenous E'.

Proposition 1. *Let $k, N \in \mathbb{Z}$ be coprime. Let $E/\mathbb{F}_q$ be a supersingular elliptic curve, let $R \in E[N]$, and let $\psi \in \operatorname{End}(E)$ be cyclic with order k. Suppose $\phi : E \to E'$ is an isogeny with kernel $\langle R \rangle$. If $\langle R \rangle$ is fixed (as a subgroup) by ψ, then there exists a cyclic* *endomorphism on E' of degree k. Furthermore, this endomorphism $\psi' \in \mathrm{End}(E')$ has kernel $\phi(\ker \psi)$.*

The converse of this proposition is also true with high probability when $k < \sqrt{p}$: if E and E' are N-isogenous, $\psi \in \mathrm{End}(E)$ with $\deg \psi = k$, and there exists an endomorphism of E' with degree k, then with high probability $\psi(\ker \phi) = \ker \phi$. This pair of results then forms a passive method for revealing information about $\ker \phi$, which is meant to be secret in supersingular isogeny cryptography. An algorithm is described to learn this information.

Theorem 2. *Given*

1. *a supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p = N_1N_2 - 1$ for coprime $N_1 \approx N_2$,*

2. *an elliptic curve E' that is the codomain of an N_1-isogeny $\phi : E \to E'$, and*

3. *a k-endomorphism ψ of E, for some integer k where $\gcd(k, N_1) = 1$ and $k < N_1$,*

there exists a classical algorithm with worst case runtime $\tilde{O}(k^3)$ which decides whether $\psi(\ker \phi) = \ker \phi$ or not, but may give false positives with probability $\approx \frac{1}{\sqrt{p}}$. Further, if k is $\log p$-smooth, then the runtime is $\tilde{O}(\sqrt{k})$.

The remainder of Chapter 3 is a study of a lower bound on the runtime of this algorithm on SIKE parameters where no improvements on the best-known attack is found, and in the multi-party setting where improvements are found.

Chapter 4 discusses the security of repeated use of static keys by both parties, i.e. static-static key-exchange or non-interactive key exchange, in SIDH. The motivation for this cryptographic primitive is to eliminate the need for key-generation in every session. However, Galbraith et al. [53] explained why a static-static implementation of textbook SIDH is vulnerable to a (classical) active attack. This active attack, repeated adaptively, breaks textbook static-static SIDH. Kirkwood et al. [63] showed how a Fujisaki-Okamoto transformation could be applied to SIDH, which would allow for one party to use a static key securely by "indirect validation" of the other participants public key. This modification to SIDH enables secure use of static-ephemeral key-exchange.

As mentioned, the goal of Chapter 4 is achieving secure static-static key-exchange. We present a generic transform of a key-exchange protocol to one which is secure against attacks of the form that broke the static-static textbook SIDH [53]. This goal is realized by having the participants perform multiple key-exchange rounds in parallel and using a key derived from all the computed shared secret values. We show that the security of this multiple-round key-exchange protocol is no less than the original key-exchange protocol. We formalize the attack type of [53], and prove that it no longer applies to the transformed protocol under the assumption that a certain type of *malicious* public key cannot be found. We then prove such malicious public-keys are infeasible to compute for SIDH as long as its underlying security assumption holds (i.e. we reduce the problem of computing a malicious key to the standard SIDH isogeny problem). Chapter 4 also contains an analysis of the number of rounds needed for the static-static SIDH protocol at a 128-bit post-quantum security level, key-size analysis, and a discussion of subsequent work published in this area.

The next chapter, Chapter 5, focuses on the problem of computing the endomorphism ring of a given supersingular elliptic curve, $E/\overline{\mathbb{F}}_p$. The intractability of computing $\mathrm{End}(E)$ is vital to the security of supersingular isogeny-based schemes. The collision resistance and second preimage resistance of the CGL hash function [26] depend on this problem being difficult [77, 46]. Further, the ℓ-isogeny problem can heuristically be solved in polynomial time when the endomorphism rings of both the domain and codomain curve are known [65]. This would lead to a break of SIDH if the endomorphism ring problem were not intractable [53] .

An endomorphism of E is an isogeny with both domain and codomain E. It is known that the ring of endomorphism of E, called $\mathrm{End}(E)$, is isomorphic to some maximal order in a quaternion algebra (ramified at p and ∞), and further this maximal order is unique to E up to the Galois conjugacy class of the j-invariant of E. The first to study the problem of computing $\mathrm{End}(E)$ for an arbitrary supersingular elliptic curve was Kohel in his PhD thesis [64]. His approach generated a finite index subring of $\mathrm{End}(E)$ by finding two independent cycles containing $j(E)$ in the ℓ-isogeny graph of supersingular elliptic curves, as these correspond to endomorphisms of E (with ℓ-power degree). This probabilistic algorithm requires exponential storage and has an expected running time of $O(p^{1+\epsilon})$. Heuristically, one expects this algorithm to be called $O(\log p)$ many times before the entire endomorphism ring of E can be determined [77]. Instead of computing cycles, one may also compute a smooth [54] (or power of ℓ [40]) degree isogeny between E and an elliptic curve with a known endomorphism ring. Delfs and Galbraith's [41] probabilistic algorithm for this problem requires polynomial storage and has expected runtime $\tilde{O}(p^{1/2})$.

Chapter 5 presents a novel algorithm for computing independent cycles containing E in the ℓ-isogeny supersingular elliptic curve graph. As mentioned, these cycles correspond to endomorphisms and form a suborder of $\mathrm{End}(E)$ of finite index. The runtime of this probabilistic algorithm is analyzed and shown to require polynomial storage and run in expected time $O((\log p)^2 p^{1/2})$.

In 2017, Daghigh et al. [38] proposed three key-exchange protocols using elliptic curve isogenies. The first approach uses an ordinary curve E, the private keys are randomly generated endomorphisms of E, and the public keys are the images of the private endomorphisms on a public point P. A shared secret can be derived by applying one's private endomorphism to the other's public point. The second approach is similar but uses a supersingular elliptic curve. As the endomorphisms no longer commute, the shared secret is instead derived by constructing an isogeny. The third and final method is to have four independent isogenies between the global parameters E and E' (constructed by precomposing four independent endomorphisms of E with an isogeny from E to E'). The private key is then a random isogeny from E to E', the public key is the image of a public point under this isogeny, and the shared secret can be constructed via new isogenies.

We examine all three proposed methods in Chapter 6. The conclusion is that all three are insecure when the order of P is a smooth number. Further, when the order of P is not smooth, the first method is not post-quantum secure, and the second and third methods only have a polynomial gap in the work of a honest participant and a malicious adversary (that is, under the assumption that honest participants can perform their operations in a feasible amount of time, these schemes succumb to classical attacks in a feasible amount of time).

In Chapter 7 we present a generic public-key encryption scheme from the learning with errors problem [80] on finitely generated groups. We then present an instantiation based on supersingular isogenies, analyze the key sizes and computation costs, and give the usual proofs of security. We conclude with cryptanalysis which shows that this scheme is insecure by an application of Shor's quantum hidden subgroup algorithm.

1.1 Organization of Work

Section 2.1 introduces the necessary mathematical background of elliptic curves over finite fields. Section 2.2 presents the background of both ordinary and supersingular isogeny-based cryptography. Chapter 3 studies a new cryptanalytic avenue for supersingular isogeny-based key-exchange and its applicability to SIKE. Chapter 4 proves the security of the novel static-static key-exchange transformation, and supersingular isogeny-based application. Chapter 5 presents a novel algorithm for constructing (power of ℓ degree) endomorphisms of arbitrary supersingular elliptic curves over a finite field. Chapter 6 studies the insecurity of three endomorphism-based schemes. Chapter 7 introduces and breaks the supersingular isogeny instantiation of a generic learning with errors encryption scheme.

Chapter 2

Elliptic Curves and Cryptography

This chapter is an introduction to the theory of elliptic curves and isogenies, and their use in post-quantum cryptography. A goal of this thesis is to be self-contained, so all necessary preliminary material is presented. The material in Section 2.1 of this chapter is collected primarily from the texts [31, 32, 51, 85].

2.1 Elliptic Curve Preliminaries

2.1.1 Coordinates and the Group Law

The central objects of isogeny-based cryptography, and the work of this thesis, are elliptic curves. The definition we will begin with is of an elliptic curve.

Definition 2.1.1. *An **elliptic curve** E over a field K, denoted E/K, is given by a non-singular projective curve of the form*

$$E : Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3, \tag{2.1}$$

*for $a_1, a_2, a_3, a_4, a_6 \in K$, along with a base point $O = O(E) = O_E = \infty = \infty_E = [0, 1, 0]$ which is referred to as the **point at infinity**.*

From the coefficients of an elliptic curve in Eq. 2.1 we can define the b-invariants and the c-invariants of a curve to simplify later algebra:

$$\begin{aligned} b_2 &= a_1^2 + 4a_4, \\ b_4 &= 2a_4 + a_1a_3, \\ b_6 &= a_3^2 + 4a_6, \\ c_4 &= b_2^2 - 24b_4, \\ c_6 &= -b_2^3 + 36b_2b_4 - 216b_6. \end{aligned}$$

Frequently this work will use affine Weierstraß equations for an elliptic curve over K instead of projective coordinates. The (affine) **long Weierstraß form** can be obtained from Eq. 2.1 by the change of coordinates $x = X/Z$ and $y = Y/Z$:

Definition 2.1.2. *An elliptic curve E over a field K is given by the equation*

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \tag{2.2}$$

along with the point at infinity, where the coefficients $a_1, a_2, a_3, a_4, a_6 \in K$ are such that for each point $P = (x_1, y_1)$ with coordinates in $\overline{K}$ satisfying Eq. 2.2, the partial derivatives at P ($2y_1 + a_1x_1 + a_3$ and $3x_1^2 + 2a_2x_1 + a_4 - a_1y_1$) do not vanish simultaneously.

The last condition in Definition 2.1.2 is equivalent to the non-singular condition of Definition 2.1.1. This change of coordinates requires that $Z \neq 0$. Indeed, each point $(x, y) \neq O$ in Weierstraß coordinate, is equivalent to $(x, y, 1)$ in projective coordinates, and $Z = 0$ for the point O.

When the characteristic of the field K is not 2 we can eliminate the xy and y terms by the change of coordinate $y \mapsto \frac{1}{2}(y - a_1x - a_3)$:

$$E : y^2 = 4x^3 + b_2x^2 + 2b_4 + b_6. \tag{2.3}$$

Further, if $char(K) \neq 3$, then we can apply one last substitution, $(x, y) \mapsto \left(\frac{x-3b_2}{36}, \frac{y}{108}\right)$, to obtain the **short Weierstraß form**:

$$E : y^2 = x^3 - 27c_4x - 54c_6. \tag{2.4}$$

The cubic polynomial has only simple roots over $\overline{K}$ if and only if its discriminant is non-zero. Therefore, checking the non-singularity condition (or checking that an equation defines an elliptic curve) can be done by computing the discriminant from the coefficients.

Definition 2.1.3. *Let E be as in Eq. 2.1, and the b-invariants be as above. The* ***discriminant*** *of the curve E is*

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

where $b_8 = a_1^2 a_6 + 4a_2 a_6 - a_1 a_3 a_4 + a_2 a_3^2 - a_4^2$.

Additionally, if E is an elliptic curve in short Weierstraß form, $E : y^2 = x^3 + ax + b$, then:

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

Hence, a short Weierstraß form equation defines an elliptic curve if and only if

$$4a^3 + 27b^2 \neq 0.$$

There is a well known geometrically constructed law that provides elliptic curves with a natural group structure. A solution $(x, y) \in K \times K$ to the defining equation of E is called a K-rational point of E. The set of all K-rational points form an Abelian group, denoted $E(K)$, with the point at infinity as the identity. The group law can be expressed by rational polynomials with coefficients in K, enabling addition and subtraction of elliptic curve points, as well as multiplication of a K-rational point by an integer. This last operation can be done efficiently using the standard double-and-add technique.

2.1.2 Torsion Subgroups

Definition 2.1.4. *Let E be an elliptic curve over the field K, and let $P \in E(\overline{K})$. Let $m \in \mathbb{Z}$, and define the* ***multiplication-by-m map***

$$\begin{aligned}
&[m]: \ E(\overline{K}) \to E(\overline{K}),\\
&[m]P = P + \ldots + P, \quad \textit{if } m \geq 0, \quad \textit{and}\\
&[m]P = (-P) + \ldots + (-P), \quad \textit{if } m < 0,
\end{aligned}$$

where $-P$ is the group inverse of P, and the sums contain m elements.

Example 2.1.5. *The map $[0]$ is defined so that $[0]P = \infty$ for all $P \in E(\overline{K})$. The map $[1]$ is the identity map.*

Definition 2.1.6. *For any $m \in \mathbb{Z}$ and elliptic curve $E(K)$, the subgroup*

$$E[m] := \{P \in E(\overline{K}) : [m]P = \infty\}$$

is called the ***m-torsion subgroup*** *of $E(\overline{K})$. An element $P \in E[m]$ is called an* ***m-torsion point****.*

This subgroup can be viewed as the kernel of $[m]$, and points in $E[m]$ all have order dividing m.

Theorem 2.1.7. [*98*, *Theorem* 3.2] *Let E be an elliptic curve defined over K. If char(K) is 0 or coprime to m, then*

$$E[m] \cong \mathbb{Z}/m\mathbb{Z} \oplus \mathbb{Z}/m\mathbb{Z}.$$

The case where the characteristic of K is some prime p and $m = p^r$ is covered in Section 2.1.4.

2.1.3 Maps Between Curves

In this section we discuss the algebraic relationships between elliptic curves.

Morphisms

The material in this section is well known, but two good sources are [51], and [88]. We start with the most elementary maps of elliptic curves.

Definition 2.1.8. *Let E and E' be two elliptic curves defined over K. A* ***morphism*** *$\phi : E(\overline{K}) \to E'(\overline{K})$ over K is a polynomial mapping with coefficients from K. If the curves are in projective coordinates we can write*

$$\phi(X : Y : Z) = (\phi_0(X, Y, Z) : \phi_1(X, Y, Z) : \phi_2(X, Y, Z)),$$

where ϕ_0, ϕ_1, ϕ_2 are homogeneous polynomials of equal degree satisfying the defining equation of E'. Alternatively, in Weierstraß coordinates, a morphism ϕ is a rational map

$$\phi(x, y) = \left(\frac{\phi_0(x, y, 1)}{\phi_2(x, y, 1)}, \frac{\phi_1(x, y, 1)}{\phi_2(x, y, 1)}\right).$$

A morphism defined over field K is commonly referred to as a K-rational morphism. Each morphism has an integer degree which will be defined later (see Definition 2.1.21), but for now note that a degree m morphism from E to E' typically implies the kernel of the morphism has cardinality m, that is, the morphism is m-to-1 from $E(\overline{K})$ to $E'(\overline{K})$.

One family of morphisms is the **translation morphisms**. For each point $P \in E$, we can define

$$\tau_P : E \to E, \ \tau_P(Q) = Q + P.$$

These are morphisms because the elliptic curve group law is defined by rational polynomials.

Definition 2.1.9. *A **homomorphism** ϕ is a morphism of elliptic curves such that*

$$\phi(P+Q) = \phi(P) + \phi(Q),$$

for all $P, Q \in E(\overline{K})$. That is, ϕ respects the group structure of the curve.

Proposition 2.1.10. *Every morphism from $E \to E'$ that maps $O(E)$ to $O(E')$ is a homomorphism.*

A translation morphism τ_P is not a homomorphism unless $P = O$, in which case it is the trivial homomorphism between E and itself.

Definition 2.1.11. *A K**-isomorphism** between elliptic curves is a group isomorphism defined over $\overline{K}$.*

A useful property of elliptic curves in Weierstraß form is that all isomorphisms between them have been classified.

Proposition 2.1.12. [85, III.1] *Elliptic curves $E/K : y^2 = x^3 + ax + b$ and $E'/K : y^2 = x^3 + a'x + b'$ are isomorphic over $\overline{K}$ if and only if there exists $\mu \in \overline{K}^*$ such that*

$$\begin{aligned} a' &= \mu^2 a, \\ b' &= \mu^3 b. \end{aligned}$$

If so, the isomorphism $E \to E'$ is given by $(x, y) \mapsto (\mu x, \mu^{\frac{3}{2}} y)$.

A special case of Proposition 2.1.12 is when μ is a quadratic non-residue in K. In this case the essential element $\mu^{\frac{3}{2}}$ is undefined in K, and so the isomorphism is defined over $K(\sqrt{\mu})$ instead.

Definition 2.1.13. *Let $E/K : y^2 = x^3 + ax + b$ be an elliptic curve and char$(K) \neq 2$. Then for any quadratic non-residue $\mu \in K \backslash \{O\}$ we define the elliptic curve $E^{(\mu)} : y^2 = x^3 + \mu^2 ax + \mu^3 b$ to be the* ***quadratic twist*** *of E by μ.*

One can verify by this definition that the twist of an elliptic curve will give a non-singular equation.

We can now define the isomorphism class of elliptic curves defined over a given field. Further, there exists a unique quantity for each such class that we can use as a label.

Definition 2.1.14. *For an elliptic curve $E : y^2 = x^3 + ax + b$ defined over a field K, where the characteristic of K is not 2 or 3, define*

$$j(E) = \frac{c_4^3}{\Delta} = 1728\frac{c_4^3}{c_4^3 - c_6^2} = 1728\frac{4a^3}{4a^3 + 27b^2} \in K$$

to be the ***j-invariant*** *of E.*

It is simple to check, using Proposition 2.1.12 that this quantity is invariant for a K-isomorphism class of elliptic curves. As well, the j-invariant is unique to a K-isomorphism class when K is an algebraically closed field.

Theorem 2.1.15. [85, III.1.4] *Two elliptic curves are isomorphic over $\overline{K}$ if and only if they have the same j-invariant.*

The construction of an elliptic curve for a given j-invariant is also known:

Theorem 2.1.16. [85, III.1.4] *Given some $j \in K$, the following formulas determine an elliptic curve defined over K whose j-invariant is equal to j:*

(1) If $j = 0$, then $E : y^2 + y = x^3$,

(2) If $j = 1728$, then $E : y^2 = x^3 + x$,

(3) Otherwise, $E : y^2 + xy = x^3 - \frac{36}{j-1728}x - \frac{1}{j-1728}$.

Observe that neither (1) nor (3) yield equations in short Weierstraß form, but as stated earlier, when the characteristic of K is neither 2 nor 3 we can rewrite these curves as $y^2 = x^3 - 27c_4 x - 54c_6$ using the c-invariants c_4 and c_6.

It is worth noting that when $j(E) = 0$ or 1728 there exist twisted elliptic curves other than quadratic twists. When $j(E) = 0$, one can twist by a cubic character μ and the twisted curve is of the form $y^2 = x^3 + \mu b$. When $j(E) = 1728$ quartic twists are possible and of the form $y^2 = x^3 + \mu ax$ [85, Prop 5.4]. In these cases, the isomorphism will be defined over the appropriate cubic or quartic extension of K.

Isogenies

Definition 2.1.17. *Let K be a field and E/K, E'/K be two elliptic curves. If F is an extension of K (possibly $\overline{K}$ or K itself), define an **isogeny over F**, or **F-isogeny**, between E and E' to be a non-zero morphism*

$$\phi : E(\overline{K}) \to E'(\overline{K})$$

*mapping $O(E)$ to $O(E')$, with coefficients from F. Two elliptic curves are defined to be **isogenous** if and only if there is an isogeny ϕ between them.*

Theorem 2.1.10 immediately shows that every isogeny is necessarily a homomorphism between $E(K)$ and $E'(K)$.

Since isomorphisms are isogenies between isomorphic curves, we can use them to define the notion of an isomorphism of isogenies.

Definition 2.1.18. *Isogenies $\phi_1 : E \to E'$ and $\phi_2 : E \to E''$ are said to be **isomorphic isogenies** if there exists an isomorphism $\psi : E' \to E''$ such that $\phi_2 = \psi \circ \phi_1$.*

Definition 2.1.19. *The kernel of an isogeny ϕ is*

$$\ker(\phi) := \{P \mid P \in E(\overline{K}) \textit{ and } \phi(P) = O\}.$$

*If $\ker \phi$ is cyclic, then ϕ is called **cyclic**.*

Recall, the form of an isogeny (2.1.8) between elliptic curves in Weierstraß form:

$$\phi(x, y) = \left(\frac{\phi_0(x, y, 1)}{\phi_2(x, y, 1)}, \frac{\phi_1(x, y, 1)}{\phi_2(x, y, 1)} \right)$$

such that ϕ fixes the identity. From this it is clear that the kernel of ϕ will be a subset of the set of zeros of ϕ_2 (or exactly the set of zeros of ϕ_2 if there is no cancellation to be made).

Definition 2.1.20. *The **coordinate ring** of an elliptic curve E over field K is*

$$K[E] := K[x, y]/\langle y^2 + a_1xy + a_3y - x^3 - a_2x^2 - a_4x - a_6\rangle.$$

*The **function field** of E over K is the field of fractions of $K[E]$, denoted $K(E)$.*

An early observation is that all morphisms between elliptic curves over K are either constant or surjective. We use this result to construct an injective homomorphism of function fields from each isogeny. Let $\phi : E \to E'$ be an isogeny, and define $\phi^* : \overline{K}(E') \to \overline{K}(E)$ by $\phi^*(f) = f \circ \phi$.

Definition 2.1.21. *The **degree** of a morphism $\phi : E \to E'$ between elliptic curves is*

$$\deg(\phi) := [\overline{K}(E) : \phi^*(\overline{K}(E'))],$$

the degree of the function field extension induced by ϕ. A degree ℓ isogeny is referred to as an ℓ-isogeny.

Note that F-isogeny and ℓ-isogeny is an abuse of notation, but the prefix being a field or an integer should be clear in each context.

Definition 2.1.22. *An **endomorphism** of an elliptic curve E/K is a homomorphism from E/K to itself. The set of all endomorphisms is denoted* $\mathrm{End}_K(E)$ *or just* $\mathrm{End}(E)$.

The set $\mathrm{End}(E)$ is a ring due to the group structure of E. Here we note that for every $m \in \mathbb{Z}$, the multiplication-by-m map is an endomorphism, and the degree of $[m]$ is m^2. Therefore, for an elliptic curve E, $\mathrm{End}(E)$ will always contain a subring isomorphic to $\mathbb{Z}$. The kernel of the map $[m] \in \mathrm{End}(E)$ are all the points whose order divides m, and so the denominator of the rational map of $[m]$ must be zero at the x-coordinates of $E[m]$. There is a family of polynomials, called the m^{th}-**division polynomial** and denoted by $\psi_m \in \mathbb{Z}[x, y, a_1, \ldots, a_6]$, indexed by m, whose zeroes are the x-coordinates of $E[m]$. The division polynomials can be computed recursively using the initial values and equations:

$$\begin{aligned}
\psi_1 = 1,\ \psi_2 &= 2y + a_1x + a_3,\ \psi_3 = 3x^4 + b_2x^3 + 3b_4x^2 + 3b_6x + b_8,\\
\psi_4 &= \psi_2\left(2x^6 + b_2x^5 + 5b_4x^4 + 10b_6x^3 + 10b_8x^2 + (b_2b_8 - b_4b_6)x + (b_4b_8 - b_6^2)\right),\\
\psi_{2m+1} &= \psi_{m+2}\psi_m^3 - \psi_{m-1}\psi_{m+1}^3,\ m \geq 2\\
\psi_2\psi_{2m} &= \psi_{m-1}^2\psi_m\psi_{m+2} - \psi_{m-2}\psi_m\psi_{m+1}^2,\ m \geq 3.
\end{aligned}$$

As endomorphisms are group homomorphisms, they act as automorphisms of torsion subgroups $E[n]$ when n is coprime to their degree. By applying Theorem 2.1.7, if n is also coprime to the field characteristic, one can view the action of an endomorphism $\phi \in \mathrm{End}(E)$ on $E[n]$ as a 2×2 matrix over $\mathbb{Z}/n\mathbb{Z}$. When a basis $P, Q \in E[n]$ is provided, we denote this matrix by $\phi|_{\langle P,Q \rangle}$, or when the basis is implied simply by $\phi|_{E[n]}$.

One of the most important morphisms in the study of elliptic curves over finite fields is the Frobenius endomorphism.

Example 2.1.23. *Let $E/\mathbb{F}_{p^n}$ be an elliptic curve. Define the p^n-th* ***Frobenius endomorphism*** *$\pi_E : E(\mathbb{F}_{p^n}) \to E(\mathbb{F}_{p^n})$ by $(x, y) \mapsto (x^{p^n}, y^{p^n})$. The notation π_{p^n} is also used.*

It can be shown that if K is finite, then π_E is not equal to $[m]$ for any integer m. This implies that $\mathrm{End}(E/K)$ will always contain a *strict* subring isomorphic to $\mathbb{Z}$ when K is finite.

Example 2.1.24. *An elliptic curve isomorphism is an isogeny of degree 1. The p^n-th Frobenius endomorphism is an isogeny of degree p^n.*

For any integer $\ell > 0$ with $p \nmid \ell$, the **Weil pairing** is a bilinear form that we denote by

$$e_\ell : E[\ell] \times E[\ell] \to \mu_\ell,$$

where $\mu_\ell = \{x \in \mathbb{F}_{p^n} | x^\ell = 1\}$. See [87, §III.8] for an in-depth discussion of the Weil pairing. The following proposition connecting the Weil pairing and isogenies follows immediately from [87, III.8.2].

Proposition 2.1.25. *Let E be an elliptic curve and $R, S \in E[\ell]$ for some positive integer ℓ. If $\phi : E \to E'$ is an isogeny, then*

$$e_\ell(\phi(R), \phi(S)) = e_\ell(R, S)^{deg(\phi)}.$$

We now return to the preliminaries of isogenies by introducing dual isogenies.

Proposition 2.1.26. [85*, III.6*] *Let $\phi : E_1 \to E_2$ be an isogeny. There exists a unique isogeny $\hat{\phi} : E_2 \to E_1$ such that $\deg(\phi) = \deg(\hat{\phi})$, and the composition of these two isogenies is the multiplication-by-deg(ϕ) map. That is, $\phi \circ \hat{\phi} = [\deg(\phi)]$ and $\hat{\phi} \circ \phi = [\deg(\phi)]$. This map is defined as the* ***dual isogeny*** *of ϕ.*

The following equation is useful for the dual of a composition of isogenies, $\phi_1 : E_1 \to E_2$, $\phi_2 : E_2 \to E_3$:

$$\widehat{\phi_2 \circ \phi_1} = \widehat{\phi_1} \circ \widehat{\phi_2}.$$

In isogeny-based cryptography, we are interested in separable isogenies over finite fields.

Definition 2.1.27. *Let $\phi : E/K \to E'/K$ be an isogeny between elliptic curves. If $\overline{K}(E)$ is a separable (inseparable, purely inseparable, resp.) field extension of $\phi^*(\overline{K}(E'))$, then we say ϕ is* ***separable*** *(inseparable, purely inseparable, resp.).*

Theorem 2.1.28. [85, III.5.5] *Let E be an elliptic curve defined over a field $\mathbb{F}_q$ of characteristic p. The isogeny $m + n\pi_E$, for $m, n \in \mathbb{Z}$, is separable if and only if $p \nmid m$.*

The following result shows where our early "definition" of degree came from:

Proposition 2.1.29. [85, II.2.6] *Let ϕ be a separable isogeny. Then,* $\deg(\phi) = |\ker(\phi)|$.

For elliptic curves defined over a finite field, Tate has shown that being isogenous is equivalent to having the same cardinality.

Theorem 2.1.30. [92] *(Tate's Isogeny Theorem) Let E and E' be elliptic curves defined over some finite field $\mathbb{F}_{p^n}$. Then E and E' are isogenous over $\mathbb{F}_{p^n}$ if and only if*

$$|E(\mathbb{F}_{p^n})| = |E'(\mathbb{F}_{p^n})|.$$

This leads to the main isogeny theorem that we will need.

Theorem 2.1.31. [51, 9.6.19] *Let E be an elliptic curve over K. Let $G \subset E(\overline{K})$ be a finite subgroup that is defined over K (i.e., $\sigma(P) \in G$ for all $P \in G$ and $\sigma \in \mathrm{Gal}(\overline{K}/K)$). Then there is a unique (up to isomorphism) elliptic curve E' over K, and a unique (up to isomorphism) isogeny $\phi : E \to E'$ over K such that* $\ker(\phi) = G$.

The unique elliptic curve E' from Theorem 2.1.31 is denoted by E/G, or $\phi(E)$. For a fixed integer ℓ, the ℓ^{th}-**modular polynomial** is a bivariate integral polynomial $\psi_\ell(x, y)$ such that the roots of $\psi_\ell(j(E), y)$ are exactly the j-invariants of the elliptic curves which are ℓ-isogenous to E.

The standard way of computing E', ϕ, or $\phi(P)$ for some $P \in E(\overline{K})$ is to use Vélu's formulas [95], which involves calculating a summation over all the elements of that subgroup $G = \ker(\phi)$ (see Section 2.1.6). The security of isogeny-based cryptography depends on the cardinality of these kernels, so the subgroup G must be large which makes Vélu's formulas impractical. However, as we will see in Section 2.2.2 the cardinality of G will be chosen to be a power of a small prime (2 or 3) and so we can apply Vélu's formulas to compute isogenies efficiently in such cases, as explained and optimized in [47].

2.1.4 Supersingularity

Let E/K be an elliptic curve defined over a field of characteristic p. A group of interest is the group of the p-torsion points, $E[p]$. In fact, the structure of p-torsion points directly determines the type of endomorphism ring of E/K.

Definition 2.1.32. *Let B be a $\mathbb{Q}$-algebra that is finitely generated over $\mathbb{Q}$ (B can be non-Abelian). An **order** R of B is a subring of B that is finitely generated as a $\mathbb{Z}$-module and satisfies $R \otimes_{\mathbb{Z}} \mathbb{Q} = B$.*

Theorem 2.1.33. [85, V.3.1] *Let K be a field of prime characteristic p, and let E/K be an elliptic curve. For each integer $r \geq 1$ let*

$$\pi_r : E \to E^{(p^r)}$$

be the p^r-th Frobenius map.

The following are equivalent:

(i) $E[p^r] = \{O\}$ for all $r \geq 1$.

(ii) The map $[p] : E \to E$ is purely inseparable and $j(E) \in \mathbb{F}_{p^2}$.

(iii) $End(E)$ is an order in a quaternion algebra.

(iv) $\hat{\pi}_r$ is purely inseparable for all $r \geq 1$.

If the equivalent conditions do not hold, then

$$E[p^r] = \mathbb{Z}/p^r\mathbb{Z}, \text{ for all } r \geq 1,$$

and $End(E)$ is an order in an imaginary quadratic field extension of $\mathbb{Q}$.

If the above conditions hold, then we say the curve E is **supersingular**, otherwise we say E is **ordinary**. The term supersingular is unrelated to the notion of singular curves, and instead refers to how elliptic curves with these endomorphism rings are uncommon. Part (ii) of Theorem 2.1.33 states that all supersingular elliptic curves can be defined over $\mathbb{F}_{p^2}$, and this will be useful in the cryptanalysis of supersingular isogeny cryptography (Section 2.2.2).

A direct consequence of Theorem 2.1.33 (i) is that isogenies preserve the type of elliptic curve and so we can discuss supersingular elliptic curve isogenies and ordinary elliptic curve isogenies.

Theorem 2.1.34. *Let $\phi : E_1 \to E_2$ be an isogeny. E_1 is supersingular if and only if E_2 is supersingular. E_1 is ordinary if and only if E_2 is ordinary.*

Since the elliptic curves discussed in this work are primarily supersingular it is worth counting the number of supersingular elliptic curves over $\overline{K}$.

Theorem 2.1.35. *[85, V.4.1(d)] Let K be a finite field of characteristic $p \geq 3$. There is one supersingular elliptic curve in characterstic 3, and for $p \geq 5$ the number of supersingular elliptic curves up to $\overline{K}$-isomorphism is*

$$\left\lfloor \frac{p}{12} \right\rfloor + \begin{cases} 0, & p \equiv 1 \mod 12, \\ 1, & p \equiv 5 \mod 12, \\ 1, & p \equiv 7 \mod 12, \\ 2, & p \equiv 11 \mod 12. \end{cases}$$

2.1.5 Hilbert Class Field Theory

The following section requires knowledge of Hilbert class fields and their general theory, and so a brief introduction is provided here (the sources of this subsection are [30] and [85]).

Let L be a number field. The set of algebraic integers in L form a ring, denoted $\mathcal{O}_L$, called the *ring of integers* of L.

Theorem 2.1.36. [30, *Theorem* 4.4.2] *The ring $\mathcal{O}_L$ is a free $\mathbb{Z}$-module of rank $[L : \mathbb{Q}]$.*

Definition 2.1.37. *An **integral ideal** is a $\mathbb{Z}$-submodule $\mathfrak{a} \subset \mathcal{O}_L$ such that for every $\alpha \in \mathcal{O}_L$ and $a \in \mathfrak{a}$ we have $\alpha a \in \mathfrak{a}$.*

Definition 2.1.38. *A **fractional ideal** $\mathfrak{a} \subset L$ is a non-zero submodule of L such that there exists a non-zero integer α with $\alpha\mathfrak{a}$ an integral ideal of $\mathcal{O}_L$.*

We would like to have a group structure for fractional ideals. The full ring of integers $\mathcal{O}_L$ will serve as the identity. We define the product of fractional ideals $\mathfrak{a}, \mathfrak{b}$ as:

$$\mathfrak{a}\mathfrak{b} := \left\{\textstyle\sum ij \mid i \in \mathfrak{a}, j \in \mathfrak{b}\right\}.$$

Note this operation is Abelian. We make use of the following theorem to define inverses:

Theorem 2.1.39. [30, *Theorem* 4.6.14] *If $\mathfrak{a}$ is a fractional ideal of $\mathcal{O}_L$ and if we set*

$$\mathfrak{a}^{-1} := \{\alpha \in L \mid \alpha\mathfrak{a} \subset \mathcal{O}_L\},$$

then $\mathfrak{a}\mathfrak{a}^{-1} = \mathfrak{a}^{-1}\mathfrak{a} = \mathcal{O}_L$.

Lastly, we say that two fractional ideals are equivalent if they differ by a non-zero element of L. Now we can define the **class group** of $\mathcal{O}_L$ to be the finite group of equivalence classes of fractional ideals with the above operation. We denote this group by $\mathcal{CL}(\mathcal{O}_L) = \mathcal{CL}(\mathcal{O}_L)$ and define the **class number** as $h(\mathcal{O}_L) := |\mathcal{CL}(\mathcal{O}_L)|$.

Since integral ideals of $\mathcal{O}_L$ are modules of maximal rank [30, Theorem 4.6.3], the quotient $\mathcal{O}_L/\mathfrak{a}$ is a finite ring. In the case where this quotient ring is an integral domain we say that $\mathfrak{a}$ is a **prime ideal** of $\mathcal{O}_L$.

Definition 2.1.40. *Let p be a prime number and $\mathfrak{p}$ a prime ideal of $\mathcal{O}_L$. Then $\mathfrak{p}$ is said to be a prime ideal* ***above*** *p if $\mathfrak{p} \cap \mathbb{Z} = p\mathbb{Z}$.*

Theorem 2.1.41. [*30*, *Theorem* 4.8.3] *Let p be a prime number. There exist positive integers e_i such that*

$$p\mathcal{O}_L = \prod_i \mathfrak{p}_i^{e_i},$$

where the $\mathfrak{p}_i$ are all the prime ideals above p.

Definition 2.1.42. *Depending on the structure of the product in Theorem* 2.2.8 *we give the prime number p different names. If $p\mathcal{O}_L = \mathfrak{p}$, then p is said to be* ***inert****. If $p\mathcal{O}_L = \prod_{i=1}^{n} \mathfrak{p}_i$, where $n = [L : \mathbb{Q}]$ and all the $\mathfrak{p}_i$'s are different, then p said to* ***split completely****. If $e_i \geq 2$ for some i, then p is* ***ramified****.*

We end this section with another look at the Frobenius endomorphism. For elliptic curves defined over $\mathbb{Q}$, the m-torsion points have coordinates in $\overline{\mathbb{Q}}$. Each element of the Galois group $\mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ fixes the m-torsion points of E, $\forall m \in \mathbb{Z}$. By Theorem 2.1.7, since $char(\mathbb{Q}) = 0$, we know that $E(\mathbb{Q})[m] \cong \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$ and so the group of homomorphisms from $E[m]$ to itself is $\mathrm{GL}_2(\mathbb{Z}/m\mathbb{Z})$. We define the map

$$R_{E,m} : \mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q}) \to \mathrm{GL}_2(\mathbb{Z}/m\mathbb{Z})$$

as the **mod m representation attached to** E. While the particular matrix associated with $R_{E,m}(\sigma)$ depends on the choice of basis for $E[m]$, its determinant and trace are invariants.

Let p be prime and $\mathfrak{p}$ be a prime ideal above p. There are **Frobenius elements** $\sigma_p \in \mathrm{Gal}(\overline{\mathbb{Q}}/\mathbb{Q})$ defined by the property that

$$\sigma_p(\alpha) \equiv \alpha^p \bmod \mathfrak{p}$$

for all $\alpha \in \overline{\mathbb{Q}}$. Evaluating $R_{E,m}$ at σ_p gives a matrix in $\mathrm{GL}_2(\mathbb{Z}/m\mathbb{Z})$. We conclude with the following theorem.

Theorem 2.1.43. *For all $m \geq 1$, $\operatorname{Trace}(R_{E,m}(\sigma_p)) \equiv t \bmod m$, where t is the trace of the Frobenius endomorphism.*

2.1.6 Isogeny Computations

This section describes a few algorithms that are commonly used within isogeny-based cryptography.

Vélu's Formula

In 1971, Jacques Vélu provided explicit formulas to compute isogenies in time proportional to half the cardinality of its kernel [95]. Later work [47] optimized this result for the case when the cardinality of the kernel is a power of a small prime number. Below are the formulas and algorithms for these computations.

Using the notation in Vélu's paper: let $E : y^2 = x^3 + ax + b$ be an elliptic curve defined over an algebraically closed field K, let F be a finite subgroup of E, and let $f : E \to E'$ be the isogeny with kernel F.

For a point $P = (x_P, y_P) \neq O$ on E, define the following quantities:

$$\begin{aligned} g_P^x &= 3x_P^2 + a, \\ g_P^y &= -2y_P, \\ u_P &= (g_P^y)^2, \\ v_P &= \begin{cases} g_P^x, & [2]P = \infty, \\ 2g_P^x, & \text{otherwise.} \end{cases} \end{aligned}$$

In order to distinguish points in F from their inverse we define F_2 to be the non-trivial points of order 2 in F, and we define R to be a subset of $(F \backslash \{O\}) \backslash F_2$ such that

$$R \cap (-R) = \emptyset \text{ and } (F \backslash \{O\}) \backslash F_2 = R \cup (-R).$$

Setting $S = F_2 \cup R$, we can define

$$f(x, y) = \left(x + \sum_{P \in S} \frac{v_P}{x - x_P} - \frac{u_P}{(x - x_P)^2},\ y - \sum_{P \in S} u_P \frac{2y}{(x - x_P)^3} + v_P \frac{y - y_P - g_P^x g_P^y}{(x - x_P)^2} \right),$$

for points $(x, y) \notin F$, and $f(x, y) = O$ when $(x, y) \in F$. Further, the equation for the image curve is given by

$$E' : y^2 = x^3 + \left(a - 5 \sum_{P \in S} v_P\right) x + \left(b - 7 \sum_{P \in S} u_P + x_P v_P\right).$$

From these formulas it is plain to see that this computation is inefficient for cryptographically secure kernel sizes, requiring $\tilde{O}(|F|)$ operations in K. Suppose that $|\ker(f)| = \ell^n$ for prime ℓ and $n \geq 1$. De Feo and Jao [47] show how we can compute f in $O(n\ell)$ time instead of the runtime $O(\ell^n)$ from Vélu's formula.

Let Q_1 be a point in F with order ℓ. Applying Vélu's formula to the subgroup $\langle Q_1 \rangle$ of E will give an isogeny

$$f_1 : E \to E/\langle Q_1 \rangle.$$

Then the image of F under f_1 will have size ℓ^{n-1}. If we then find a point Q_2 in $f_1(F) \subset E/\langle Q_1 \rangle$ with order ℓ, we can perform this procedure for a second time to determine

$$f_2 : E/\langle Q1 \rangle \to (E/\langle Q_1 \rangle)/\langle Q_2 \rangle,$$

and codomain. Iterating this n times gives n isogenies, $f_1, f_2, \ldots, f_n$ whose composition is the desired isogeny f with kernel F. The codomain of f is the elliptic curve $E' = (\ldots((E/\langle Q_1 \rangle)/\langle Q_2 \rangle)\ldots)/\langle Q_n \rangle$. Additionally, see [47, §4.2] for optimal implementation details.

Recent work by Bernstein et al. [9] gave an asymptotic improvement on implementation of Vélu's formula, achieving $\tilde{O}(\sqrt{|F|})$ operations in K. If $|F|$ is prime, then the runtime is more efficient than regular Vélu's formula around $|F| \approx 100$.

Bröker's Algorithm

For the purposes of isogeny-based cryptography, we need to be able to efficiently find a supersingular elliptic curve over $\mathbb{F}_{p^n}$ with trace t of the p^n-Frobenius endomorphism, for a given p^n and t. In 2009, Reinier Bröker solved this computational problem [19] and this section is based on that work. The existence of such a curve is guaranteed by the following result of Waterhouse:

Theorem 2.1.44. [99, 4.1] *There exists a supersingular elliptic curve E over $\mathbb{F}_{p^n}$ with trace t of the p^n-Frobenius endomorphism π_E if and only if one of the following holds:*

(a) if n is even and one of the following is true:

(i) $t = \pm 2\sqrt{p^n}$,

(ii) $t = \pm\sqrt{p^n}$ *and* $p \not\equiv 1$ *mod* 3,

(iii) $t = 0$ *and* $p \not\equiv 1$ *mod* 4;

(b) if n is odd and one of the following is true:

(i) $t = 0$,

(ii) $t = \pm\sqrt{2p^n}$ *and* $p = 2$,

(iii) $t = \pm\sqrt{3p^n}$ *and* $p = 3$.

The cases which are relevant in the context of supersingular isogeny-based cryptography are when $t = 0$. The first step is to construct a supersingular curve over $\mathbb{F}_p$ as a reduction of a curve in characteristic 0 using the following result of Deuring.

Theorem 2.1.45. [67, 13.12] *Let E be an elliptic curve defined over a number field L whose endomorphism ring is the maximal order $\mathcal{O}_K$ in an imaginary quadratic field K. Let $\mathfrak{p}$ be a prime ideal of L, and let p be a prime number such that $p \nmid \Delta(E)$ and $\mathfrak{p}$ is above p. Then $E/(L/\langle\mathfrak{p}\rangle)$ is supersingular if and only if p does not split in K.*

From the theory of complex multiplication we have that the j-invariant of E generates the Hilbert class field H of K. That is,

$$H = K[x]/\langle P_K\rangle = K(j(E)),$$

where P_K is the minimal polynomial of $j(E)$ over $\mathbb{Q}$. The polynomial P_K can be explicitly computed efficiently [20] and its degree is equal to the Hilbert class number h_K. If p is inert in $\mathcal{O}_K$, then the roots of $P_K \in \overline{\mathbb{F}}_p[x]$ are supersingular j-invariants. As $j(E) \in \mathbb{F}_{p^2}$ [85, V.3.1], the polynomial P_K splits over $\mathbb{F}_{p^2}$. The following lemma due to Bröker gives a condition for the degree h_K of P_K to be odd, and since each of the factors of P_K have degree 1 or 2, this condition is sufficient for $P_K \in \mathbb{F}_p[x]$ to have a root in $\mathbb{F}_p$.

Lemma 2.1.46. [19, 2.3] *Let K be an imaginary quadratic field with odd class number h_K. Then, $K = \mathbb{Q}(i)$, or $K = \mathbb{Q}(\sqrt{-2})$, or $K = \mathbb{Q}(\sqrt{-q})$ with q prime and congruent to 3 modulo 4.*

Combining these results, Bröker gives an algorithm for constructing a supersingular elliptic curve over $\mathbb{F}_p$.

Algorithm 2.1.47.
Input: *a prime number* p.

Output: *a supersingular elliptic curve over* $\mathbb{F}_p$.

1. *If* $p = 2$, *return* $y^2 + y = x^3$.
2. *If* $p \equiv 3 \bmod 4$, *return* $y^2 = x^3 - x$.
3. *Let* $q \equiv 3 \bmod 4$ *be the smallest prime with* $-q$ *a non-quadratic residue mod* p.
4. *Compute* $P_K \in \mathbb{Z}[x]$ *for* $K = \mathbb{Q}(\sqrt{-q})$.
5. *Compute a root* $j \in \mathbb{F}_p$ *of* $P_K \in \mathbb{F}_p[x]$.
6. *If* $q = 3$, *return* $y^2 = x^3 - 1$. *Otherwise, set* $a \leftarrow \frac{27j}{4(1728-j)} \in \mathbb{F}_p$ *and return* $y^2 = x^3 + ax - a$.

Let $q = p^n$ be a prime power and let t be a trace of the Frobenius endomorphism of the form described in Theorem 2.1.44. From Algorithm 2.1.47, we compute a supersingular elliptic curve E over $\mathbb{F}_p$. Let $E'/\mathbb{F}_q$ be E defined over $\mathbb{F}_q$, and let t' be the trace of the Frobenius endomorphism of $E'(\mathbb{F}_q)$.

Lemma 2.1.48. [19, 3.1] *If* n *is odd, then* $t' = 0$. *If* $n \equiv 0 \bmod 4$, *then* $t' = 2\sqrt{q}$. *Otherwise,* $n \equiv 2 \bmod 4$, *and* $t' = -2\sqrt{q}$.

If $p \not\equiv 1 \bmod 4$, then a twist by a primitive fourth root of unity $i \in \mathbb{F}_q$ will give curves with Frobenius trace $\pm 2\sqrt{q}$ and 0. If $p \not\equiv 1 \bmod 3$, then a twist by a primitive sixth root of unity $\zeta_6 \in \mathbb{F}_q$ will give curves with Frobenius trace $\pm 2\sqrt{q}$, and $\pm\sqrt{q}$. If $p \equiv 1 \bmod 12$, then a twist by -1 suffices. By Theorem 2.1.44 we know that these are the only three cases.

2.2 Isogeny-Based Cryptography

This section will cover the two types of isogeny-based cryptography, the computational problems their security depend on, and their susceptibility to known attacks. Recall the definitions for ordinary and supersingular curves in Section 2.1.4 are based on the p-torsion points of the curve when defined over a finite field of characteristic p.

2.2.1 Ordinary Elliptic Curve Cryptography

The first public-key cryptosystems based on the intractability of constructing an isogeny between two known elliptic curves are due to Couveignes [36] and Stolbunov [89] independently, using ordinary curves. Couveignes introduced the notion of a Hard Homogeneous Space (HHS) to generalize the discrete logarithm problem and showed how it can be used for key exchange and authentication schemes.

Definition 2.2.1. *Let G be a finite, Abelian group. Then a **homogeneous space** H for G is a set that is acted on by G such that $|H| = |G|$ and the action is simply transitive (for all $h_0, h_1 \in H$ there exists a unique $g \in G$ such that $g \cdot h_0 = h_1$). For $h_1, h_2 \in H$ denote the unique element $g \in G$ with $g \cdot h_1 = h_2$ by $\delta(h_1, h_2)$.*

This definition alone is not enough to produce a cryptographic scheme. As in the setting of the discrete logarithm problem over a finite field, we require that the basic group operations are efficiently computable while the inverse of the action is not.

Definition 2.2.2. [36] *Let H be a homogeneous space for G, and suppose the elements of G and H are represented by strings (not necessarily uniquely).*

Suppose the following computations are efficient:

(i) the group operation of G,

(ii) inverting an element of G,

(iii) testing membership in G and H,

(iv) testing equality in G and in H,

(iv) finding a random element in G with uniform probability,

(v) computing $g \cdot h$ for all $g \in G$ and $h \in H$.

Further suppose these two problems are computationally difficult:

(vi) given $h_1, h_2 \in H$ compute $\delta(h_1, h_2)$,

(vii) given $h_1, h_2, h_3 \in H$ compute the unique $h_4 \in H$ with $\delta(h_1, h_2) = \delta(h_3, h_4)$.

*Then we say H is a **hard homogeneous space**.*

This basic setup allows us to create cryptosystems based on the action of isogenies on ordinary elliptic curves defined over finite fields. Let E be an ordinary elliptic curve over $\mathbb{F}_{p^n}$. Recall from Section 2.1.4 that $\mathcal{O} := \mathrm{End}(E)$ is an order in an imaginary quadratic field extension over $\mathbb{Q}$, say $K = \mathbb{Q}(\sqrt{\Delta(E)}) = \mathcal{O} \otimes_{\mathbb{Z}} \mathbb{Q}$. If we assume that the discriminant $\Delta(E)$ is square-free, then $\mathcal{O}$ is the maximal order, $\mathcal{O}_K$, in K. From the theory of complex multiplication [86, II.1.5], the ideal class group of $\mathcal{O}_K$ induces a simply transitive action on the set of elliptic curves isogenous to E.

Definition 2.2.3. [*86, Chapter 2 - Section 1*] *Define $\mathcal{ELL}(\mathcal{O})$ to be the quotient space $\{E/\mathbb{C}$ with $\mathcal{O} \cong \mathrm{End}(E)\}/\{$isomorphisms over $\mathbb{C}\}$.*

By Theorem 2.1.15, we can associate each element of $\mathcal{ELL}(\mathcal{O})$ with a j value. Couveignes and Stolbunov independently observed that if we set

$$G = \mathcal{CL}(\mathcal{O}_K) = \text{ ideal class group of } \mathcal{O}_K, \text{ and } H = \mathcal{ELL}(\mathcal{O}_K),$$

then there is an action

$$* : \mathcal{CL}(\mathcal{O}_K) \times \mathcal{ELL}(\mathcal{O}_K) \to \mathcal{ELL}(\mathcal{O}_K)$$

that satisfies the conditions of the hard homogeneous space definition. Some of the necessary material to confirm this as true has been given in Theorem 2.1.15, Section 2.1.3, and Section 2.1.5, however [86, Chapter 2 - Section 1] is the recommended source for details (specifically Proposition 1.2). Lercier and Morain [69] is an early source for computing the action required in condition (v) of Definition 2.2.2 while [17] details a more efficient computation.

Figure 2.2.1 details the ordinary isogeny key exchange protocol, where the public parameters are $\mathcal{CL}(\mathcal{O}_K)$, $\mathcal{ELL}(\mathcal{O}_K)$, and $x \in \mathcal{ELL}(\mathcal{O}_K)$.

In 2018, the authors of [23] propose a new key establishment protocol using supersingular elliptic curves over a field $\mathbb{F}_p$. While not technically ordinary elliptic curve cryptography, the endomorphism rings $\mathrm{End}_{\mathbb{F}_p}(E)$ over this base field are isomorphic to imaginary quadratic orders, instead of the usual quaternion orders to which $\mathrm{End}_{\overline{\mathbb{F}}_p}(E)$ are isomorphic. The above theory of class group actions can then be applied, and by setting p to be one less than the product of many small primes, a technique borrowed from SIDH, the resulting protocol is a much more efficient variant of the CRS scheme [36] [82] called Commutative Supersingular Isogeny Diffie-Hellman, or CSIDH.

Security and the Ordinary Isogeny Graph

As one would expect, the underlying computational problems of this ordinary elliptic curve isogeny scheme arise from conditions (vi) and (vii) of Definition 2.2.2. Let p be prime and $q = p^n$.

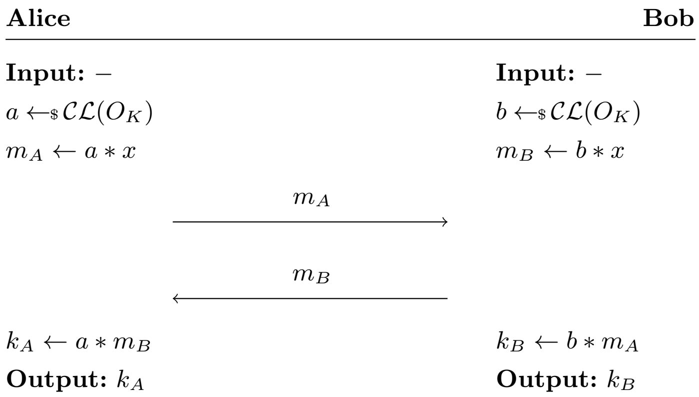


Figure 2.1: Ordinary isogeny-based key exchange protocol

Problem 2.2.4. *Let $E_1/\mathbb{F}_q$ and $E_2/\mathbb{F}_q$ be ordinary elliptic curves with $|E_1(\mathbb{F}_q)| = |E_2(\mathbb{F}_q)|$. Compute an $\mathbb{F}_q$-isogeny $\phi : E_1 \to E_2$.*

Problem 2.2.5. *Let $E_1/\mathbb{F}_q, E_2/\mathbb{F}_q$ and $E_3/\mathbb{F}_q$ be ordinary elliptic curves with $|E_1(\mathbb{F}_q)| = |E_2(\mathbb{F}_q)| = |E_3(\mathbb{F}_q)|$. Let $[\alpha] \in \mathcal{CL}(O_K)$ be such that $[\alpha] * E_1 = E_2$. Compute the unique (up to $\mathbb{F}_q$-isomorphism) elliptic curve $E_4 = [\alpha] * E_3$.*

The fastest known classical algorithm for solving Problem 2.2.4 is probabilistic with a worst-case and average-case of $O(q^{1/4+o(1)} \log^2(q) \log(\log(q)))$ [55]. This result is an improvement over [52] and is achieved by taking a pseudorandom walk in the isogeny-graph and using the easily computable small degree isogenies more often than larger degree isogenies.

With a quantum computer the most efficient algorithm [29] for the same problem has a subexponential running time of $L_q(\frac{1}{2}, \frac{\sqrt{3}}{2})$ under the Generalized Riemann Hypothesis, where

$$L_N(\alpha, c) := \exp[(c + o(1))(\ln N)^{\alpha}(\ln \ln N)^{1-\alpha}].$$

The authors propose reducing the problem to an instance of the Abelian hidden shift problem, and then using Kuperberg's quantum algorithm [66] which applies here because the reduction will be an *injective* hidden shift problem [29, 4.1].

As for the CSIDH protocol, much work has gone into adapting and improving the quantum algorithm of [29], with mixed results [11] [59] [75] [14].

2.2.2 Supersingular Elliptic Curve Cryptography

In 2011, De Feo, Jao and Plût [47] proposed a cryptosystem from supersingular elliptic curve isogenies. The central difference in the supersingular setting is that the endomorphism ring of the curve is non-Abelian (see Theorem 2.1.33, (iii)). The authors overcome this difficulty by having the participating members of the key-exchange send additional information about the isogeny: the image of four points on the curve.

Setup: Let p be a fixed prime number of the form $p = \ell_A^{e_A}\ell_B^{e_B}f \pm 1$, where ℓ_A and ℓ_B are distinct primes, and f is some small prime. The typical choices are $\ell_A = 2$ and $\ell_B = 3$. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve with $E[\ell_A^{e_A}]$ and $E[\ell_B^{e_B}]$ defined over $\mathbb{F}_{p^2}$. Let $P_A, Q_A, P_B, Q_B \in E(\mathbb{F}_{p^2})$ be four points such that $\langle P_A, Q_A\rangle = E[\ell_A^{e_A}]$ and $\langle P_B, Q_B\rangle = E[\ell_B^{e_B}]$ (by Theorem 2.1.7 each of these torsion subgroups require two points to generate).

Alice chooses two random elements $m_A, n_A \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$, not both divisible by ℓ_A, and computes the point $R_A := [m_A]P_A + [n_A]Q_A$ and the isogeny

$$\phi_A : E \to E_A$$

such that $ker(\phi_A) = \langle R_A\rangle$. Additionally, Alice computes the images of the $E[\ell_B^{e_B}]$ generators; $\phi_A(P_B), \phi_A(Q_B) \in E_A(\mathbb{F}_{p^2})$. Similarly, Bob computes a random linear combination R_B (chosen so that not both m_B and n_B are divisible by ℓ_B) of P_B and Q_B, the isogeny

$$\phi_B : E \to E_B$$

with kernel $\langle R_B\rangle$, and the points $\phi_B(P_A), \phi_B(Q_A) \in E_B(\mathbb{F}_{p^2})$. Alice and Bob's secret keys are the numbers m_A, n_A and m_B, n_B, respectively.

Using an unsecured channel, Alice sends $(E_A, \phi_A(P_B), \phi_A(Q_B))$ to Bob, and Bob sends $(E_B, \phi_B(P_A), \phi_B(Q_A))$ to Alice. The shared secret elliptic curve can now be computed by both parties. Alice computes the point $S_A := [m_A]\phi_B(P_A) + [n_A]\phi_B(Q_A) \in E_B(\mathbb{F}_{p^2})$ and the isogeny from E_B with kernel generated by S_A,

$$\phi_A' : E_B \to E_{BA}.$$

Similarly, Bob computes $S_B := [m_B]\phi_A(P_B) + [n_B]\phi_A(Q_B) \in E_A(\mathbb{F}_{p^2})$ and

$$\phi_B' : E_A \to E_{AB}$$

such that $ker(\phi_B') = \langle S_B\rangle$. The two curves E_{AB} and E_{BA} are isomorphic over $\mathbb{F}_{p^2}$, in particular they have equal j-invariants, and so the shared secret key is $j(E_{AB}) = j(E_{BA})$.

Alice		**Bob**
$m_A, n_A \leftarrow_{\$} \mathbb{Z}_{\ell_A^{e_A}}$		$m_B, n_B \leftarrow_{\$} \mathbb{Z}_{\ell_B^{e_B}}$
$R_A \leftarrow [m_A]P_A + [n_A]Q_A$		$R_B \leftarrow [m_B]P_B + [n_B]Q_B$
$\phi_A : E \to E/\langle R_A \rangle$		$\phi_B : E \to E/\langle R_B \rangle$
$\phi_A(P_B), \phi_A(Q_B)$		$\phi_B(P_A), \phi_B(Q_A)$
	$\xrightarrow{E_A, \phi_A(P_B), \phi_A(Q_B)}$	
	$\xleftarrow{E_B, \phi_B(P_A), \phi_B(Q_A)}$	
$S_A \leftarrow [m_A]\phi_B(P_A) + [n_A\phi_B(Q_A)$		$S_B \leftarrow [m_B]\phi_A(P_B) + [n_B]\phi_A(Q_B)$
$\phi'_A : E/\langle R_B \rangle \to (E/\langle R_B \rangle)/\langle S_A \rangle$		$\phi'_B : E/\langle R_A \rangle \to (E/\langle R_A \rangle)/\langle S_B \rangle$
$j_A \leftarrow j((E/\langle R_B \rangle)/\langle S_A \rangle)$		$j_B \leftarrow j((E/\langle R_A \rangle)/\langle S_B \rangle)$

Figure 2.2: Supersingular isogeny-based key exchange protocol

Figure 2.2.2 details the supersingular isogeny key exchange protocol. Here E, P_A, Q_A, P_B, and Q_B are all public parameters.

The SIDH key-exchange protocol can be made into a public-key encryption scheme in the following way. Let the setup be as above except include a hash function family $\mathcal{H} = \{H_k : k \in K\}$ indexed by a finite set K, where each H_k is a function from $\mathbb{F}_{p^2}$ to the message space $\{0,1\}^w$. Let key generation be the same except with an additional $k \in_R K$ included in the public and private keys.

Encryption: Given a public key $(E_A, \phi_A(P_B), \phi_A(Q_B), k)$ and message $m \in \{0,1\}^w$, choose two random $m_B, n_B \in_R \mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$ not both divisible by ℓ_B, and compute

$$h = H_k(j(E_{AB})),$$
$$c = m \oplus h.$$

The ciphertext is $(E_B, \phi_B(P_A), \phi_B(Q_A), c)$.

Decryption: Given a ciphertext $(E_B, \phi_B(P_A), \phi_B(Q_A), c)$ and a private key(m_A, n_A, k), compute the j-invariant $j(E_{AB})$ and set

$$h = H_k(j(E_{AB})),$$
$$m = h \oplus c.$$

The plaintext is m.

$$\begin{tikzcd}
 & E/\langle S\rangle \arrow[dr, "ker(\psi_3) = \langle \phi_2(R)\rangle"] \arrow[d, dashed, shift left] & \\
E \arrow[ur, "ker(\phi_2) = \langle S\rangle"] \arrow[dr, "ker(\phi_3) = \langle R\rangle"'] & & E/\langle S, R\rangle \\
 & E/\langle R\rangle \arrow[ur, "ker(\psi_2) = \langle \phi_3(S)\rangle"'] \arrow[u, dashed, shift left] &
\end{tikzcd}$$

Figure 2.3: Supersingular isogeny-based key exchange diagram

Security and the Supersingular Isogeny Graph

Listed below are the security assumptions under which the security of supersingular isogeny-based cryptosystems can be proven. The corresponding security theorems are included here, and the proofs can be found in [47]. Figure 2.3 helps explain why these are the underlying security assumptions.

The CSSI problem is perhaps the most important isogeny problem discussed in this body of work.

Problem 2.2.6 (Computational Supersingular Isogeny (CSSI) Problem)**.** *Let E_0 be a supersingular elliptic curve over $\mathbb{F}_{p^2}$ where $p = N_1N_2 - 1$ is a prime such that $\gcd(N_1, N_2) = 1$, and let $\phi_A : E_0 \to E_A$ be an isogeny whose kernel is generated by $\langle P_A + [r_A]Q_A\rangle$, for some $r_A \in \mathbb{Z}/N_1\mathbb{Z}$ and basis P_A, Q_A for $E_0[N_1]$. Given E_0, P_A, Q_A, E_A and the action of ϕ_A on $E_0[N_2]$, find r_A.*

For the sake of provable security of cryptosystems, other isogeny problems have been described [4, 58] including the supersingular decisional and computational Diffie-Hellman Problems. The following isogeny problems were originally stated in [47].

Problem 2.2.7 (SSDDH)**.** *Given a tuple sampled with probability 1/2 from one of the following two distributions:*

1. *(E_A, E_B, $\phi_A(P_B)$, $\phi_A(Q_B)$, $\phi_B(P_A)$, $\phi_B(Q_A)$, E_{AB}), generated via the SIDH protocol where*
$$E_{AB} \cong E_0/\langle [m_A]P_A + [n_A]Q_A, [m_B]P_B + [n_B]Q_B\rangle,$$

2. *$(E_A,\ E_B,\ \phi_A(P_B),\ \phi_A(Q_B),\ \phi_B(P_A),\ \phi_B(Q_A),\ E_C)$, where everything except E_C is generated via the SIDH protocol and*

$$E_C \cong E_0/\langle [m'_A]P_A + [n'_A]Q_A, [m'_B]P_B + [n'_B]Q_B\rangle,$$

where $m'_A, n'_A \in_R \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ are not both divisible by ℓ_A, and $m'_B, n'_B \in_R \mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$ are not both divisible by ℓ_B,

the Supersingular Decision Diffie-Hellman problem is to determine from which distribution the tuple is sampled.

Problem 2.2.8 (SSCDH)**.** *Let $\phi_A : E \to E_A$ be an isogeny whose kernel is $\langle [m_A]P_A + [n_A]Q_A\rangle$, and let $\phi_B : E \to E_B$ be an isogeny whose kernel is $\langle [m_B]P_B + [n_B]Q_B\rangle$, where m_A, n_A (respectively m_B, n_B) are randomly chosen from $\mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ (respectively $\mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$) and are not both divisible by ℓ_A (respectively ℓ_B). Given the curves E_A, E_B and the points $\phi_A(P_B), \phi_A(Q_B), \phi_B(P_A), \phi_B(Q_A)$, the Supersingular Computational Diffie-Hellman problem is to find the j-invariant of $E/\langle [m_A]P_A + [n_A]Q_A, [m_B]P_B + [n_B]Q_B\rangle$.*

SSDDH reduces to SSCDH in the obvious way, and the converse is true as well [93]: by adaptively solving the SSDDH problem (with the isogeny degree input as a parameter) one may compute the isogeny which solves the SSCDH problem in runtime a polynomial of $\log p$. Both the decisional and computational Diffie-Hellman assumptions depend on the CSSI problem in the obvious way.

We now provide the security theorems for these two cryptosystems. But before we do, as the public-key security definitions have not been stated yet in this work, we give those first. In the coming three definitions, let $\varepsilon(n)$ be any real-valued function such that for every polynomial $f(n)$, there exists an integer N such that

$$\varepsilon(n) < \frac{1}{f(n)},$$

for all $n > N$. Similarly, poly(n) is any real-valued function such that for some polynomial $f(n)$, one has poly$(n) < f(n)$ for all $n > N$. A cryptosystem is said to be **KP**, **OW-CPA**, or **IND-CPA**, if for each choice of poly(n) there exists a choice of $\varepsilon(n)$ for which the definition holds.

Definition 2.2.9. *A public-key cryptosystem is **key private**, or **KP**, if for any security parameter λ, for any probabilistic Turing machine $\mathcal{A}$ which terminates after time poly(λ),* *and any randomly chosen* $(pubkey, privkey)$ *pair produced by the key generation function* G *on input* λ*,*

$$Prob(\mathcal{A}(pubkey) = privkey) < \varepsilon(\lambda),$$

when the probability is over the joint probability distribution of output of G *and the random choices of* $\mathcal{A}$*.*

Definition 2.2.10. *A public-key cryptosystem is* ***one way****, or* ***OW-CPA****, if for any fixed* $(pubkey, privkey)$ *pair produced by the key generation function* G *on input of security parameter* λ*, and random message* m*,*

$$Prob(\mathcal{A}(pubkey, E(m)) = m) < \varepsilon(\lambda),$$

for any probabilistic Turing machine $\mathcal{A}$ *which terminates after time poly*(λ)*, and the probability is over the joint probability distribution of random choises of* m*, and the random choices made by the encryption function* E*, and the random choices of* $\mathcal{A}$*.*

Definition 2.2.11. *A public-key cryptosystem is* ***indistinguishable under chosen plaintext attack****, or* ***IND-CPA****, if there exists no probabilistic Turing machine* $\mathcal{A}$ *which terminates after time poly*(λ) *and for which, given a fixed* $(pubkey, privkey)$ *pair, and a ciphertext* $c = E(m_b)$ *for* $b \in \{0, 1\}$ *chosen uniformly at random, where* m_0 *and* m_1 *are messages chosen by* $\mathcal{A}$ *after receiving the public key, can determine the value of* b *in the sense that*

$$\left| Prob\left(\mathcal{A}(pubkey, c) = b\right) - \frac{1}{2} \right| < \varepsilon(\lambda).$$

We now return to the topic of the security of supersingular isogeny-based cryptography. The following theorems from Jao, De Feo, Plût [47] pertain to the security of SIDH key-exchange and encryption.

Theorem 2.2.12 ([47], 6.1)**.** *Under the SSDDH assumption, the key agreement protocol above is session-key secure in the authenticated-links adversary model of Canetti and Krawczyk [22].*

Theorem 2.2.13 ([47], 6.2)**.** *If the SSDDH assumption holds, and the hash function family* $\mathcal{H}$ *is entropy-smoothing, then the public-key cryptosystem above is IND-CPA.*

For any field K with $char(K) = p > 0$ and non-empty set of prime L with $p \notin L$, define the *supersingular isogeny graph* $X(K, L)$ where each vertex is a K-isomorphism class of elliptic curves defined over K (vertices have an associated, unique j-invariant), and the edges are equivalence classes of degree ℓ isogenies defined over K for $\ell \in L$, connecting isogenous curves. Given a path on $X(K, L)$ connecting say j_1 and j_2, an explicit starting curve with j-invariant j_1 must be chosen before the isogeny can be computed (see the canonical elliptic curve associated to each j in Section 2.1.16). Once this is done, the isogeny can be computed by composing each isogeny (edge) in the path [21].

If E_1 and E_2 are such that $j(E_1) = j_1$ and $j(E_2) = j_2$, then the isogeny computed in this way may have codomain elliptic curve isomorphic to E_2 in which case the composition of isogeny and isomorphism will give the correct isogeny. This differs from the ordinary isogeny graph where two elliptic curves in the same equivalence class (vertex) may be quadratic twists of each other.

By Theorem 2.1.33.(ii) every supersingular curve has j-invariant in $\mathbb{F}_{p^2}$, so setting $K = \mathbb{F}_{p^2}$ gives the full supersingular isogeny graph of L-isogenies. The authors of [41] instead set $K = \mathbb{F}_p$ and look at this restricted graph; below is their main result. First, let $h(\theta)$ denote the Hilbert class number of $\mathbb{Q}(\sqrt{\theta})$, and $(\frac{a}{b})$ denote the Legendre symbol for quadratic residues.

Theorem 2.2.14. *[41, Theorem 2.7] Let $p > 3$ be prime.*

(a) If $p \equiv 1 \pmod 4$, then there are $h(-4p)$ $\mathbb{F}_p$-isomorphism classes of supersingular elliptic curves over $\mathbb{F}_p$, all with the endomorphism ring $\mathbb{Z}[\sqrt{-p}]$. From each vertex there is exactly one outgoing $\mathbb{F}_p$-rational 2-isogeny, and two outgoing ℓ-isogenies for every prime $\ell > 2, \ell \in L$ such that $(\frac{-p}{\ell}) = 1$.

(b) If $p \equiv 3 \pmod 4$, then there are two cases. In both cases each vertex has two ℓ-isogenies for every prime $\ell > 2, \ell \in L$ such that $(\frac{-p}{\ell}) = 1$. Additionally, in both cases there are two "levels" to the graph defined as the "surface" and the "floor" of the graph. The endomorphism ring of every vertex on the surface is isomorphic to the order $\mathbb{Z}[\frac{1+\sqrt{-p}}{2}]$, while the endomorphism ring of every vertex on the floor is isomorphic to the order $\mathbb{Z}[\sqrt{-p}]$.

(i) If $p \equiv 7 \pmod 8$, then each level has $h(-p)$ vertices. The surface and the floor are connected $1 : 1$ with 2-isogenies, and on the surface there are also two 2-isogenies from each vertex to other vertices on the surface.

(ii) If $p \equiv 3 \pmod 8$, then there are $h(-p)$ vertices on the surface and $3h(-p)$ vertices on the floor. The surface and the floor are connected $1 : 3$ with 2-isogenies.

A consequence of this theorem is an algorithm to solve the supersingular isogeny problem over $\mathbb{F}_p$ with a classical running time of $O(p^{1/4})$ (assuming the Generalized Riemann Hypothesis), and the full supersingular isogeny problem over $\bar{\mathbb{F}}_p$ in $O(p^{1/2})$. Given two vertices of the graph (j-invariants) representing E and E_A, take isogenies to the surface if necessary and then perform a random walk using isogenies of degree $2^{e_A/2}$ from each vertex until a collision, j', is found. Choose an elliptic curve E' with $j(E') = j'$. The isogeny from E to E_A will be the composition of each isogeny in the walk from E to E' and the dual of each isogeny in the walk from E' to E_A.

Another consequence of this result is a quantum algorithm for the full supersingular isogeny problem (that is, over the base field $\mathbb{F}_{p^2}$ instead of $\mathbb{F}_p$) [12]. The first step is: given two supersingular elliptic curves over $\mathbb{F}_{p^2}$, use Grover's algorithm to find isogenous elliptic curves defined over $\mathbb{F}_p$. This computation has a quantum runtime of $O(p^{1/4})$. Then, using Theorem 2.2.14 and the algorithm of [41], the isogeny between these to elliptic curves defined over $\mathbb{F}_p$ can be computed classically in $O(p^{1/4})$. This gives a path in the supersingular isogeny graph and so the composition of each directed isogeny (dual isogenies give opposite directions) in the path will give the correct output (up to isomorphism) with a total quantum runtime of $O(p^{1/4})$.

Formally, the problem of finding a collision in a graph can be formulated in the following way:

Problem 2.2.15 (Claw Problem)**.** *Given function $f : A \to C, g : B \to C$ with $|A| = |B|$, find a pair $(a, b) \in A \times B$ such that $f(a) = g(b)$.*

A solution to this complexity problem using quantum computers was shown to be optimal in the black-box model [91], with runtime $O(\sqrt[3]{|A||B|})$. We can apply this to theSIDH setting. The degree of the isogeny between E and Alice's elliptic curve E_A is 2^{e_A}, so let A be the set of all isogenies of degree $2^{e_A/2}$ from E and let B be the set of all isogenies of degree $2^{e_A/2}$ from E_A. Here C will be the set of all elliptic curves E' defined over $\mathbb{F}_{p^2}$ with $|E(\mathbb{F}_{p^2})| = |E'(\mathbb{F}_{p^2})|$ (see Theorem 2.1.30). The sets A and B have equal cardinality and so the algorithm [91] applies. Hence, there is a quantum attack in $O(\sqrt[3]{2^{e_A/2}2^{e_A/2}}) = O(2^{e_A/3}) = O(p^{1/6})$ against supersingular isogeny schemes.

More recent work [61] has shown that instantiating Tani's claw finding algorithm [91] requires a greater quantum gate count than previously thought. Combined with more thourough classical cryptanalysis [1] [35] suggests that claw-finding attacks may in fact be slower than generic van Oorschot-Wiener collision finding algorithms (optimized for SIDH contexts) for this problem.

Chapter 3

The Isogeny Problem using $\mathrm{End}(E)$

Portions of this chapter were published in Bottinelli et al. [18]. The contents of this chapter represent my contribution to that work.

This chapter presents novel algorithms for passively solving the CSSI problem, and therefore the SSCDH and SSDDH problems as well. In addition to providing the action of ϕ_A on $E_0[N_2]$ in the CSSI problem, some parameter choices in isogeny-based cryptography schemes provide $\mathrm{End}(E)$ as well. All results in this chapter will rely on knowledge of $\mathrm{End}(E)$ (or some subring), so in fact, the algorithms presented solve a potentially easier problem. However, no asymptotic improvements have been made on solving this variant of the CSSI problem compared to the original. Later sections will use the action of ϕ_A on $E_0[N_2]$ to achieve efficiency improvements to the mentioned CSSI solving algorithms, and to multi-party variants of CSSI. We then analyze these algorithms to determine their optimal runtimes, and compare to the best-known algorithms for the CSSI problem.

3.1 Introduction

Supersingular isogeny-based public-key cryptosystems, such as SIDH, typically involve an smooth-degree isogeny as a secret key, the codomain elliptic curve as a public key, and have their security based on the SSDDH problem. While there seems to be only one active attack on SIDH [53], there are numerous passive attacks [91] [12] [76] [39]. The active attack showed that the repeated use of keys (static keys) in SIDH is insecure, but the passive attacks have not been able to find an algorithm for solving CSSI with a better runtime than those described in the original SIDH paper [58], which have a classical runtime of $O(p^{1/4})$ and a quantum runtime of $O(p^{1/6})$, where p is the field characteristic. In fact, under further analysis of those mentioned algorithms, the estimated runtimes and hence the estimated security of SIDH have increased [61] [1] [35].

This chapter introduces a passive algorithm for solving the CSSI problem. The high-level idea for this algorithm, presented formally in Section 3.3, is to determine some high-order bits of a secret key using a test which succeeds with some non-negligible probability. The test can then be altered using this new information to continuously test lower-order bits of the secret key, and eventually the entire secret key can be revealed from this process. This test itself is therefore a subroutine, described in Section 3.2, which takes as input an endomorphism of the starting elliptic curve and the public codomain elliptic curve, and outputs a Boolean depending on if the kernel subgroup of the secret isogeny between these two elliptic curves is fixed by the input endomorphism.

In order to determine if the input endomorphism fixes the secret kernel or not, we develop theory showing that if the kernel is fixed, then the codomain elliptic curve will have a similar endomorphism. The converse is true–if a similar endomorphism exists on the codomain curve, then the secret kernel is fixed–with overwhelming probability assuming the degree of the endomorphism is not too great relative to p. The efficiency of this subroutine test is then examined and quantified, and then improved upon in Section 3.4 using the action of ϕ_A on $E_0[N_2]$, as this information is provided in the CSSI problem setup.

By establishing an oracle to find endomorphisms of the starting elliptic curve satisfying degree and action constraints, we can then reduce the CSSI problem (and others) to the problem of realizing this oracle. However, Sections 3.5 and 3.6 study the realization of this oracle when the starting curve has j-invariant 1728, and prove that the CSSI solving algorithm requires $\tilde{O}(p^{1/4})$ computations, giving no speedup over the literature. Section 3.6 also explores the adaptation of this attack to multi-party variantions of SIDH and demonstrates asymptotic improvements in those cases. The improvements to the multi-party setting are due to the greater ratio of public torsion information to p.

3.2 Theory of Fixed Kernels

In this section we develop the basic theory for this chapter. The following proposition is the foundation for all other results in this chapter.

Proposition 3.2.1. *Let $k, N \in \mathbb{Z}$ be coprime. Let $E/\mathbb{F}_q$ be a supersingular elliptic curve, let $R \in E[N]$, and let $\psi \in \operatorname{End}(E)$ be cyclic with order k. Suppose $\phi : E \to E'$ is an* *isogeny with kernel $\langle R\rangle$. If $\langle R\rangle$ is fixed as a subgroup (or stabilized) by ψ, then there exists a cyclic endomorphism ψ' on E' of degree k. Furthermore, the kernel of ψ' is $\phi(\ker\psi)$.*

Proof. Let ψ' be the isogeny with domain E' and kernel $\phi(\ker\psi)$, and let ϕ' be the isogeny with domain $\psi(E)$ and kernel $\psi(\ker\phi)$. As $\gcd(k, N) = 1$, Figure 3.1 commutes.

By assumption ψ fixes $\langle R\rangle$, and so

$$\ker\phi' = \psi(\ker\phi) = \psi(\langle R\rangle) = \langle R\rangle = \ker\phi.$$

It follows from Theorem 2.1.31 that $\phi' \cong \phi$, and so $E^* \cong E'$. Hence, $\psi' : E' \to E^* \cong E'$ which implies that $\psi' \in \mathrm{End}(E')$ and is of the claimed degree k. Since $\ker\psi$ is cyclic, and $\gcd(k, N) = 1$, it follows that $\phi(\ker\psi)$ is cyclic as well. □

$$\begin{tikzcd}
E \arrow[r, "\phi"] \arrow[d, "\psi"'] & E' \arrow[d, "\psi'"] \\
E \arrow[r, "\phi'"] & E^*
\end{tikzcd}$$

Figure 3.1: Commutative diagram

See Appendix A for a digression relating Proposition 3.2.1 to Lemma 42.2.9 of [96].

The converse to Proposition 3.2.1 is also relevant. While the converse is not true in general, it is shown in Lemma 3.2.2 that the number of supersingular k-endomorphisms can be asymptotically bounded above by a function of only k, and therefore the probability that an arbitrary supersingular endomorphism ring contains a cyclic element of degree k decreases as p grows. Hence, when k is suitably small compared to p, the existence of a cyclic k-endomorphism on E' heuristically implies that, with overwhelming probability, $\ker\phi$ is fixed by ψ.

Lemma 3.2.2. *The number of j-invariants (supersingular or ordinary) over $\overline{\mathbb{F}}_p$ which are cyclically k-isogenous to themselves is $O(k\log\log k)$, for $k \geq 3$.*

Notice that the characteristic of the finite field p is absent from this formula.

Proof. The value in question can be computed as the number of roots of the classical k^{th}-modular polynomial, $\psi_k(x, x)$. This is then bounded by $\deg\psi_k(x, x) \leq 2\mu(k)$ [72, Theorem

6.1], where $\mu(k) = k\prod_{p|k}\left(1+\frac{1}{p}\right)$ is the Möbius function. We now show that μ has the claimed asymptotic growth.

If $\phi(k) = k\prod_{p|k}\left(1-\frac{1}{p}\right)$ is Euler's totient function, then note that

$$\mu(k)\phi(k) = k^2\prod_{p|k}\left(1+\frac{1}{p}\right)\left(1-\frac{1}{p}\right) \leq k^2,$$

which implies $\mu(k) \leq \frac{k^2}{\phi(k)}$. If $k \geq 3$, then $\phi(k) > \frac{k}{e^\gamma \log\log k + \frac{3}{\log\log k}}$ [81, Theorem 15], and the result follows. □

Under the assumption that these cyclic k-endomorphisms are randomly distributed among the supersingular isomorphism families, Lemma 3.2.2 gives an approximation of $\frac{e^\gamma k \log\log k}{\lfloor p/12 \rfloor}$ on the probability that an arbitrary supersingular elliptic curve admits a k-endomorphism.

The combination of Proposition 3.2.1 and Lemma 3.2.2 leads to the first main theorem of this chapter. If an endomorphism ψ of E is known, then the algorithm from Theorem 3.2.3 serves as an offline test of whether the kernel of an unknown isogeny ϕ with domain E is fixed by an endomorphism ψ.

Theorem 3.2.3. *Given*

1. *a supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p = N_1N_2 - 1$ for coprime $N_1 \approx N_2$,*
2. *an elliptic curve E' that is the codomain of an N_1-isogeny $\phi : E \to E'$, and*
3. *a k-endomorphism ψ of E, for some integer k where $\gcd(k, N_1) = 1$ and $k < N_1$,*

there exists a classical algorithm with worst case runtime $\tilde{O}(k^3)$ which decides whether $\psi(\ker\phi) = \ker\phi$ or not, but may give false positives with probability $\approx \frac{1}{\sqrt{p}}$. Further, if k is $\log p$-smooth, then the runtime is $\tilde{O}(\sqrt{k})$.

When refering to an integer k as being B-smooth, for some real value B, we mean the standard definition that each of the prime factors of k is less than or equal to B. The runtime of the algorithm mentioned in Theorem 3.2.3 will arise frequently in this section, and so we provide it as a lemma.

Lemma 3.2.4. *Given a prime p, two supersingular elliptic curves $E/\mathbb{F}_{p^2}$, $E'/\mathbb{F}_{p^2}$, and an integer k, there exists a classical algorithm to check if E and E' are k-isogenous with worst-case runtime $\tilde{O}(k^3)$ and best-case runtime $\tilde{O}(\sqrt{k})$, which happens when k is $\log(p)$-smooth.*

Proof. The following are two approaches to testing if E is k-isogenous to E':

1. computing an extension field $\mathbb{F}_q/\mathbb{F}_{p^2}$ such that $E[k] \subset E(\mathbb{F}_q)$ and then computing all k-isogenies with domain E and checking if any codomains are isomorphic to E', or even better applying a meet-in-the-middle strategy if k factors nicely to test if the curves are k-isogenous, and

2. computing the classical k^{th}-modular polynomial $\psi_k(x, y)$ and testing whether or not $\psi_k(j(E), j(E')) = 0$.

Consider first the computational problem of finding and constructing an appropriate field extension. This step requires an irreducible polynomial $f(x)$ for which the extension field $\mathbb{F}_{p^2}[x]/\langle f(x)\rangle$ contains the x and y-coordinates of the points in $E[k]$. One approach is to use an upper bound of k^2-1 on the extension degree needed for these points to exist, and to compute an arbitrary irreducible polynomial with that degree. A more efficient approach is to use division polynomials. Factoring the k^{th}-division polynomial over $\mathbb{F}_{p^2}$ will give many irreducible polynomials of equal degree which will give an appropriate field extension to contain all x-coordinates of $E[k]$. The degree of the k^{th}-division polynomial in x is $\frac{k^2-1}{2}$ when k is odd, and $\frac{k^2-4}{2}$ when k is even [98, Lemma 3.5]. Computing the k^{th}-division polynomial using the recursive formula from Section 2.1.3 requires $\tilde{O}(k^2)$ operations and $O(k^2)$ space (see [84, §5.1]). Therefore, by [62], finding a root of this polynomial takes $\tilde{O}(k^3)$ time. However, when k is, say B-smooth, this field can be constructed as a tower of extensions, which only takes $O(B \log k)$ time. A quadratic extension of this field will then be guaranteed to contain the y-coordinates as well, and thus all of $E[k]$.

Next we assume the appropriate field extension has been constructed. Now the goal is to test if $E/\mathbb{F}_q$ is k-isogenous to $E/\mathbb{F}_q$. When k is prime, constructing all k-isogenies with domain E using Vélu's formulas involves computing the $k+1$ isogenies of prime degree k and domain E. Prime degree isogenies currently require $O(k)$ operations to compute [95]. This case, therefore, gives us the worst-case bound of $O(k^2)$, as there are approximately k such isogenies to check. When k is not prime, claw-finding methods can be applied to improve performance. In the case where $k = k_1k_2$ for some $\log p$-smooth positive integers k_1 and k_2 each approximately of size $\sqrt{k}$, classical claw-finding will require computing $O(k_1)$ many isogenies of degree k_1 and computing $O(k_2)$ isogenies of degree k_2 [58, 5.1], and

$O(\sqrt{k})$ space. When k is $\log p$-smooth, the isogeny computations themselves are $O(\log k)$ which is negligible. Thus, the worst-case for testing for k-isogenies (assuming the extension field is computed) is when k is prime where the runtime is $\tilde{O}(k^2)$, and the best case is when k is $\log p$-smooth where the runtime is $\tilde{O}(\sqrt{k})$. □

Observe that if k is small (say, less than $100,000$ [90]) testing for k-isogenies can be performed by checking if the tuple $(j(E_A), j(E_A))$ is a root of the k^{th}-modular polynomial. However, this approach scales very poorly.

Proof of Theorem 3.2.3. The algorithm of Lemma 3.2.4 can be used with the prime $p = N_1N_2 - 1$, integer k, and supersingular elliptic curve $E'/\mathbb{F}_{p^2}$, to test if E' is k-isogenous to itself. We then output True if E' is k-isogenous to itself, and False otherwise.

If there are no k-isogenies from E' to itself, then there are no k-endomorphisms, so $\psi(\ker\phi) \neq \ker\phi$ by the contrapositive of Proposition 3.2.1. The algorithm outputs False in this case which coincides with the correct answer. If the algorithm from Lemma 3.2.4 does find a k-endomorphism of E', the algorithm outputs True, but there is a chance that $\psi(\ker\phi) \neq \ker\phi$. By Lemma 3.2.2, this false positive occurs with probability $\approx \frac{e^{\gamma}k\log\log k}{\lfloor p/12\rfloor} = \tilde{O}\left(\frac{1}{\sqrt{p}}\right)$ since $k < N_1 \approx \sqrt{p}$ by assumption. The runtime of this algorithm follows directly from Lemma 3.2.4. □

The result of Theorem 3.2.3 raises two main questions:

1. is it likely that a k-endomorphism exists for E when k is small enough to make this algorithm practical?

2. how much information can be learned by determining whether $\psi(\ker\phi) = \ker\phi$ or not?

In particular, the algorithm from Theorem 3.2.3 works best when k is small and ψ fixes approximately half of all subgroups in $E[N]$.

The remainder of the chapter is organized as follows: We examine the basic uses of Theorem 3.2.3 in Section 3.3. Section 3.4 discusses improvements when the action of ϕ is known; in particular the runtimes of the algorithms in Theorems 3.3.4 and 3.3.5 can be improved. Section 3.5 considers instantiating Oracle 3.3.3 on the supersingular elliptic curve with j-invariant 1728 when combined with the improvements of Section 3.4. Finally, Section 3.6 analyzes when the instantiation will be sufficient to affect the cryptanalysis of supersingular isogeny-based schemes.

3.3 Cryptanalysis from Fixed Kernels

In this section we will assume the existence of an oracle which outputs endomorphisms which fix a non-negligible proportion of subgroups in some $E[N]$, and show that knowing if $\psi(\ker\phi) = \ker\phi$ is sufficient to solve the SSDDH and CSSI problems. Additionally, we will discuss extracting more torsion images of secret isogenies from endomorphisms, even when the kernels of the secret isogenies themselves remain unknown.

First, we introduce a definition for the proportion of fixed N-subgroups of the endomorphism. These are appropriately called the N-eigenspaces of the endomorphism, as they are the eigenspaces of vectors when viewing the action of the endomorphism as a 2×2 matrix on the N-torsion subgroup. We focus on the subgroups of the form $P + [r]Q$, for a basis P, Q of $E[N]$ because this form is simpler to work with and contains most subgroups.

Definition 3.3.1. *Let $\psi \in \mathrm{End}(E)$ be an endomorphism of some supersingular elliptic curve $E/\overline{\mathbb{F}}_p$. For an integer N which is not divisible by p, and a basis P, Q for $E[N]$, define*

$$\mathsf{Eig}_N^{P,Q}(\psi) = \{X \in E[N] : \exists r \in \mathbb{Z}/N\mathbb{Z},\ X = P + [r]Q, \text{ and } \exists \lambda \in (\mathbb{Z}/N\mathbb{Z})^*, \psi(X) = [\lambda]X\}.$$

Observe that $\mathsf{Eig}_N^{P,Q}(\psi)$ contains only points which generate subgroups which are fixed by ψ. Indeed, if $Y \in \langle X\rangle$, then $\exists\ m \in \mathbb{Z}/N\mathbb{Z} : [m]X = Y$, and so

$$\psi(Y) = [m]\psi(X) = [m\lambda]X = [\lambda]Y \in \langle X\rangle.$$

Further, if $X \in \mathsf{Eig}_N^{P,Q}(\psi)$, then $|\langle X\rangle| = |\psi(\langle X\rangle)|$. Lastly, note that if ψ fixes all subgroups in $E[N]$, then $|\mathsf{Eig}_N^{P,Q}(\psi)| = N$.

We also introduce the following notation for the entire set of generators of subgroups of order N of the form $P + [r]Q$.

Notation 3.3.2. *Let $E(\mathbb{F}_{p^2})$ be an elliptic curve, N an integer, and P, Q a basis for $E[N]$. Set $E^{P,Q}[N] := \{P + [r]Q \in E[N] : r \in \mathbb{Z}/N\mathbb{Z}\}$.*

Observe that for all $N \in \mathbb{Z}$, $|E^{P,Q}[N]| = N$, and for all $\psi \in \mathrm{End}(E)$ and bases P, Q we have $\mathsf{Eig}_N^{P,Q}(\psi) \subseteq E^{P,Q}[N]$. We now introduce the oracle that will be used in our two reductions.

Oracle 3.3.3.
Input: *$p = N_1N_2 - 1$, a supersingular elliptic curve E, an integer ℓ, a set $S \subset E[N_1]$, and a basis P, Q for $E[N_1]$.*

Output: *A generator $K_0 \in E(\overline{\mathbb{F}}_p)$ of the kernel of a cyclic endomorphism ψ of E such that the following constraints hold:*

1. $|K_0| \leq N_1$,

2. $\gcd(|K_0|, N_1) = 1$,

3. $\left|\mathsf{Eig}_{N_1}^{P,Q}(\psi)\right| \geq \frac{1}{\ell}|S|$, *and*

4. $\left|E^{P,Q}[N_1] \setminus \mathsf{Eig}_{N_1}^{P,Q}(\psi)\right| \geq \frac{1}{\ell}|S|$,

or $\perp$ *if no endomorphism satisfying these constraints exists.*

The choice of ℓ will usually be $O(\log p)$ in the instances of Oracle 3.3.3 discussed in this chapter, and when performing cryptanalysis on SIDH, $\ell = \ell_A$. The set S will usually start as $E^{P,Q}[N_1]$ and iteratively become smaller as we search for generator of a secret isogeny.

The conditions 3 and 4 guarantee that a non-negligible proportion of subgroups will be fixed by the output endomorphisms when $\ell \in O(\log p)$, and likewise for the subgroups that will not be fixed. Lemma 3.2.4 can be used to determine the runtime of searching for an endomorphism with degree $|K_0|$. Combining these facts, Oracle 3.3.3 can be used to solve both the SSDDH and the CSSI problems, as we will now exhibit.

Theorem 3.3.4. *Let $E_0/\mathbb{F}_{p^2}$ be a supersingular elliptic curve such that $p = \ell_A^{e_A}\ell_B^{e_B} - 1$ for distinct primes ℓ_A and ℓ_B. When given access to an $\mathcal{O}$ as in Oracle 3.3.3, such that $\mathcal{O}$ succeeds for a non-negligible proportion of sets S, there exists a (classical) distinguisher for the SSDDH problem with advantage $\dfrac{1}{\ell_A^2}$, which makes $O(1)$ calls to $\mathcal{O}$, and runs in worst-case time $\tilde{O}\left(\ell_A^{3e_A}\right)$. Further, if the endomorphisms all have* $\log p$*-smooth degree, then the runtime is $\tilde{O}\left(\sqrt{\ell_A^{e_A}}\right)$.*

Proof. We will describe an algorithm $\mathcal{A}$ with access to Oracle 3.3.3 with non-negligible advantage in solving the SSDDH problem. The algorithm $\mathcal{A}$ takes in the following input to the SSDDH problem:

$$p, E_0, P_A, Q_A, P_B, Q_B, E_A, E_B, \phi_A(P_B), \phi_A(Q_B), \phi_B(P_A), \phi_B(Q_A), E',$$

where E' is either the shared secret E_{AB} or a randomly generated E_C which is $\ell_A^{e_A}\ell_B^{e_B}$-isogenous to E_0. Let $\mathcal{A}$ pass $p, E_0, \ell_A, S = E_0^{P,Q}[\ell_A^{e_A}]$, and the basis P_A, Q_A to Oracle 3.3.3, and receive some endomorphism ψ of E_0 that fixes some of the points in S as output. Letting $\mathcal{A}$ apply the algorithm from Theorem 3.2.3, with input E_0, E_A, and ψ, reveals if $\psi(\ker\phi_A) = \ker\phi_A$.

Observe, the set $\{\phi_B(X) \in E_B(\mathbb{F}_{p^2}) : X \in \mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\psi)\} \subset E_B[\ell_A^{e_A}]$ can be determined by decomposing $\mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\psi)$ into P_A and Q_A and using $\phi_B(P_A)$ and $\phi_B(Q_A)$. Let $\mathcal{A}$ pass $p, E_B, \ell_B, \phi_B(\mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\psi))$ and the basis $\phi_B(P_A), \phi_B(Q_A)$ to Oracle 3.3.3, and receive some endomorphism $\tilde{\psi}$ of E_B that fixes some of $\phi_B(\mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\psi))$ as output. Letting $\mathcal{A}$ apply the algorithm from Theorem 3.2.3, with input E_B, E', and $\tilde{\psi}$, reveals if $\tilde{\psi}$ fixes the kernel of the isogeny from E_B to E'.

With probability $\frac{1}{\ell_A}$, the endomorphism ψ does not fix $\mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\psi)$ (condition 4 of Oracle 3.3.3), and in that case it follows from Theorem 3.2.3 that $\ker \phi_A \not\subset \mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\psi)$. If moreover $\tilde{\psi}$ does fix the kernel of the isogeny from E_B to E', which occurs with conditional probability $\frac{1}{\ell_A}$ (condition 3 of Oracle 3.3.3), then it follows from Theorem 3.2.3 that $\phi_B(\ker \phi_A) \subset \mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\tilde{\psi})$. This contradicts the previous sentence because $\mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\tilde{\psi}) \subset \phi_B(\mathsf{Eig}^{P,Q}_{\ell_A^{e_A}}(\psi))$. In this case, where ψ does not fix the kernel and $\tilde{\psi}$ does, it follows that E' must not be E_{AB}. The distinguisher $\mathcal{A}$ for the SSDDH problem may then use this information to gain the claimed advantage in distinguishing between E_{AB} and E_C with the mentioned runtime following from Theorem 3.2.3. □

Oracle 3.3.3 can also be iterated to solve the CSSI problem.

Theorem 3.3.5. *Suppose we are given a starting supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$ such that $p = \ell_A^{e_A}\ell_B^{e_B} - 1$ for distinct primes ℓ_A and ℓ_B, the image of an $\ell_A^{e_A}$-degree isogeny $E_A = \phi_A(E_0)$, and access to an $\mathcal{O}$ as in Oracle 3.3.3, such that $\mathcal{O}$ fails only for a negligible proportion of sets S. Then there exists a (classical) algorithm which outputs $\ker \phi_A$ with non-negligible probability, makes $m = O(\log \ell_A^{e_A})$ queries to $\mathcal{O}$, and runs in worst-case time $\tilde{O}\left(\ell_A^{3e_A} \cdot m\right)$. Further, if the endomorphisms all have $\log p$-smooth degree, then the runtime is $\tilde{O}\left(\sqrt{\ell_A^{e_A}} \cdot m\right)$.*

The algorithm for Theorem 3.3.5 is presented now, and the proof will follow.

Algorithm 3.3.6.
Input: *E_0, p, E_A, and access to Oracle 3.3.3 denoted $\mathcal{O}$.*

Output: $\ker \phi_A$ *or* $\perp$.

1. *Let* $\langle P, Q\rangle = E_0[\ell_A^{e_A}]$.

2. *Let* $S_0 = E_0^{P,Q}[\ell_A^{e_A}]$*, and* $i = 0$*.*

3. *Call* $\mathcal{O}$ *with* $(p, E_0, \ell_A, S_i, P, Q)$ *:*

   If $\mathcal{O}$ *outputs* $\perp$*, then return* $\perp$*.*

   Else, obtain ψ_i *from* $\mathcal{O}$ *from Oracle 3.3.3.*

4. *Let* $X = S_i \cap \mathsf{Eig}_{\ell_A^{e_A}}^{P,Q}(\psi_i)$ *and* $Y = S_i \setminus \mathsf{Eig}_{\ell_A^{e_A}}^{P,Q}(\psi_i)$*.*

5. *Use the algorithm in Theorem 3.2.3 with input* E_0*,* p*,* E_A*, and* ψ_i *to determine if* $\psi_i(\ker\phi_A) = \ker\phi_A$ *or not.*

6. *If* $\psi_i(\ker\phi_A) = \ker\phi_A$*, then let* $S_{i+1} = X$*, otherwise if* $\psi_i(\ker\phi_A) \neq \ker\phi_A$*, then let* $S_{i+1} = Y$*.*

7. *Increment* i *and repeat Steps 3 to 7 until* $|S_i| = 1$*.*

8. *Return* S_i*.*

Proof of Theorem 3.3.5. We analyze the success probability and runtime of Algorithm 3.3.6. Let σ_1 be the fraction of sets $S \subseteq E_0^{P,Q}[\ell_A^{e_A}]$ for which $\mathcal{O}$ will succeed. By hypothesis we can write $\sigma_1 > 1 - \frac{1}{c\log p}$, for some constant c. Hence, with probability σ_1, the reduction makes it to Step 7 instead of outputting $\perp$.

Note that at the end of Step 6, $|S_{i+1}| \leq \frac{1}{\ell_A}|S_i|$ for all i. Let $m = \lceil \log \ell_A^{e_A} \rceil$. Then

$$|S_m| \leq \frac{1}{\ell_A^m}|S_0| \approx \frac{\ell_A^{e_A-1}(\ell_A+1)}{\ell_A^{e_A}} \approx 1.$$

This implies that $O(m)$ calls to $\mathcal{O}$ are required. The success probability of Algorithm 3.3.6 is therefore σ_1^m. Since $m \approx \frac{1}{2}\log p$, we have $\sigma_1^m \approx \left(1 - \frac{1}{2cm}\right)^m$ which approaches $\frac{1}{e^{1/2c}}$ as p grows.

Step 6 runs in $\tilde{O}(\ell_A^{3e_A})$ or $\tilde{O}(\sqrt{\ell_A^{e_A}})$ time depending on if the degree of ψ is a smooth number or not. □

These two theorems show how outputs from Oracle 3.3.3, that is, endomorphisms which fix a large proportion of subgroups in $E_0[\ell_A^{e_A}]$, can be used to solve different isogeny problems. In each theorem, only the existence of the endomorphisms on E_A, namely those ψ' which correspond to the output from Oracle 3.3.3, are being used with Theorem 3.2.3 to infer information about $\ker\phi_A$. However the endomorphism ψ' itself is discovered in the process (see the proofs of Theorem 3.2.3 and Lemma 3.2.4). We propose a technique which uses the explicit endomorphism ψ' on E_A to determine the images of torsion points under ϕ_A other than those in the SSDDH and CSSI problems. We discuss the idea underlying our methods in the simplest case: when the endomorphisms have the easiest form to work with.

Let E be a supersingular elliptic curve and consider some $\psi \in \text{End}(E)$ and $\phi : E \to E'$ satisfying $\psi(\ker \phi) = \ker \phi$. Recall Proposition 3.2.1 and Figure 3.1. By our kernel-fixing assumption, we can reillustrate the figure as in Figure 3.2.

$$\begin{tikzcd}
E \arrow[r, "\phi"] \arrow[d, "\psi"'] & E' \arrow[d, "\psi'"] \\
E \arrow[r, "\phi"] & E'
\end{tikzcd}$$

Figure 3.2: Commutative diagram assuming $\psi(\ker \phi) = \ker \phi$

Examining Figure 3.2, one would expect the action of ψ on E to be similar to the action of ψ' on E'. This is exactly what we aim to show.

Recall the notation $\phi|_{E[N]}$ representing the isogeny ϕ restricted to the subgroup $E[N]$, and similarly $\phi|_{\langle P,Q\rangle}$ for independent points P and Q. Write

$$\phi|_{\langle P,Q\rangle} = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right),$$

where $\phi(P) = [a]P + [c]Q$ and $\phi(Q) = [b]P + [d]Q$. We start with a basic preliminary lemma.

Lemma 3.3.7. *Let $\phi : E \to E'$ be a separable isogeny and let $\{P, Q\}$ be a basis for $E[N]$, where $N, \deg \phi$ and p are pairwise coprime. Then $E'[N] = \langle \phi(P), \phi(Q)\rangle$.*

Proof. As $|E[N]| = N^2$ and $\deg \phi$ are coprime, we have $\ker \phi \cap E[N] = \{\infty\}$. Applying the first isomorphism theorem on the restricted isogeny $\phi|_{E[N]}$, we find that

$$\phi(E[N]) \cong \frac{E[N]}{\ker\left(\phi|_{E[N]}\right)} \cong E[N].$$

There is only one subgroup of E' of this form, namely $E'[N]$. Thus, $\phi(E[N]) \cong E'[N]$. □

In Propoisiton 3.3.8 we show that the action of ψ on $E[N]$ is similar (i.e. conjugate) to the action of ψ' on $E'[N]$.

Proposition 3.3.8. *Let $E_0/\mathbb{F}_{p^2}$ be a supersingular elliptic curve such that $p = N_1N_2 - 1$ for coprime N_1 and N_2. Let $\phi : E \to E'$ be a separable isogeny and $\psi \in \operatorname{End}(E)$ such that $\psi(\ker\phi) = \ker\phi$. Let $\psi' \in \operatorname{End}(E')$ be the endomorphism from Proposition 3.2.1. Suppose $\{P, Q\}$ is a basis for $E[N]$ for some $N \geq 2$, and N_1, N, p, and $\deg\psi$ are all pairwise coprime. Let $\psi|_{\langle P,Q\rangle} = M$ for some matrix M. Then, $M \in GL_2(\mathbb{Z}/N\mathbb{Z})$, and*

$$\psi'|_{\langle\phi(P),\phi(Q)\rangle} = M.$$

Proof. The invertibility of M follows from the fact that the action of $\psi \circ \widehat{\psi} = N_1 \cdot I_{2\times 2}$ (with respect to all bases of $E[N]$) and $\gcd(N_1, N) = 1$.

By Lemma 3.3.7, we know that $E'[N] = \langle\phi(P), \phi(Q)\rangle$. Let

$$M = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}.$$

Then,

$$\begin{aligned} \phi \circ \psi \circ \widehat{\phi}\,(\phi(P)) &= \phi \circ \psi\,([N_1]P) \\ &= [N_1]\phi(\psi(P)) \\ &= [N_1]\phi([m_{11}]P + [m_{21}]Q) \\ &= [m_{11}N_1]\phi(P) + [m_{21}N_1]\phi(Q). \end{aligned}$$

Similarly,

$$\phi \circ \psi \circ \widehat{\phi}\,(\phi(Q)) = [m_{12}N_1]\phi(P) + [m_{22}N_1]\phi(Q).$$

Therefore,

$$\left(\phi \circ \psi \circ \widehat{\phi}\right)\Big|_{\langle\phi(P),\phi(Q)\rangle} = N_1M.$$

Since $\psi(\ker\phi) = \ker\phi$, we have

$$\phi \circ \psi = \psi' \circ \phi,$$

which implies

$$\phi \circ \psi \circ \widehat{\phi} = \psi' \circ \phi \circ \widehat{\phi} = \psi' \circ [N_1].$$

Therefore,

$$\begin{aligned} N_1 M &= \left(\phi \circ \psi \circ \widehat{\phi}\right)\Big|_{\langle \phi(P), \phi(Q) \rangle} \\ &= ([N_1] \circ \psi')|_{\langle \phi(P), \phi(Q) \rangle} \\ &= N_1|_{\langle \phi(P), \phi(Q) \rangle} \cdot \left(\psi'|_{\langle \phi(P), \phi(Q) \rangle}\right) \\ &= N_1 \cdot \left(\psi'|_{\langle \phi(P), \phi(Q) \rangle}\right). \end{aligned}$$

As N_1 is invertible modulo N, dividing both sides of the above equality by N_1 gives the proposed statement. □

To recap, if ψ preserves $\ker \phi$, then by Proposition 3.3.8, the matrix of ψ (with respect to $\{P, Q\}$) is equal to the matrix of ψ' (with respect to $\{\phi(P), \phi(Q)\}$). We are interested in the converse:

- Does the equation
$$\psi|_{\langle P, Q \rangle} = \psi'|_{\langle P', Q' \rangle} \tag{3.1}$$
reveal anything about the relationship between $\{P, Q\}$ and $\{P', Q'\}$?
- Furthermore, can we use this relationship to deduce information about the action of ϕ on $E_0[N]$?

We answer these questions in the affirmative in the case where the action is a diagonal matrix. First, Proposition 3.3.9 equates such P' and Q' from Eq. 3.1 to scalars of $\phi(P)$ and $\phi(Q)$. Second, Proposition 3.3.10 equates such P' and Q' to $\pm\phi(P)$ and $\pm\phi(Q)$ when Eq. 3.1 can be satisfied twice, by two distinct pairs (where ψ and ψ' would be one such pair) of endomorphisms.

Proposition 3.3.9. *Let the setup be the same as in Proposition 3.3.8. Further, suppose N is prime, and the matrix $\psi|_{\langle P, Q \rangle} = M$ is diagonal. If $\{R, S\}$ is some basis of $E'[N]$ such that $\psi'|_{\langle R, S \rangle} = M$, then there exist $\alpha, \beta \in \mathbb{Z}/N\mathbb{Z}$ such that $\phi(P) = [\alpha]R$ and $\phi(Q) = [\beta]S$.*

Proof. We will show that both $\phi(P)$ and R are eigenvectors of M of the same eigenvalue, and similarly both $\phi(Q)$ and S are eigenvectors of M of the same eigenvalue. Since $\psi(\ker \phi) = \ker \phi$, ψ' exists by Proposition 3.2.1. By Proposition 3.3.8 we know there exists some basis (namely $(\phi(P), \phi(Q))$) such that $\psi'|_{E'[N]} = M$, and further, by the proof, for any other basis of $E'[N]$ the matrix $\psi'|_{E'[N]}$ is similar to M.

Let $M = \left[\begin{smallmatrix} a & 0 \\ 0 & d \end{smallmatrix}\right]$. Proposition 3.3.8 states that ψ' acts as M with respect to the basis $\{\phi(P), \phi(Q)\}$, and so $\phi(P)$ and $\phi(Q)$ are eigenvectors of M of eigenvalue a and d, respectively. But we also have that R and S are eigenvectors of M of eigenvalue a and d, respectively, by assumption. This implies $\phi(P) = [\alpha]R$ and $\phi(Q) = [\beta]S$ for some $\alpha, \beta \in \mathbb{Z}/N\mathbb{Z}$ as N is prime. □

We are concerned with the computational cost of determining the action of a secret isogeny $\phi : E \to E'$. Proposition 3.3.9 tells us that by searching for a basis on a prime torsion subgroup of E' such that an endomorphism $\psi' \in \mathrm{End}(E')$ behaves like the corresponding endomorphism $\phi \in \mathrm{End}(E)$ (the correspondence arising from Proposition 3.2.1), we can infer how ϕ acts up to scalar multiples. Given how ψ' acts with respect to any basis for $E[N]$, it is simple linear algebra to find the basis to make ψ' act with the same matrix as ψ, and so the computational cost of this step is small.

Next, we assume further that we have two distinct endomorphisms, $\psi_1, \psi_2 \in \mathrm{End}(E)$, which fix the kernel of some isogeny ϕ to examine what benefit this gives to the goal of determining the action of ϕ.

Proposition 3.3.10. *Let E be a supersingular elliptic curve over $\mathbb{F}_{p^2}$, let $\phi : E \to E'$ be a separable isogeny, and N a prime coprime to $\deg \phi$. Given $\psi_1, \psi_2 \in \mathrm{End}(E)$ such that*

1. *$\psi_i(\ker \phi) = \ker \phi$ for $i = 1, 2$,*

2. *ψ_1', ψ_2' are the endomorphisms of E' corresponding repsectively to ψ_1, ψ_2 from Proposition 3.2.1,*

3. *ψ_1 acts on $E[N]$ as a diagonal matrix M_1 with respect to a basis $\{P_1, Q_1\}$,*

4. *ψ_2 acts on $E[N]$ as a diagonal matrix M_2 with respect to a basis $\{P_2, Q_2\}$,*

5. *$\deg \psi_1, \deg \psi_2$ each are pairwise coprime with $N, \deg \phi$ and p, and*

6. *$P_2 = [\gamma]P_1 + [\delta]Q_1$ for some integers $\gamma, \delta \in (\mathbb{Z}/N\mathbb{Z})^*$,*

there exists an algorithm which can determine the tuple $(\phi(P_1), \phi(Q_1))$ up to a sign, from the input of $E, E', \psi_1, \psi_2, \psi_1', \psi_2'$, with runtime $\tilde{O}(N^3)$.

By "up to a sign" here we mean that either $(\phi(P_1), \phi(Q_2))$ or $(-\phi(P_1), -\phi(Q_2))$ will be output.

Proof. We give a proof by presenting the claimed algorithm followed by a justification of its correctness and analysis of its runtime.

Algorithm 3.3.11.
Input: *E, E', N, bases $\{P_1, Q_1\}, \{P_2, Q_2\}$ of $E[N]$, $\psi_1, \psi_2 \in \mathrm{End}(E)$, and $\psi_1', \psi_2 \in \mathrm{End}(E)$ as defined in the statement of the proposition.*

Output: *$\pm(\phi(P_1), \phi(Q_1))$.*

1. *For $i = 1, 2$: find a basis $\{R_i, S_i\}$ of $E'[N]$ such that the action of ψ_i' on $E'[N]$ with respect to a basis $\{R_i, S_i\}$ is M_i.*
2. *Determine $\gamma, \delta \in (\mathbb{Z}/N\mathbb{Z})^*$ such that $P_2 = [\gamma]P_1 + [\delta]Q_1$.*
3. *Without loss of generality (by switching R_2 and S_2 if necessary) it is possible to find $\gamma', \delta' \in (\mathbb{Z}/N\mathbb{Z})^*$ such that $R_2 = [\gamma']R_1 + [\delta']S_1$.*
4. *Evaluate the Weil pairings $g = e_N(P_1, Q_1)$ and $h = e_N(R_1, S_1)$.*
5. *Solve the discrete log for x in $g^{N_1} = h^x$.*
6. *Solve $\alpha^2 \equiv \frac{\delta\gamma'}{\delta'\gamma}x \mod N$ for $\alpha \in (\mathbb{Z}/N\mathbb{Z})^*$.*
7. *Set $\beta \equiv \alpha^{-1}x \mod N$.*
8. *Return $\phi(P_1) = [\alpha]R_1$, $\phi(Q_1) = [\beta]S_1$.*

We now justify the correctness of Algorithm 3.3.11. By Proposition 3.3.9, there exist $\alpha, \beta \in (\mathbb{Z}/N\mathbb{Z})^*$ such that $\phi(P_1) = [\alpha]R_1$ and $\phi(Q_1) = [\beta]S_1$. Observe

$$\phi(P_2) = [\gamma]\phi(P_1) + [\delta]\phi(Q_1) = [\gamma\alpha]R_1 + [\delta\beta]S_1,$$

and by Proposition 3.3.9 there is some integer ϵ such that

$$\phi(P_2) = [\epsilon]R_2 = [\epsilon\gamma']R_1 + [\epsilon\delta']S_1.$$

By comparing coefficients we see that

$$\epsilon = \alpha\gamma\gamma'^{-1} = \beta\delta\delta'^{-1},$$

which implies $\alpha = \left(\frac{\delta\gamma'}{\delta'\gamma}\right)\beta$. The Weil pairing gives us our second equation relating α and β by applying Proposition 2.1.25:

$$e_N\left(\phi(P_1), \phi(Q_1)\right) = e_N(P_1, Q_1)^{N_1} = g^{N_1},$$

and,

$$e_N\left(\phi(P_1), \phi(Q_1)\right) = e_N\left(R_1, S_1\right)^{\alpha\beta} = h^x.$$

Therefore, $x = \alpha\beta$ can be computed by solving discrete log as described in Step 5. Substituting for β from the previous equation in terms of α, we get the expression for α^2 in Step 6. Hence, we obtain two solutions $\pm(\alpha, \beta)$, and the result follows. The cost of Steps 2–6 is dominated by the cost of constructing the extension field defining $E[N]$, which is $\tilde{O}(N^3)$ by the proof of Lemma 3.2.4. □

3.4 Improvements from Isogeny Actions

With motivation from the public key encryption scheme SIDH, we consider the case when $\phi|_G$ is known for some subgroup $G \subset E(\mathbb{F}_{p^2})$. When this information is public it becomes easier to compute endomorphisms on $E' = E/\ker\phi$. It has been shown that the cost of computing endomorphisms on E' can be reduced by a factor of $|G|$ when $\phi|_G$ is known [76]. This section shows how to improve this factor to $|G|^2$.

Consider the setup of Theroem 3.2.3: let N, g and k be integers, and suppose $\mathrm{End}(E)$, $\phi(E[g])$, where $\phi : E \to E'$ has degree N, and $\psi \in \mathrm{End}(E)$ with degree k, which fixes $\ker\phi$, are known. By Theorem 3.2.3, there exists $\psi' \in \mathrm{End}(E')$ with kernel $\phi(\ker\psi)$. Further, suppose $g^2 \mid k$, and write $k = g^2k'$. This section will show that it is then drastically more computationally simple to find the g^2 component of $\ker\psi'$ on E' than was outlined in the proof of Theorem 3.2.3. We begin with a definition.

Definition 3.4.1. *Suppose ψ is a cyclic endomorphism of E. A **triangular decomposition** of ψ with respect to g is a triple of cyclic isogenies ψ_0, ψ_1, ψ_2, where*

1. *ψ_0 and ψ_1 have degrees dividing g,*
2. *$\psi = \widehat{\psi}_0 \circ \psi_2 \circ \psi_1$, and*
3. *if $\gcd(g, \deg\psi_2) \neq 1$, then $\deg\psi_0 = \deg\psi_1 = g$.*

A ***triangular kernel*** *of ψ with respect to g is a triple of torsion points denoted by $\Delta_\psi = (K_0, K_1, K_2)$, which generate the kernels of the corresponding isogenies of a triangular decomposition, that is, $\ker \psi_i = \langle K_i \rangle$. Furthermore, let $k' = |K_2|$.*

This representation has the advantage that only the extension field containing the k'-torsion points is needed to write the kernel, instead of the g^2k'-torsion points. This is because $K_0, K_1 \in E[g]$. Notice that K_0, K_1 and K_2 could theoretically all be trivial.

Notation 3.4.2. *Let ψ_0, ψ_1, ψ_2 denote a triangular decomposition of ψ with respect to g. Let E_0, E_1 and E_2 denote the images of ψ_0, ψ_1, ψ_2, respectively, as illustrated in Figure 3.3. Then, up to isomorphism, $E_0 \cong E_2$.*

$$\begin{tikzcd}
 & & E \arrow[dr, "\psi_1"] \arrow[ddrr, bend right, "\psi_0"'] & & \\
E \arrow[loop right, "\psi"] & & & E_1 \arrow[dr, "\psi_2"] & \\
 & & & & E_0 \cong E_2
\end{tikzcd}$$

Figure 3.3: Decomposing endomorphisms

We wish to study a passive adversary's ability to transfer ψ on E over to a corresponding potential endomorphism on E', using the isogenies in the triangular decomposition of ψ (so that they can test if Alice's private key is an eigenvector of ψ). In order to calculate the corresponding objects on E', we introduce notation for additional isogenies.

Notation 3.4.3. *Let ψ'_0 be the isogeny with domain E' and kernel $\phi(\ker \psi_0)$, and ϕ_0 be the isogeny with domain E_0 and kernel $\psi_0(\ker \phi)$. Let the image elliptic curves be $E'_{0,?} = \psi'_0(E')$ and set $E'_0 = \phi_0(E_0)$, as illustrated in Figure 3.4.*

Since $\deg \phi_0$ and $\deg \phi$ are relatively prime, $E'_0 \cong E'_{0,?}$.

$$\begin{tikzcd}
E \arrow[r, "\phi"] \arrow[dr, "\psi_0"'] & E' \arrow[dr, "\psi'_0"] & \\
 & E_0 \arrow[r, "\phi_0"] & E'_0 \cong E'_{0,?}
\end{tikzcd}$$

Figure 3.4: Maps with known kernels

Notation 3.4.4. *We decompose the isogeny with domain E' and kernel equal to $\phi(\ker \psi_2 \circ \psi_1)$ as $\psi'_2 \circ \psi'_1$, where $\ker \psi'_1 = \phi(\ker \psi_1)$ and $\ker \psi'_2 = \psi'_1 \circ \phi(\ker \psi_2)$. Let ϕ_2 be the isogeny with domain E_2 whose kernel is $\psi_2 \circ \psi_1(\ker \phi)$. Denote the images by $E'_1 = \psi'_1(E'), E'_{2,?} = \psi'_2(E'_1)$ and $E'_2 = \phi_2(E_2)$.*

Note that $\deg \psi'_1 = \deg \psi_1$ (which implies, $\deg \psi'_1 \mid g$), and $\deg \psi'_2 = \deg \psi_2$. Since $\deg \psi_2 \circ \psi_1$ and $\deg \phi$ are relatively prime, $E'_{2,?} \cong E'_2$ as shown in Figure 3.5.

$$\begin{tikzcd}
E \arrow[rr, "\phi"] \arrow[dr, "\psi_1"'] & & E' \arrow[dr, "\psi'_1"] & & \\
& E_1 \arrow[dr, "\psi_2"'] & & E'_1 \arrow[dr, "\psi'_2"] & \\
& & E_2 \arrow[rr, "\phi_2"] & & E'_2 \cong E'_{2,?}
\end{tikzcd}$$

Figure 3.5: Maps with known and unknown kernels

We now see that the knowledge of $\phi(E[k])$ can be used to calculate ψ'_0 and ψ'_1. Putting the previous two diagrams together gives us Figure 3.6.

$$\begin{tikzcd}
E \arrow[rrr, "\phi"] \arrow[dr, "\psi_1"] \arrow[ddrr, bend right, "\psi_0"'] & & & E' \arrow[dr, "\psi'_1"] \arrow[dddrr, bend right, "\psi'_0"'] & & \\
& E_1 \arrow[dr, "\psi_2"] & & & E'_1 \arrow[dr, "\psi'_2"] & \\
& & E_0 \cong E_2 \arrow[rrr, "\phi_2"] \arrow[drrr, "\phi_0"'] & & & E'_2 \\
& & & & & E'_0
\end{tikzcd}$$

Figure 3.6: Combining 3.4 and 3.5

The goal is to adapt the results of Theroem 3.2.3 to incorporate the addition of torsion information. The following lemma is analogous to Proposition 3.2.1.

Lemma 3.4.5. *Suppose* $\gcd(k, N) = 1$*. If* $\psi(\ker\phi) = \ker\phi$*, then* E_1' *is* k*-isogenous to* E_0' *via the isogeny* ψ_2' *described above.*

Proof. Suppose $\ker\phi = \langle R\rangle$. Choose a triangular decomposition of ψ as follows:

$$\psi = \widehat{\psi}_0 \circ \psi_2 \circ \psi_1.$$

Let $\tilde{\psi} = \psi_2 \circ \psi_1 \circ \widehat{\psi}_0$. Then $\tilde{\psi}$ is an endomorphism of E_0. Moreover,

$$\begin{aligned}\tilde{\psi}(\psi_0(R)) &= \psi_2 \circ \psi_1 \circ \widehat{\psi}_0(\psi_0(R)) \\ &= \psi_2 \circ \psi_1([\deg\psi_0]R) \\ &= [\deg\psi_0]\psi_2 \circ \psi_1(R) \\ &= \psi_0(\widehat{\psi}_0 \circ \psi_2 \circ \psi_1)(R) \\ &= \psi_0(\psi(R)).\end{aligned}$$

However, as R generates a subgroup which is fixed by ψ, say $\psi(R) = [\lambda]R$ for some integer λ satisfying $\gcd(\lambda, N) = 1$, this and the above equation imply

$$\tilde{\psi}(\psi_0(R)) = [\lambda]\psi_0(R).$$

Thus $\langle\psi_0(R)\rangle$ is fixed by $\tilde{\psi}$.

Recall that $\langle\psi_0(R)\rangle$ is the kernel of the isogeny ϕ_0 on E_0, see Figure 3.7. Therefore, we can apply Proposition 3.2.1, proving the existence of an endomorphism $\tilde{\psi}'$ on E_0', where $\ker\tilde{\psi}' = \phi_0(\ker\tilde{\psi})$. Let ϕ_0' be the isogeny on E_0 with kernel $\tilde{\psi}(\ker\phi_0)$. Since $\gcd(k, N) = 1$, the following equation holds: $\phi_0' \circ \tilde{\psi} \cong \tilde{\psi}' \circ \phi_0$.

$$\begin{tikzcd}
E_0 \arrow[rr, "\phi_0"] \arrow[dr, "\tilde{\psi}"'] & & E_0' \arrow[dr, "\tilde{\psi}'"] & \\
& E_0 \arrow[rr, "\phi_0'"] & & E_0'
\end{tikzcd}$$

Figure 3.7: $\tilde{\psi}$ fixing the kernel of ϕ_0

It remains to be shown that $E_0' \cong E_2'$, that is, the codomain of ϕ_0' is isomorphic to E_0'.

Similar to the above we find

$$\begin{aligned}\ker \phi_0' &= \tilde{\psi}(\ker \phi_0)\\ &= \psi_2 \circ \psi_1 \circ \widehat{\psi}_0(\langle \psi_0(R) \rangle)\\ &= \psi_2 \circ \psi_1(\langle [\deg \psi_0] R \rangle)\\ &= [\deg \psi_0](\ker(\phi_2))\\ &= \ker(\phi_2).\end{aligned}$$

Then,

$$\begin{aligned}E_2' = \phi_2(E_2) &\cong \phi_0'(E_0)\\ &\cong \phi_0' \circ \tilde{\psi}(E_0) \cong \tilde{\psi}' \circ \phi_0(E_0) \cong E_0'.\end{aligned}$$

Thus E_0' and E_2' are isomorphic. Finally, ψ_2' is a k-isogeny between E_1' and E_2', which implies E_1' and E_0' are k-isogenous. □

Now that Lemma 3.4.5 is proven, it is clear how to improve the algorithm in Theorem 3.2.3 in the case when the action on some $E[g]$ is also provided for some $g \mid N_2^2$.

Theorem 3.4.6. *Given*

1. *a supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p = N_1N_2 - 1$ for coprime $N_1 \approx N_2$, where N_2 is $\log p$-smooth,*
2. *an elliptic curve E' that is the codomain of an N_1-isogeny $\phi : E \to E'$,*
3. *the image of ϕ on $E[N_2]$, and*
4. *a k-endomorphism ψ of E for some integer k where $\gcd(k, N_1) = 1$, and if g is the greatest integer such that $g \mid N_2^2$, and $g \mid k$, then $k' := \frac{k}{g} < N_1$,*

there exists a classical algorithm with worst case runtime $\tilde{O}(k'^3)$ which decides whether $\psi(\ker \phi) = \ker \phi$ or not, but may give false positives with probability $\approx \frac{1}{\sqrt{p}}$. Further, if k' is $\log p$-smooth, then the runtime is $\tilde{O}\left(\sqrt{k'}\right)$.

Proof. We start by describing the algorithm referred to in the theorem, thereby showcasing its existence, and subsequently analyze its running time and success probability to prove the theorem. The probability of a false-positive (the algorithm outputing that $\psi(\ker\phi) = \ker\phi$ when that is not the case), can be approximated using the mixing properties of the isogeny graph.

Algorithm 3.4.7.
***Input**: $E, p, N_1, N_2, E', \phi(E[N_2])$, and a triangular kernel (with respect to g) for ψ : (K_0, K_1, K_2).*

***Output**: True or False.*

1. *Use $\phi(E[N_2])$ to compute $\phi(K_0)$ and $\phi(K_1)$.*
2. *Compute the isogenies ψ_0', and ψ_1' with respective kernels $\langle\phi(K_0)\rangle$ and $\langle\phi(K_1)\rangle$ (see Figure 3.6).*
3. *For all k'-isogenies from E_0', check if their codomain has j-invariant $j(E_1')$.*
4. *If such an isogeny is found, then return True, otherwise return False.*

First, we discuss the success probability. If Algorithm 3.4.7 returns False, then $\ker\phi$ is not fixed by ψ by the contrapositive of 3.4.5. Suppose Algorithm 3.4.7 returns True. Notice that the total number of non-backtracking isogenies from E_0' of degree k', if we write the factorization $k' = \prod_{1\le i\le r} q_i^{e_i}$, is

$$\prod_{1\le i\le r}(q_i+1)q_i^{e_i-1}.$$

Also, we know that there are approximately $\frac{p}{12}$ isomorphism classes of supersingular elliptic curves in an isogeny graph. From these two pieces of information we deduce that the probability that there is a cyclic k'-isogeny between E_0' and E_1' is no more than

$$\frac{12}{p}\prod_{1\le i\le r}(q_i+1)q_i^{e_i-1}.$$

This probability is negligible since $k' < N_1 \approx \sqrt{p}$. Therefore, under this assumption on k', if there is a k'-isogeny from E_0' to E_1', then the kernel subgroup is fixed by ψ.

Next, we discuss the runtime. Steps 1 and 2 are efficient in p since N_2 is $\log p$-smooth. The analysis of verifying when E_0' and E_1' are k'-isogenous is identical to the proof of Theorem 3.2.3. Thus, the worst case is when k' is prime, with runtime $O(k'^3)$. □

In Section 3.5 we demonstrate that supersingular elliptic curves with j-invariant 1728 likely do not have an endomorphism which satisfies the conditions of Theorem 3.4.6 (for instance, those in the Round 1 submission of SIKE). However, in Section 3.6 we give heuristic arguments as to why such endomorphisms may exist in multi-party variants of SIDH, such as group key-exchange [50, 4].

Algorithm 3.4.7 will be a subroutine in the following reduction (Theorem 3.4.10) of computational problems. That reduction will assume an oracle which outputs triangular kernels of endomorphisms, and then uses Algorithm 3.4.7 with each of those endomorphisms. We start by presenting the oracle which we will use in our reduction.

As mentioned previously, it is useful for the oracle to output a triangular kernel, instead of the kernel, to avoid unnecessary extension fields. Since we are no longer discussing a single k'-isogeny, with $k' \leq N_1$, but potentially multiple isogenies from repeated calls to an oracle, we instead use $K \leq N_1$ to denote the upper bound on all such k'. We now introduce the oracle that will be used in our two reductions.

Oracle 3.4.8.
Input: *$p = N_1 N_2 - 1$, a supersingular elliptic curve E, integers K and ℓ, a set $S \subset E[N_1]$, and a basis P, Q for $E[N_1]$.*

Output: *The triangular kernel (K_0, K_1, K_2) of a cyclic endomorphism ψ of E such that the following constraints hold:*

1. $|K_2| \leq K$,
2. $\gcd(|K_2|, N_1) = 1$,
3. $\left|\mathsf{Eig}_{N_1}^{P,Q}(\psi)\right| \geq \frac{1}{\ell}|S|$, *and*
4. $\left|E^{P,Q}[N_1] \setminus \mathsf{Eig}_{N_1}^{P,Q}(\psi)\right| \geq \frac{1}{\ell}|S|$,

or $\perp$ if no endomorphism satisfying these constraints exists.

Theorem 3.4.9. *Suppose we are given a supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p = N_1 N_2 - 1$ for coprime $N_1 \approx N_2$, where N_2 is $\log p$-smooth, and access to an $\mathcal{O}$ as in Oracle 3.4.8 such that $\mathcal{O}$ succeeds for a non-negligible proportion of sets S when $K = N_1$ and $\ell = O(\log p)$. Then there exists a (classical) distinguisher for the SSDDH problem with advantage $\frac{1}{\ell^2}$, which makes $O(1)$ calls to $\mathcal{O}$, and runs in worst-case time $\tilde{O}(K^3)$. Further, if the endomorphisms returned by $\mathcal{O}$ all have $\log p$-smooth degree, then the runtime is $\tilde{O}\left(\sqrt{K}\right)$.*

Proof. The distinguisher in question is exactly that described in the proof of Theorem 3.3.4, except calls to Oracle 3.3.3 are replaced by Oracle 3.4.8. □

The next theorem is the main reduction of this section: from Oracle 3.4.8 to the CSSI problem.

Theorem 3.4.10. *Given*

1. *a starting supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p = N_1N_2 - 1$ for coprime N_1 and $\log p$-smooth N_2,*
2. *the image of an N_1-degree isogeny $E' = \phi(E)$,*
3. *the action of ϕ on $E[N_2]$, and*
4. *access to an $\mathcal{O}$ as in Oracle 3.4.8 such that $\mathcal{O}$ fails only for a negligible proportion of sets S when $K = N_1$,*

there exists a (classical) algorithm which outputs $\ker\phi$ with non-negligible probability, makes $m = O(\log N_1)$ queries to $\mathcal{O}$, and runs in worst-case time $\tilde{O}(K^3 \cdot m)$. Further, if the endomorphisms returned by $\mathcal{O}$ all have $\log p$-smooth degree, then the runtime is $\tilde{O}\left(\sqrt{K} \cdot m\right)$.

We now present the algorithm that is referred to in Theoerem 3.4.10. Algorithm 3.4.11 iteratively reduces the size of the search space, which is denoted S_i at the i^{th} iteration, for a generator of $\ker\phi$.

Algorithm 3.4.11.
Input:

1. *E, p, N_1, N_2, such that $p = N_1N_2 - 1$ and E is supersingular over $\mathbb{F}_{p^2}$,*
2. *the action of ϕ on $E[N_2]$,*
3. *E', where $\phi : E \to E'$, and*
4. *access to an $\mathcal{O}$ as in Oracle 3.4.8 denoted $\mathcal{O}$.*

Output: *A generator of $\ker\phi$, or $\perp$.*

1. *Let P, Q be a basis for $E[N_1]$, set $S_0 = E^{P,Q}[N_1]$, and set $i = 0$.*

```
2. Set $K = N_1$.
3. Call $\mathcal{O}$ with $p, E, K$, and $S_i$.
      If $\mathcal{O}$ outputs $\perp$, then return $\perp$.
4. While $\mathcal{O}$ outputs a solution:
      Halve $K$ and call $\mathcal{O}$. Let $K$ be the last value where $\mathcal{O}$ did not output $\perp$.
   Let $(K_0, K_1, K_2)$ be the triangular kernel output of $\mathcal{O}$ called with $p, E, K$, and $S_i$.
5. Let $X = S_i \cap \mathsf{Eig}^{P,Q}_{N_1}(\psi)$ and $Y = S_i \setminus \mathsf{Eig}^{P,Q}_{N_1}(\psi)$.
6. Use Algorithm 3.4.7 with input $E, p, N_1, N_2, E', \phi(E[N_2])$, and a triangular kernel
   (with respect to $g$) for $\psi, (K_0, K_1, K_2)$, to determine whether $R \in X$ or $R \in Y$.
7. If $R \in X$, then let $S_{i+1} = X$. Otherwise if $R \in Y$, then let $S_{i+1} = Y$.
8. Increment $i$ and repeat Steps 2 to 7 until $|S_i| = 1$.
9. Return $S_i$.
```

We now analyze Algorithm 3.4.11, thereby proving Theorem 3.4.10.

Proof. (of Theorem 3.4.10) The proof consists of analyzing the success probability and runtime of Algorithm 3.4.11. In particular, we show that in the setting of Theorem 3.4.10, Algorithm 3.4.11 runs in time $O(K^3 \cdot \mathrm{poly}(\log p))$.

Let σ_1 be the fraction of sets $S \subseteq E^{P,Q}[N_1]$ for which there exists a non-negligible proportion of $K \in \{0, 1, \ldots, N_1\}$ for which $\mathcal{O}$ will succeed. By hypothesis σ_1 is exponentially close to 1. Hence, with probability σ_1, the reduction makes it to Step 7 instead of outputting $\perp$.

Note that in Step 5, X and Y partition S_i, and each have size at least $\frac{1}{\ell}|S_i|$. Step 6 then removes a part of S_i with at least this size. Therefore, for all i, at the end of Step 6, we have that $|S_{i+1}| \leq \frac{\ell-1}{\ell}|S_i|$. Let $C = \mathrm{poly}(\log p)$ and $m = \lceil \frac{\log N_1 - \log C}{-\log \frac{\ell-1}{\ell}} \rceil$. Then

$$\begin{aligned}
\log |S_m| &\leq \log\left(\left(\frac{\ell-1}{\ell}\right)^m N_1\right) \\
&= m \log \frac{\ell-1}{\ell} + \log N_1 \\
&\approx \tfrac{\log N_1 - \log C}{-\log \frac{\ell-1}{\ell}} \log \frac{\ell-1}{\ell} + \log N_1 \\
&= \log C.
\end{aligned}$$

This implies that to ensure $|S_m| \leq \left(\frac{\ell-1}{\ell}\right)^m N_1$ has polynomial size, $O(m)$ calls to $\mathcal{O}$ are required. Therefore, we expect there to be at least $O(\log p)$ many iterations of Steps 2 to 7.

Step 4 performs a binary search using $\mathcal{O}$. The search for the minimum K takes $\log N_1$ calls. By the statement of Theorem 3.4.6, Step 6 will terminate with high probability, say σ_2, in worst case time $\tilde{O}(K^3)$. Therefore, since Steps 2 to 7 happen $O(\log p)$ many times, Algorithm 3.4.11 terminates in worst-case time $K^3 \cdot \text{poly}(\log p)$, and succeeds with probability $(\sigma_1\sigma_2)^{O(\log p)}$ which is non-negligible (by analysis similar to that in the proof of Theorem 3.3.5). □

Before we end this section, it is important to note (and will come up in Section 3.6 and the end of Section 3.5) that there exists a potential tradeoff for the endomorphisms output by Oracle 3.4.8. In particular, the tradeoff is between the proportion of the set S that is fixed and the number of queries to Oracle 3.4.8 in Algorithm 3.4.11. Consider the following modified oracle.

Oracle 3.4.12.
Input: *$p = N_1N_2 - 1$, a supersingular elliptic curve E, an integer K, a set $S \subset E[N_1]$, a basis P, Q for $E[N_1]$, and $\rho \in (0, \frac{1}{2}]$.*

Output: *The triangular kernel (K_0, K_1, K_2) of a cyclic endomorphism ψ of E such that the following constraints hold:*

1. *$|K_2| \leq K$,*
2. *$\gcd(|K_2|, N_1) = 1$,*
3. *$\left|S \cap \mathsf{Eig}_{N_1}^{P,Q}(\psi)\right| \geq \rho|S|$, and*
4. *$\left|E[N_1] \setminus \mathsf{Eig}_{N_1}^{P,Q}(\psi)\right| \geq \rho|S|$,*

or $\perp$ if no endomorphism satisfying these constraints exists.

Oracle 3.4.12 can be used similarly to Oracle 3.4.8, as seen in the following theorem.

Theorem 3.4.13. *Given*

1. *a starting supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p = N_1 N_2 - 1$ for coprime N_1 and $\log p$-smooth N_2,*
2. *the image of an N_1-degree isogeny $E' = \phi(E)$,*
3. *the action of ϕ on $E[N_2]$, and*
4. *access to an $\mathcal{O}$ as in Oracle 3.4.12, such that for an overwhelming fraction of sets S, $\mathcal{O}$ succeeds for a non-negligible proportion of $K \in \{0, \ldots, N_1\}$ and $\rho \in \left(0, \frac{1}{2}\right)$,*

there exists a (classical) algorithm which outputs $\ker\phi$ with non-negligible probability, makes $m = O\left(\frac{\log N_1}{-\log(1-\rho)}\right)$ queries to $\mathcal{O}$, and runs in worst-case time $\tilde{O}\left(K^3 \cdot m\right)$. Further, if the endomorphisms output from $\mathcal{O}$ all have $\log p$-smooth degree, then the runtime is $\tilde{O}\left(\sqrt{K} \cdot m\right)$.

Proof. Instead of rewriting Algorithm 3.4.11 with slight changes, we present only the steps which need modifying:

1. In Step 3 include ρ in the call to $\mathcal{O}$ (Oracle 3.4.12),
2. Replace Step 4 with:

```
While $\mathcal{O}$ outputs a solution:
    Halve $K$ and call $\mathcal{O}$.
Let $K$ be the last value where $\mathcal{O}$ did not output $\perp$.
While $\rho \leq 1/2$, and $\mathcal{O}$ outputs a solution:
    Double $\rho$ and call $\mathcal{O}$.
Let $\rho$ be the last value for which $\mathcal{O}$ did not output $\perp$.
Let $(K_0, K_1, K_2)$ be the triangular kernel output of $\mathcal{O}$ called with $p, E, K$, and $S_i$.
```

Similar analysis from the proof of Theorem 3.4.10 holds with $m = \left\lceil \frac{\log N_1 - \log C}{-\log(1-\rho)} \right\rceil$ for some $C = \mathrm{poly}(\log p)$, as at the end of Step 6, $|S_{i+1}| \leq (1-\rho)|S_i|$ for all i. □

The remainder of this chapter studies the feasibility of Oracle 3.4.8. In Section 3.5 we describe the set of endomorphisms that can be output by Oracle 3.4.8 for the isomorphism class of supersingular elliptic curves with j-invariant 1728. In Section 3.6 we give heuristic analyses for the existence of such endomorphisms in a multiparty setting (see [50, 4]).

3.5 Test Case: $j(E) = 1728$

The focus of this section will be to determine the difficulty of instantiating Oracle 3.4.8 on supersingular elliptic curves with j-invariant 1728. Eq. 3.6 will provide an if and only if for the criterion required for an output of Oracle 3.4.8. We begin with some background on such curves.

In any field with prime characteristic p satisfying $p \equiv 3 \mod 4$ it is well-known that elliptic curves with j-invariant 1728, those with short Weierstrass equation

$$E : y^2 = x^3 + ax$$

for $a \in \overline{\mathbb{F}}_p^*$, are supersingular. If E is supersingular with j-invariant 1728 over $\overline{\mathbb{F}}_p$, then the field contains some element $i \in \overline{\mathbb{F}}_p \backslash \mathbb{F}_p$ such that $i^2 = -1$, and further the elliptic curve has the two following endomorphisms:

$$\pi : E \to E,\ (x, y) \mapsto (x^p, y^p),$$
$$\iota : E \to E,\ (x, y) \mapsto (-x, i \cdot y).$$

The enodmorphism ring of E is isomorphic to the maximal order

$$\mathrm{End}(E) \cong \mathbb{Z}\left\langle 1, \iota, \frac{1+\pi}{2}, \frac{\iota + \iota \circ \pi}{2} \right\rangle$$

in the quaternion algebra ramified at p and ∞ [78, p 368–369].

We now consider the problem of instantiating Oracle 3.4.8 on this isomorphism class of elliptic curves. As Oracle 3.4.8 restricts the degree and the action of the output endomorphisms, these are the two problems we address. First, recall that the degree of an endomorphism is equal to the norm of the associated quaternion element. From this, we see the following.

Lemma 3.5.1. *Let $E/\overline{\mathbb{F}}_p$ be a supersingular elliptic curve with j-invariant 1728. For variables $w, x, y, z \in \mathbb{Z}$, let ψ be the endomorphism*

$$\psi = [w] + [x]\iota + [y]\frac{1+\pi}{2} + [z]\frac{\iota + \iota \circ \pi}{2}.$$

Then,

$$\deg(\psi) = w^2 + x^2 + \left(\frac{p+1}{4}\right)(y^2 + z^2) + wy + xz.$$

Proof. This is a straightforward calculation of the quaternion norm:

$$\begin{aligned}[\deg \psi] &= \psi \circ \widehat{\psi} \\ &= \left([w] + [x]\iota + [y]\left(\frac{1+\pi}{2}\right) + [z]\left(\frac{\iota + \iota\circ\pi}{2}\right)\right) \\ &\quad \cdot \left([w] - [x]\iota + [y]\left(\frac{1-\pi}{2}\right) + [z]\left(\frac{-\iota - \iota\circ\pi}{2}\right)\right) \\ &= [w^2] + [wy] + [x^2] + [xz] + \left(\frac{p+1}{4}\right)([y^2] + [z^2]) \\ &= \left[w^2 + wy + x^2 + xz + \left(\frac{p+1}{4}\right)(y^2 + z^2)\right],\end{aligned}$$

where the second to last equality follows from the fact that the maps corresponding to coefficients $[yz]$ and $[zy]$ are negatives of one another. □

Next, we discuss the computational cost of determining the action of a fixed endomorphism.

Lemma 3.5.2. *Let N be a natural number, and $E/\overline{\mathbb{F}}_p$ a supersingular elliptic curve with $\mathrm{End}(E) \cong \mathbb{Z}\langle b_1, b_2, b_3, b_4\rangle$. For variables $w, x, y, z \in \mathbb{Z}$, the action of any endomorphism*

$$[w]b_1 + [x]b_2 + [y]b_3 + [z]b_4$$

on $E[N]$ can be written as a 2×2-matrix $M(w, x, y, z)$ whose entries are linear in the four variables. In the worst case, this can be done in $\tilde{O}(N^3)$ time, and in $\Omega(\log^2 p)$ when N is $\log p$-smooth.

Proof. We prove Lemma 3.5.2 by describing an algorithm that returns the required output and analyzing its runtime. Consider the factorization of $N = \prod_{1\le i\le r} q_i^{e_i}$.

Algorithm 3.5.3.
Input: *A supersingular elliptic curve E, a basis $\{b_1, b_2, b_3, b_4\}$ of $\mathrm{End}(E)$, $w, x, y, z \in \mathbb{Z}$, $N \in \mathbb{Z}$, and optionally a basis $\{P, Q\}$ for $E[N]$.*

Output: *A 2×2-matrix $M(w, x, y, z)$ whose entries are linear in the four variables and a basis $\{P, Q\}$ for $E[N]$ if it was not provided.*

```
1. If $P, Q$ is not given, find a basis $\{P, Q\}$ of $E[N] \subset E/\overline{\mathbb{F}}_{p^2}$.
```

2. *Calculate $b_i(P)$ and $b_i(Q)$ for $i = 1, \dots, 4$. Solving the discrete logarithm for these values, in terms of P and Q, gives the action of b_i on $E[N]$ which we can write as a matrix M_i.*

3. *Calculate $M = wM_1 + xM_2 + yM_3 + zM_4$, where w, x, y, z are integer variables.*

4. *Return P, Q, M.*

The most difficult part of Algorithm 3.5.3 is constructing the field extension in Step 1. Once the extension is constructed, the arithmetic in that extension is efficient in N and p. Recall that Step 1 has runtime $\tilde{O}(N^3)$ in the case where $E[N]$ is not defined over $\mathbb{F}_{p^2}$, as seen in the proof of Lemma 3.2.4. The best-case scenario for constructing the basis in Step 1 takes $\Omega(\log^2 p)$ when N is $\log p$-smooth or the N-torsion is defined over a small field extension (see Lemma 3.2.4).

Step 2 has runtime $O\Big(\sum_{1\leq i\leq r} e_i(\log N + \sqrt{q_i})\Big)$[79], which is always less than the runtime in Step 1. □

With Algorithm 3.5.3 we see how to compute the action of a generic basis of $\mathrm{End}(E)$ on an arbitrary torsion subgroup. When applying this to supersingular elliptic curves with j-invariant 1728, certain endomorphisms can have their action on certain torsion subgroups represented by 2×2 matrices whose entries are linear in only two variables. Indeed, consider the subring (and rank 4-submodule)

$$\mathcal{O}_0 \cong \mathbb{Z}\left\langle 1, \iota, \pi, \iota\circ\pi \right\rangle \subset \mathbb{Z}\left\langle 1, \iota, \frac{1+\pi}{2}, \frac{\iota+\iota\circ\pi}{2} \right\rangle.$$

Then for any $\psi = [w] + [x]\iota + [y]\pi + [z]\iota\circ\pi \in \mathcal{O}_0$, the norm equation of the subring $\mathcal{O}_0$ gives

$$\deg(\psi) = w^2 + x^2 + p\left(y^2 + z^2\right).$$

Additionally, let $P, Q \in E[N]$ be a basis for some integer N, and suppose that $Q = \iota(P) = \pi(P)$ (this assumption will be studied in Appendix B). Then a simple calculation shows that the matrix computed in Algorithm 3.5.3 on the input $E, \{1, \iota, \pi, \iota\circ\pi\}, w, x, y, z, N$, and the basis P, Q, is

$$\begin{bmatrix} w-z & -x+y \\ x+y & w+z \end{bmatrix}.$$

If, instead, the entire endomorphism ring is used, then the action of $\psi = [w] + [x]\iota + [y]\dfrac{1+\pi}{2} + [z]\dfrac{\iota+\iota\circ\pi}{2}$ is slightly more involved. Again, if we let $P, Q \in E[N]$ be a basis for some integer N, and suppose that $Q = \iota(P) = \pi(P)$ (again, see Appendix B), then the matrix computed in Algorithm 3.5.3 on the input $E, \left\{1, \iota, \dfrac{1+\pi}{2}, \dfrac{\iota + \iota \circ \pi}{2}\right\}, w, x, y, z, N,$ and the basis P, Q, is

$$\begin{bmatrix} w + \frac{y-z}{2} & -x + \frac{y-z}{2} \\ x + \frac{y+z}{2} & w + \frac{y+z}{2} \end{bmatrix}.$$

We now examine when a matrix of this above form will satisfy constraints 3 and 4 of Oracle 3.4.8. Lemma 3.5.5 characterizes the eigenvectors of this matrix of the form $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ when N is a prime power. In particular, it calculates the number of eigenvectors of that form (which is the value we need to bound from below for the constraints of Oracle 3.4.8 to be satisfied).

Lemma 3.5.4. *Let M be the matrix*

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

over the ring $\mathbb{Z}/\ell^e\mathbb{Z}$. Then $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ is an eigenvector of M if and only if $-br^2 + (d-a)r + c \equiv 0 \mod \ell^e$.

Proof. If $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ is an eigenvector, then there exists some $\lambda \in \mathbb{Z}/\ell^e\mathbb{Z}$ such that the following equations hold:

$$a + rb = \lambda, \quad \text{and } c + rd = \lambda r.$$

Substituting gives

$$c + rd \equiv \lambda r \equiv ar + br^2 \mod \ell^e,$$

and the forward direction follows.

Conversely, suppose $-br^2 + (d-a)r + c \equiv 0 \mod \ell^e$. Then, $dr + c \equiv br^2 + ar \mod \ell^e$, and so

$$\left[\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right] \left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right] = \left[\begin{smallmatrix} br+a \\ dr+c \end{smallmatrix}\right] = \left[\begin{smallmatrix} br+a \\ ar+br^2 \end{smallmatrix}\right] = (br+a) \left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right].$$

So $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ is an eigenvector of M. □

Lemma 3.5.5. *Let M be the matrix*

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

over the ring $\mathbb{Z}/\ell^e\mathbb{Z}$ and suppose it has the eigenvector $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$. Let

- *ν denote the largest natural number such that $\ell^\nu \mid b$, $\ell^\nu \mid d - a$, and $\nu \leq e$,*

- *α, β denote the numbers such that $b = \ell^\nu \beta$ and $d - a = \ell^\nu \alpha$,*

- *ϵ denote the largest natural number such that $\ell^\epsilon \mid \alpha - 2\beta r$ and $\epsilon \leq \frac{e-\nu}{2}$, and*

- *$\gamma = e - \nu - \epsilon$.*

With these notations,

- *if $\ell \mid \beta$, then $\left[\begin{smallmatrix}1\\x\end{smallmatrix}\right]$ is an eigenvector of M if and only if $x = r + y\ell^\gamma$ for some $y \in \mathbb{Z}/\ell^e\mathbb{Z}$, and*

- *if $\ell \nmid \beta$, then $\left[\begin{smallmatrix}1\\x\end{smallmatrix}\right]$ is an eigenvector of M if and only if $x = r + y\ell^\gamma$ or $x = -r + \alpha\beta^{-1} + y\ell^\gamma$ for some $y \in \mathbb{Z}/\ell^e\mathbb{Z}$.*

Proof. By Lemma 3.5.4, $\left[\begin{smallmatrix}1\\z\end{smallmatrix}\right]$ is an eigenvector of M if and only if $-bz^2 + (d-a)z + c \equiv 0 \mod \ell^e$. Suppose that $\left[\begin{smallmatrix}1\\r+z\end{smallmatrix}\right]$ is an eigenvector of M. Then,

$$-b(r+z)^2 + (d-a)(r+z) + c \equiv 0 \mod \ell^e. \tag{3.2}$$

Since $\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right]$ is an eigenvector of M, we know that

$$-br^2 + (d-a)r + c \equiv 0 \mod \ell^e,$$

and so we can subtract this from Eq. (3.2) to conclude that $\left[\begin{smallmatrix}1\\r+z\end{smallmatrix}\right]$ is an eigenvector of M if and only if

$$(d-a)z - b(2rz + z^2) \equiv 0 \mod \ell^e.$$

Equivalently, $\left[\begin{smallmatrix}1\\r+z\end{smallmatrix}\right]$ is an eigenvector of M if and only if

$$z(\alpha - \beta(2r + z)) \equiv 0 \mod \ell^{e-\nu}. \tag{3.3}$$

Suppose that $\ell \mid \beta$. Then $\ell \nmid \alpha$, and so the vector $\left[\begin{smallmatrix}1\\r+z\end{smallmatrix}\right]$ is an eigenvector of M if and only if $z \equiv 0 \mod \ell^{e-\nu}$. Equivalently, since $\epsilon = 0$, $\left[\begin{smallmatrix}1\\x\end{smallmatrix}\right]$ is an eigenvector of M if and only if it is of the form $\left[\begin{smallmatrix}1\\r+y\ell^\gamma\end{smallmatrix}\right]$, for some $y \in \mathbb{Z}/\ell^e\mathbb{Z}$.

Suppose next that $\ell \nmid \beta$. Then Eq. (3.3) can be rewritten as

$$z(\alpha\beta^{-1} - 2r - z) \equiv 0 \mod \ell^{e-\nu}. \tag{3.4}$$

If z satisfies Eq. (3.4), then one of two cases holds:

$$\begin{aligned} &z \equiv \alpha\beta^{-1} - 2r \mod \ell^\gamma, \text{ and} \\ &z \equiv 0 \mod \ell^\epsilon, \end{aligned}$$

or

$$\begin{aligned} z &\equiv 0 \mod \ell^{\gamma}, \quad \text{and} \\ z &\equiv \alpha\beta^{-1} - 2r \mod \ell^{\epsilon}. \end{aligned}$$

Therefore, $\left[\begin{smallmatrix}1\\x\end{smallmatrix}\right]$ is an eigenvector of M if it is of the form $\left[\begin{smallmatrix}1\\r+y\ell^{\gamma}\end{smallmatrix}\right]$ or $\left[\begin{smallmatrix}1\\-r+\alpha\beta^{-1}+y\ell^{\gamma}\end{smallmatrix}\right]$, for some $y \in \mathbb{Z}/\ell^e\mathbb{Z}$.

Conversely, suppose still that $\ell \nmid b$, and that $x = r + y\ell^{\gamma}$ or $x = -r + \alpha\beta^{-1} + y\ell^{\gamma}$, for some $y \in \mathbb{Z}/\ell^e\mathbb{Z}$. It is straightforward to show that in each case the vector $\left[\begin{smallmatrix}1\\x\end{smallmatrix}\right]$ is an eigenvector of M from the definitions of α, β, and γ. □

Corollary 3.5.6. *Let M be the matrix*

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

over the ring $\mathbb{Z}/\ell^e\mathbb{Z}$. Then, using the notation of Lemma 3.5.5, the number of ℓ^e-eigenvectors of M is 0, $\ell^{e-\gamma}$ or $2\ell^{e-\gamma}$.

We now examine conditions 3 and 4 of Oracle 3.4.8 under the above assumptions.

Theorem 3.5.7. *Let E be a supersingular elliptic curve with $j(E) = 1728$, let $N_1 = \ell^e$, and let $S = E^{P,Q}[\ell^e]$. If $\psi = [w] + [x]\iota + [y]\dfrac{1+\pi}{2} + [z]\dfrac{\iota + \iota\circ\pi}{2} \in \mathrm{End}(E)$ satisfies the output conditions of Oracle 3.4.8, then*

$$\ell^{e-2} \mid x, \;\; \ell^{e-2} \mid y, \;\; \textit{and } \ell^{e-2} \mid z,$$

or if $\ell = 2$, then

$$\ell^{e-2} \mid 2x, \;\; \ell^{e-2} \mid y, \;\; \textit{and } \ell^{e-2} \mid z.$$

Proof. Let

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

be the matrix corresponding to the action of ψ on $E[\ell^e]$, and let ν, ϵ, and γ be as in Lemma 3.5.5. Consider the conditions from Oracle 3.4.8:

$$\left|\mathsf{Eig}_{\ell^e}^{P,Q}(\psi)\right| \geq \frac{1}{\ell}|S|, \;\text{and}\; \left|E^{P,Q}[\ell^e] \setminus \mathsf{Eig}_{\ell^e}^{P,Q}(\psi)\right| \geq \frac{1}{\ell}|S|.$$

Following Corollary 3.5.6, we need

$$\ell^{e-\gamma} \geq \frac{1}{\ell}|S| = \ell^{e-1}.$$

Therefore, $e - \gamma \geq e - 1$ must be satisfied. Substituting the bound of $\epsilon \leq \dfrac{e-\nu}{2}$ into the definition of γ we see that $e-2 \leq \nu$ and $\epsilon \leq 1$. Hence, for conditions 3 and 4 of Oracle 3.4.8 to be satisfied, we need $e - 2 \leq \nu$.

Since M must have at least one eigenvector for the condition 3 to be satisfied (as this was used by assumption in Lemma 3.5.5), by Lemma 3.5.4 there must exist r such that

$$-br^2 + (d-a)r + c \equiv 0 \mod \ell^e. \tag{3.5}$$

By definition of ν, we also see that $\ell^{e-2} \mid b, d-a$. Combining these modular restrictions with Eq. 3.5 , we see that $\ell^{e-2} \mid c$ as well. Recall from the discussion after Lemma 3.5.2 that

$$M = \begin{bmatrix} w + \frac{y-z}{2} & -x + \frac{y-z}{2} \\ x + \frac{y+z}{2} & w + \frac{y+z}{2} \end{bmatrix}.$$

Thus

$$\ell^{e-2} \mid -x + \tfrac{y-z}{2},\ \ \ell^{e-2} \mid z,\ \text{ and } \ell^{e-2} \mid x + \tfrac{y+z}{2},$$

and the statement follows. □

Finally, we combine Theorem 3.5.7 with the degree equation (see Lemma 3.5.1). Recall $p = N_1 N_2 - 1$. In order for an endomorphism $\psi = [w] + [x]\iota + [y]\dfrac{1+\pi}{2} + [z]\dfrac{\iota + \iota \circ \pi}{2}$ to satisfy the conditions of Oracle 3.4.8, we must have some integers $\tilde{x}, \tilde{y}, \tilde{z}$ such that $x = \ell^{e-2}\tilde{x}, y = \ell^{e-2}\tilde{y}$, and $z = \ell^{e-2}\tilde{z}$. Furthermore, the following equation must hold:

$$w^2 + \ell^{2e-4}\tilde{x}^2 + \ell^{2e-4}\left(\tfrac{p+1}{4}\right)\left(\tilde{y}^2 + \tilde{z}^2\right) + \ell^{e-2}w\tilde{y} + \ell^{2e-4}\tilde{x}\tilde{z} = k, \tag{3.6}$$

where, if g is the greatest integer such that $g \mid k$ and $g \mid \ell^{2e}$, then $k' := \frac{k}{g} < K$ and $\gcd(k', \ell) = 1$.

3.6 Cryptanalysis of Multi-Party Schemes

Section 3.5 turns an algebraic problem—do special endomorphisms exist for an isomorphism class of elliptic curves—into an arithmetic problem—do there exist solutions to

Eq. 3.6. A useful quality of this reformation is that there are clear lower bounds to solutions of quadratic equations. In this section we study Eq. 3.6 for different values of the right-hand side k. These different values of k correspond to a different number of participants in a multi-party variant of SIDH [50, 4].

We begin by presenting such a lower bound for the quadratic equation in Eq. 3.6 and relate it to endomorphisms.

Lemma 3.6.1. *Assume the following:*

1. $E(\mathbb{F}_{p^2})$ *has* j*-invariant* 1728*, where* $p = N_1 \cdots N_r - 1$*,*
2. $\forall i, j \in \{1, \dots, r\}$ *we have* $N_i \approx N_j$*, and* $N_1 = \ell^e$*,*
3. $\psi = [w_0] + [x_0]\iota + [y_0]\frac{1+\pi}{2} + [z_0]\frac{\iota+\iota\pi}{2}$ *is an endomorphism on* E *of degree* k*, for integers* w_0, x_0, y_0 *and* z_0*, where if* g *is the greatest integer such that* $g \mid (N_2 \cdots N_r)^2$ *and* $g \mid k$*, then* $k' := \frac{k}{g}$ *satisfies* $\gcd(k', \ell) = 1$*,*
4. *at least one of* y_0 *and* z_0 *is nonzero, and*
5. $\ell^\mu \mid x_0, y_0, z_0$ *for a positive integer* μ*.*

Then k' *has a lower bound of approximately* $\ell^{2(\mu+e)-re}$*.*

Proof. If the point $(w_0, x_0, y_0, z_0) = (w, x, y, z)$ is a solution to

$$w^2 + x^2 + \left(\tfrac{p+1}{4}\right)(y^2 + z^2) + wy + xz = gk',$$

then $(w_1, x_1, y_1, z_1) = (2w_0 + y_0, 2x_0 + z_0, y_0, z_0)$ is a solution to

$$w^2 + x^2 + p(y^2 + z^2) = 4gk'.$$

As $\ell^\mu \mid x_1, y_1, z_1$, we can let $(w_1, x_1, y_1, z_1) = (w_2, \ell^\mu x_2, \ell^\mu y_2, \ell^\mu z_2)$. Then (w_2, x_2, y_2, z_2) is a solution to

$$w^2 + \ell^{2\mu} x^2 + \ell^{2\mu} p(y^2 + z^2) = 4gk'.$$

Following the transformations above, since at least one of y_0 and z_0 is non-zero, at least one of y_2 or z_2 is non-zero. Then,

$$k' \geq \tfrac{\ell^{2\mu} p}{4g} \approx \ell^{2(\mu+e)-re},$$

since $p \approx \ell^{re}$ and $g \approx \ell^{2(r-1)e}$. □

We now recall constraint 1 of Oracle 3.4.8. As the runtime of the algorithms in Theorems 3.4.9 and 3.4.10 depends on the bound K, which corresponds to k' in Eq. 3.6, we need to examine the different values of the lower bound for k' and compare these to the runtimes of the best known kernel-finding algorithms.

Proposition 3.6.2. *The algorithm in Theorem 3.4.9 in the r-party case has a lower bound on its best-case runtime of $\tilde{O}\left(\ell^{\frac{(4-r)e}{2}-2}\right)$.*

Proof. Consider an endomorphism $\psi = [w_0] + [x_0]\iota + [y_0]\frac{1+\pi}{2} + [z_0]\frac{\iota+\iota\pi}{2}$ on E of degree k, where if g is the greatest integer such that $g \mid (N_2 \cdots N_r)^2$ and $g \mid k$, then $k' := \frac{k}{g}$ satisfies $\gcd(k', \ell) = 1$. If there is a convenient basis $\{P, Q\}$ of $E[\ell]$, then ψ acts as the matrix M_0 with respect to $\{P, Q\}$. Let ν, ϵ, and γ be defined as in Lemma 3.5.5 with respect to M_0, and let μ be defined as in Lemma 3.6.1 (that is, $\ell^\mu \mid x_0, y_0, z_0$).

By Theorem 3.5.7, $\mu \geq e - 2$. Then, by Lemma 3.6.1, $k' \geq \ell^{2(e-2+e)-re} = \ell^{(4-r)e-4}$. The theorem statement then follows from the best-case runtime in Theorem 3.4.9. □

Corollary 3.6.3. *The algorithm in Theorem 3.4.9 in the 2-party case has a lower bound on its best-case runtime of $\tilde{O}\left(\ell^{e-2}\right)$. Similarly, in the 3-party case it has a lower bound on its best-case runtime of $\tilde{O}\left(\ell^{\frac{e}{2}-2}\right)$. No lower bound is imposed by this work on the runtime in the r-party case for $r \geq 4$.*

In both of these cases, the algorithm in Theorem 3.4.9 is worse than the best known-algorithms for SSDDH. However, the algorithm can be improved. We now show that the value $\frac{1}{\ell}$ in constraints 3 and 4 of Oracle 3.4.8 forms a tradeoff with the value of k', which can be utilized to achieve a faster algorithm in these cases. We now study the runtime of the algorithm in Theorem 3.4.13 in the multi-party setting. As constraints 3 and 4 of Oracle 3.4.12 do not impose such strict size requirements on the partition of S, we have more freedom to tradeoff between k' and the number m of oracle calls. We start with a lemma analogous to Theorem 3.5.7.

Lemma 3.6.4. *Suppose the matrix*

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} w + \frac{y-z}{2} & -x + \frac{y-z}{2} \\ x + \frac{y+z}{2} & w + \frac{y+z}{2} \end{bmatrix}$$

has at least one eigenvector of the form $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$. Then $\ell^\nu \mid x, y, z$ if and only if $\ell^\nu \mid b, d - a$.

Proof. The forward direction is straightforward. Conversely, if $\ell^\nu \mid d-a$, then $\ell^\nu \mid z$. Since $\ell^\nu \mid b$, showing that $\ell^\nu \mid c$ would yield $\ell^\nu \mid x, y$. Since M has an eigenvector, by Lemma 3.5.4 we have

$$br^2 + (a-d)r - c \equiv 0 \bmod \ell^e.$$

Then, since $\ell^\nu \mid b, d-a$, we see that $\ell^\nu \mid c$. □

Proposition 3.6.5. *The algorithm in Theorem 3.4.13 in the r-party case has a lower bound on its best-case runtime of $\tilde{O}\left(\ell^{\frac{(3-r)e}{2}}\right)$.*

Proof. Let

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} w + \frac{y-z}{2} & -x + \frac{y-z}{2} \\ x + \frac{y+z}{2} & w + \frac{y+z}{2} \end{bmatrix}$$

be the matrix representation of the action of some $\psi \in \mathrm{End}(E)$ returned by Oracle 3.4.12, and let ν and ϵ be as in Lemma 3.5.5 with respect to M.

Increasing ν increases the runtime (see Theorem 3.5.7), but this is not true of ϵ. The optimal value for ϵ is therefore $\frac{e-\nu}{2}$ (as this increases the eigenspace with no additional cost to the algorithm). Then, by Corollary 3.5.6, there are at least $\ell^{\nu+\epsilon} = \ell^{\frac{e+\nu}{2}}$ eigenvectors (there cannot be 0 eigenvectors, by the constraints on the output of Oracle 3.4.12). From this we see that the expected number of calls to Oracle 3.4.12 before a single kernel-fixing endomorphism is found is $\frac{1}{2}\ell^{\frac{e-\nu}{2}}$. Hence, if m is the expected number of calls to the oracle (as in Theorem 3.4.13), we have $m \geq \frac{1}{2}\ell^{\frac{e-\nu}{2}}$.

By the definition of ν in Lemma 3.5.5, $\ell^\nu \mid b, d-a$. By Lemma 3.6.1, $\ell^\nu \mid x, y, z$. Then, by Lemma 3.6.4, k is at least $\ell^{2(\nu+e)-re} = \ell^{2\nu+2e-re}$. Thus the expected runtime for the algorithm in Theorem 3.4.13 has the lower bound

$$m\sqrt{k} \geq \ell^{\frac{e-\nu}{2}}\sqrt{k} \geq \ell^{\frac{e-\nu}{2}}\ell^{\nu+e-\frac{re}{2}} = \ell^{\frac{(3-r)e+\nu}{2}}.$$

This is optimal when $\nu = 0$. □

Corollary 3.6.6. *The algorithm in Theorem 3.4.13 in the 2-party case has a lower bound on its best-case runtime of $\tilde{O}\left(\ell^{\frac{e}{2}}\right)$. Similarly, in the 3-party case it has a lower bound on its best-case runtime of $\tilde{O}\left(\ell^{\frac{e}{4}}\right)$.*

Proof. The runtime in the 2-party case can be obtained by substituting $r = 2$ into Proposition 3.6.5. When analyzing the runtime for the 3-party case, we must note that the lower bound of k' from Lemma 3.6.1 is actually $\max\{1, \ell^{2(\mu+e)-re}\}$, where $r = 3$. Then the runtime of the algorithm in Theorem 3.4.13 in the best case is

$$m\sqrt{k} \geq \ell^{\frac{e-\nu}{2}} \sqrt{\max\{1, \ell^{2(\nu+e)-3e}\}} = \max\{\ell^{\frac{e-\nu}{2}}, \ell^{\frac{\nu}{2}}\}.$$

This is optimal when $\nu = \frac{e}{2}$, which gives the runtime $\tilde{O}\left(\ell^{\frac{e}{4}}\right)$. □

Chapter 4

Security of Static-Key SIDH

Portions of this chapter were published in Azarderakhsh, Jao, Leonardi [6]. The contents of this chapter represent my contribution to that work.

Some key agreement protocols leak information about secret keys if dishonest participants use malformed public keys. We formalize these protocols and attacks, and present a generic transformation that can be made to such key agreement protocols to resist such attacks. In the transformed protocol each party generates k different keys and the two parties perform key agreement using all k^2 combinations of their individual keys. We consider this transformation in the context of various post-quantum key agreement schemes and analyze the attacker's success probabilities (which decrease exponentially in k and depend on the details of the underlying key agreement protocol) to determine the necessary parameter sizes for 128-bit security. Our transformation increases key sizes by a factor of k and computation times by k^2, which represents a significant cost—but nevertheless still feasible as k will be logarithmic in the security parameter. Our transformation is particularly well-suited to supersingular isogeny Diffie-Hellman, in which one can take $k = 113$ instead of the usual $k = 256$ at the 128-bit quantum security level.

This work initiated a potential path forward towards solving the open problem of securing long-term static-static key exchange against quantum adversaries. Additional work on this subject has been performed by other authors. To give a complete picture of the state of this topic we discuss here a summary of that work.

After the publication of this work in the International Conference on Selected Areas in Cryptography 2017 [6], Jao and Urbanik [94] published an impovement at MathCrypt 2018. The nature of this improvement is to take advantage of the non-trivial automorphisms of the supersingular elliptic curves with j-invariants 0 and 1728 to the effect of decreasing the value of k in the k-SIDH protocol. The end result is a lower value of k needed for the participants, $k = 18$ is proposed by the authors for $\ell = 11$, resulting in a more efficient scheme. The Jao-Urbanik scheme still retains security against the form of attack discussed in this chapter, namely active attacks using malicious public keys as in [53].

Dobson et al. published an adaptive attack to totally break the 2-SIDH protocol [44], that is k-SIDH when $k = 2$, using the same techniques as [53]. The attack can be implemented against the generic k-SIDH protocol, but as shown in this chapter, the complexity of that algorithm grows exponentially with k and is not deployable for the values of k proposed here.

Most recently, Basso et al. [8] describe an adaptive and active attack on the Jao-Urbanik variant of k-SIDH, adapting the previous attacks [53, 44] for this variant protocol. The authors of [8] are able to reduce the number of queries needed in these active attacks, achieving an algorithm which requires $O(32^{k/3})$ queries to break the Jao-Urbanik variant. The result of the work in [8] is that the k-SIDH protocol described in this chapter and [6] is more efficient at the same level of security when compared to the Jao-Urbanik variant.

4.1 Introduction

In Asiacrypt 2016, Galbraith, Petit, Shani, and Ti [53] introduced an active attack against the supersingular isogeny-based cryptosystem of De Feo, Jao, and Plût [47], which circumvents all extant (at the time) direct validation techniques. The attack allows an adversary who interacts with a static key over multiple rounds of key exchange to efficiently compute the private key corresponding to the static key over multiple sessions. When communicating, the participants in an SIDH key exchange each publish a supersingular elliptic curve and two points on the curve. By manipulating the values of the two points, the attacker can learn one bit of information about the other participant's private key (depending on whether or not the key exchange operation succeeds using the manipulated points), and then repeat this process over additional sessions to learn additional private key bits. As stated in [53], a countermeasure to their attack was already available in the earlier work of Kirkwood et al. [63], who proposed so-called "indirect key validation" using a Fujisaki-Okamoto type transform [49] in order to allow the honest participant to detect whether or not the other party is manipulating points. Unfortunately, this countermeasure requires the untrusted party to disclose their SIDH private key to the verifier, precluding the use of SIDH as a drop-in replacement for Diffie-Hellman or other protocols that support static-static key exchange using direct key validation.

Although [53] specifically targets SIDH, similar attacks apply against all other post-quantum encryption schemes submitted to NIST. No currently known post-quantum scheme achieves secure static-static key exchange without the use of ephemeral keys or indirect validation techniques that would expose one's key in the static-static setting. Major lattice-based key establishment schemes such as "A New Hope" [2] and "Frodo" [15] achieve only passive security and are intended and designed to be used with ephemeral keys. Peikert's Ring-LWE based scheme [74] is a key encapsulation mechanism that uses a Fujisaki-Okamoto type transform to achieve IND-CCA security [74, §5]. In Peikert's scheme, the encrypting participant must reveal their random coins to the decrypting participant, and so one member must use an ephemeral key. The Module-LWE key exchange Kyber [16, §5] has at least one party using an ephemeral key, and both parties using both a static and ephemeral key in the authenticated variant. In Niederreiter hybrid encryption [97, §3.1], the error vector is revealed and used to derive the shared symmetric key. Similarly, in McEliece encryption [73], although the error vector is not explicitly used in decryption, it is trivial to compute once the message is determined, and therefore one party must use an ephemeral key.

In this work we present a new generic transformation that takes any key establishment protocol satisfying certain security properties (see Definition 4.2.3) and converts it into a different protocol that is immune to attacks of the form presented in [53]. In our transformation, each party generates k different key pairs and publishes for their public key the list of k individual public keys. During key agreement, two parties compute k^2 different shared secrets obtained by performing shared key agreement with each of their keys in all possible combinations, and use a key derivation function on the shared secrets to derive a final shared key. Under this scheme, any use of an invalid public key will, with all but negligible probability, cause at least one of the k^2 shared secret computations to fail, which neutralizes the attack of [53]. Moreover, the number of possible failure outcomes is exponential in k, making it impossible for an attacker to predict a likely failure outcome in advance and lie about the value of their final shared key in order to salvage the attack of [53].

The necessary value of k depends on the details of the original protocol with which we started. The easiest (and worst) case is where each invalid key attempt in the original protocol leads to one of two possible (invalid) shared secret computations on the part of the honest party, depending on the value of one of the bits in the honest party's private key. In this case, one simply needs $k \approx \ell$ to achieve ℓ-bit classical security, and $k \approx 2\ell$ in the quantum case to account for Grover search. However, if there are more possible invalid outcomes, then the attacker's job is harder, and (as a designer) we can use a smaller value of k while still achieving ℓ-bit security. For example in Section 4.4 we perform a detailed analysis of SIDH and conclude that a value of $k = 113$ is sufficient to achieve 128-bit quantum security. While a key size penalty of a factor of $O(\ell)$ and performance penalty of a factor of $O(\ell^2)$ might seem untenable, we point out that our scheme is far from the worst in this regard compared to some recently published articles such as [10].

In Section 4.2 we present our security theorem which states that, for SIDH and other suitable protocols, our transformation is secure in the sense that finding even a single invalid key resulting in a successful key exchange (in the sense that the attacker can guess the shared secret computed by the honest party under this invalid key) is equivalent to breaking the passive security of the original un-transformed protocol. We recognize and emphasize that our security reduction falls short of a full proof of active security, as it only shows that attacks of the type that involve feeding an honest party invalid keys must fail, and not that arbitrary attacks must fail.

4.2 Multiple Instances of Key Establishment

We begin with a review of the format for key agreement protocols. The content of this paper focuses on two participants establishing a shared secret key that depends on inputs from both members; it does not address authentication.

Definition 4.2.1. *We let* $\boldsymbol{KE}$ *be a key establishment function (the requirements of which will be stated shortly). A key agreement protocol,* $\boldsymbol{KA}$*, for Alice and Bob using* $\boldsymbol{KE}$ *consists of the phases:*

0. ***Setup**: Both members obtain a valid copy of the global parameters, gp.*

1. ***Key Generation**: Alice generates a secret key* s_A *and public key* p_A*, likewise Bob generates* s_B *and* p_B*.*

2. ***Communication**: Alice obtains* p_B *and Bob obtains* p_A*.*

3. ***Key Establishment**: Alice computes* $\boldsymbol{KE}(gp, p_B, s_A)$ *and likewise Bob computes* $\boldsymbol{KE}(gp, p_A, s_B)$*.*

4. ***Verification**: If applicable, each participant test the validity of the others public key. Alice and Bob verify that they have computed the same shared secret. If they have not, communication is terminated.*

For the verification step to succeed, clearly the key establishment function $\boldsymbol{KE}$ *has the requirement that these two outputs are equal when the participants operate honestly. Additionally, the following values must be computationally infeasible to compute: a secret key from its corresponding public key, a secret key* s *from* $\boldsymbol{KE}(gp, p, s)$*, and* $\boldsymbol{KE}(gp, p_B, s_A)$ *from* gp, p_B*, and* p_A*.*

Note, this protocol is incomplete as it does not state how Alice and Bob check if they computed the same secret in the verification phase. However this step of the protocol will become explicit below, and the security of our choice will be examined in detail. We now formally state and analyze the security of performing multiple simultaneous instances of key agreement. First is the attack model that will be used throughout.

Definition 4.2.2. *Consider the attack model on a key agreement protocol where Bob may use a specially chosen public key/private key* (p_B, s_B) *and additionally act dishonestly in the verification phase.*

Following [53, §3] we define a two types of oracles that we will consider Bob having access to once per verification phase:

1. $Oracle_1(p_B) = \boldsymbol{KE}(gp, p_B, s_A)$*, which corresponds to Bob somehow obtaining the output of Alice's key establishment function.*
2. $Oracle_2(p_B, h')$ *returns* 1 *if* $h' = \boldsymbol{KE}(gp, p_B, s_A)$*, and returns* 0 *otherwise, which corresponds to Alice either terminating or continuing a session after she and Bob performed verification in which Bob used some* h' *as his secret.*

Suppose Bob chooses p_B *in such a way that a response from a type* (1) *oracle, or a response of* 1 *from a type* (2) *oracle, will reveal* $\kappa(p_B)$ *bits of Alice's secret key to Bob (where* $\kappa(\cdot)$ *returns non-negative integers). Then the output of* $Oracle_1(p_B)$ *follows some discrete probability distribution (as those* $\kappa(p_B)$ *bits vary); denote the corresponding probability mass function by* $\chi_{KE}(p_B, \cdot)$*, where the second input domain is the space of ciphertexts. Likewise for the type* (2) *oracle, let* $\chi_{KE}(p_B, h')$ *denote the probability that* $Oracle_2(p_B, h') = 1$*, where the probability is taken over the space of ciphertexts in the second input domain.*

In protocols where these attacks apply, a malicious Bob will typically know the distribution $\chi_{KE}(p_B, \cdot)$ (loosely speaking, if p_B is "close" to the actual public key derived by s_B, then $\mathbf{KE}(gp, p_B, s_A)$ will be "close" to $\mathbf{KE}(gp, p_A, s_B)$). Then Bob can use the values of h' for which $\chi_{KE}(p_B, h') > 0$ and have Alice respond as a type (2) oracle during verification which reveals those $\kappa(p_B)$ bits of her private key when he guesses h' correctly. Our goal is to modify key agreements susceptible to such attacks so that we can bound all probabilities in $\chi_{KE}(\cdot)$ arbitrarily from above. We first need to define a specific type of key agreement protocol.

Definition 4.2.3. *Let $\boldsymbol{KA}$ be a key agreement protocol which uses the key establishment function $\boldsymbol{KE}(gp, \cdot, \cdot)$, for some global parameters gp. If Bob has a public key/secret key pair (p_B, s_B) for $\boldsymbol{KA}$ and is given two public keys p_1 and p_2 (derived from some secret keys s_1, s_2 which are unknown to Bob), then $\boldsymbol{KE}(gp, p_B, s_1) = \boldsymbol{KE}(gp, p_1, s_B)$ and $\boldsymbol{KE}(gp, p_B, s_2) = \boldsymbol{KE}(gp, p_2, s_B)$ by requirement of $\boldsymbol{KE}$. A public key which has been altered in any way will be referred to as **modified**. A modified public key p^* that is guaranteed to satisfy:*

1. *p^* passes all validation tests Alice performs in the verification phase,*

2. *$\kappa(p^*) > 0$,*

3. *$\boldsymbol{KE}(gp, p^*, s_1) = \boldsymbol{KE}(gp, p_B, s_1)$, and*

4. *$\boldsymbol{KE}(gp, p^*, s_2) = \boldsymbol{KE}(gp, p_B, s_2)$,*

*will be called **malicious**. If it is computationally infeasible for Bob to modify his public key to some malicious p^* then we will say $\boldsymbol{KA}$ is **irreducible**.*

While it might be reasonable that a malformed public key could be chosen in such a way that it successfully performs key establishment with one known public key, constructing a malformed key to succeed simultaneously on two public keys poses a harder computational problem, as we will exhibit in Section 4.3. It is for this reason that conditions 3 and 4 are included in Definition 4.2.3.

We can now define our key agreement transformation. With the above general framework for a key agreement in mind, consider the following variant.

Definition 4.2.4. *Let $\mathbf{KE}$ be a key establishment function as above, let k be a positive integer, and let $\mathbf{H}$ be a random oracle. Consider the following key agreement process between Alice and Bob, called $k - \mathbf{KA}$:*

0. ***Setup**: Both members obtain a valid copy of the global parameters, gp.*

1. ***Key Generation**: Alice generates k secret key/public key pairs (s_{Ai}, p_{Ai}), $1 \leq i \leq k$. Likewise Bob generates (s_{Bi}, p_{Bi}) for $1 \leq i \leq k$.*

2. ***Communication**: Alice initiates communication and sends all k of her public keys to Bob. Bob then sends all k of his public keys to Alice.*

3. ***Key Establishment**: Alice computes $z_{i,j} \leftarrow \boldsymbol{KE}(gp, p_{Bi}, s_{Aj})$ for every pair $1 \leq i, j \leq k$, then computes*

$$h \leftarrow \boldsymbol{H}(z_{1,1}, \ldots, z_{1,k}, z_{2,1}, \ldots, z_{2,k}, \ldots, z_{k,1}, \ldots, z_{k,k}).$$

Similarly, Bob computes $z'_{i,j} \leftarrow \boldsymbol{KE}(gp, p_{Aj}, s_{Bi})$ for each pair $1 \leq i, j \leq k$, and then computes

$$h' \leftarrow \boldsymbol{H}(z'_{1,1}, \ldots, z'_{1,k}, z'_{2,1}, \ldots, z'_{2,k}, \ldots, z'_{k,1}, \ldots, z'_{k,k}).$$

4. ***Verification**: If applicable, Alice and Bob test the validity of each others public keys. Alice and Bob verify that h is equal to h' as follows: Alice sends $\mathbf{H}(\mathbf{H}(h))$ to Bob, and Bob responds with $\mathbf{H}(h')$. Alice checks that $\mathbf{H}(h) = \mathbf{H}(h')$ and Bob checks that $\mathbf{H}(\mathbf{H}(h')) = \mathbf{H}(\mathbf{H}(h))$. Either party terminates the session if their verification fails.*

When Alice and Bob perform honestly, it is clear that they will share the same key and verification will pass on both ends. We now present our main theorem which explains how the parameter k can affect the security of the protocol from attacks of the type mentioned in Definition 4.2.2.

Theorem 4.2.5. *Let $\boldsymbol{KA}$ be an irreducible key agreement protocol which uses the key establishment function $\boldsymbol{KE}(gp, \cdot, \cdot)$, for some global parameters gp. Let p^* be a modified public key with $\kappa(p^*) > 0$ that passes all validity tests of $\boldsymbol{KA}$, and let ρ denote the largest probability in the image of $\chi_{KE}(p^*, \cdot)$. Suppose that in k-$\boldsymbol{KA}$ one of the k parts to Bob's public key is p^*. If Bob has access to a type (1) oracle for k-$\boldsymbol{KA}$, then the largest probability in the image of $\chi_{k\text{-}KA}(p_B, \cdot)$ is ρ^{k-1}.*

In k-**KA** Bob has access to a type (2) oracle (see Definition 4.2.2) in the form of Alice sending $\mathbf{H}(\mathbf{H}(h))$ (or $\mathbf{H}(h)$ if the roles are reversed) as he can guess at the preimage and check his guess. However we are assuming that Bob has access to a type (1) oracle, that is he somehow recovers h from Alice during verification, which provides the adversary with greater capabilities. We now prove Theorem 4.2.5.

Proof. During the k-**KA** session, denote by $(p_{A1}, s_{A1}), \ldots, (p_{Ak}, s_{Ak})$ the keys generated by Alice and likewise $(p_{B2}, s_{B2}), \ldots, (p_{Bk}, s_{Bk})$ the keys generated by Bob, along with $p_{B1} = p^*$ (without loss of generality). Bob can potentially learn about Alice's secret keys during the verification phase. Alice will compute $z_{1,j} \leftarrow \mathbf{KE}(gp, p^*, s_{Aj})$ and $z_{i,j} \leftarrow \mathbf{KE}(gp, p_{Bi}, s_{Aj})$ for every $2 \leq i \leq k$ and $1 \leq j \leq k$. She then computes

$$h \leftarrow \mathbf{H}(z_{1,1}, \ldots, z_{1,k}, z_{2,1}, \ldots, z_{2,k}, \ldots, z_{k,1}, \ldots, z_{k,k}).$$

We are assuming Bob has access to a type (1) oracle, and so he has obtained h from Alice. As $\mathbf{H}$ is a random oracle, in order to determine any information about Alice's secret keys Bob must guess at the preimage of h. Bob can easily compute $z_{i,j} = \mathbf{KE}(gp, p_{Aj}, s_{Bi})$ for all $2 \leq i \leq k$, $1 \leq j \leq k$. Therefore determining the preimage relies completely on Bob's ability to find $z_{1,j} = \mathbf{KE}(gp, p^*, s_{Aj})$ for every $1 \leq j \leq k$, each of which is an instance of the original $\mathbf{KA}$ protocol, however he is only able to test a guess for the tuple $(z_{1,1}, \ldots, z_{1,k})$ instead of each one individually. By assumption, $\mathbf{KA}$ is irreducible and p^* is modified with $\kappa(p^*) > 0$ and passes all applicable validity tests. It follows that $\mathbf{KE}(gp, p_{Aj}, s_{B1})$ is not guaranteed to be equal to $\mathbf{KE}(gp, p^*, s_{Aj}) = z_{1,j}$ for more than one value of j. Bob can therefore be certain of no more than one value of $z_{1,j}$ before testing guesses.

Note that if Bob guesses $(x_1, \ldots, x_k) = (z_{1,1}, \ldots, z_{1,k})$, then the probability of success is unaffected by his previous guesses. Therefore the probability that each of Bob's guesses of $z_{1,j}$ is bounded above by ρ, except for possibly the one value which can be forced to be $\mathbf{KE}(gp, p_{Aj}, s_{B1})$ by Bob's choice of p^*. Since the type (2) oracle only returns 1 if all k instances are correct, Bob's maximum probability of success on any guess is ρ^{k-1}. □

More than the theorem's result, the proof shows that the probability that a guess $(x_1, \ldots, x_k)$ is equal to $(z_{1,1}, \ldots, z_{1,k})$ is the product that each individual x_j is equal to $z_{1,j}$, $1 \leq j \leq k$, with the exclusion of no more than one j by the irreducibility assumption.

4.3 Multiple Instances of SIDH

In this section we will apply the previous theory to the SIDH key agreement protocol to enable secure use of static keys. We then estimate the expected amount of work required to break our transformation in this case.

Regarding Definition 4.2.1, the SIDH protocol as defined in 2.2.2 is incomplete since it does not state how Alice and Bob check if they computed the same secret in the verification phase. This step is made explicit when we apply our multiple instances model.

For general background on elliptic curves and SIDH refer to Chapter 2. Let the field characteristic be $p = 2^m 3^n - 1$. In the original SIDH scheme [47] the key generation phase produces two values, say α_1 and α_2 (not both divisible by 2) as Alice's private key, her isogeny ϕ_A has kernel $\langle [\alpha_1]P_A + [\alpha_2]Q_A \rangle$, and she takes the analogous linear combination during the key establishment phase. However, through a change of variables one can always obtain kernel $\langle P_A + [\alpha]Q_A \rangle$ or $\langle [\alpha]P_A + Q_A \rangle$ since at least one of α_1 or α_2 is invertible modulo 2^m. Throughout the remainder of this work we assume without loss of generality that we fall into the former case because it simplifies our analysis.

The SIDH key establishment protocol relies on the difficulty of the CSSI computation problem (see Problem 2.2.6). As mentioned, this Diffie-Hellman type protocol is susceptible to an active attack if Alice uses the same private key in different sessions [53, §3]. We describe this attack now. Note a similar attack applies when the torsion subgroups are not powers of 2 and 3.

Instead of using the public key $(E_B, \phi_B(P_A), \phi_B(Q_A))$ when communicating with Alice, a dishonest Bob can send

$$(E_B, R, S) = (E_B, [\theta]\phi_B(P_A), [\theta](\phi_B(Q_A) + [2^{m-1}]\phi_B(P_A))),$$

where θ is chosen such that $e_{2^m}(R, S) = e_{2^m}(P_A, Q_A)^{3^n}$. This modified public key is certain to pass the validation methods in [34, §9]. The parity of Alice's private key α can then be determined as follows. The subgroup computed by Alice during key establishment is $\langle R + [\alpha]S \rangle$. When α is even this subgroup is equal to $\langle \phi_B(P_A) + [\alpha]\phi_B(Q_A) \rangle$, but the subgroup will be different when α is odd. Therefore, if Bob performs his half of the key establishment honestly and uses the shared secret key $E_A/\langle \phi_A(P_B) + [\beta]\phi_A(Q_B) \rangle$ during verification, then he can determine the parity of α based on Alice terminating the session or not. This attack can be extended adaptively to learn each bit of α efficiently and without detection when using the described validation methods. An indirect validation technique [63] is available which prevents the attack, but at the cost of Bob revealing his private key so that Alice can verify the message he sends was computed honestly, which also causes Alice to perform twice as many computations as in SIDH.

This active attack suggests that static keys can no longer be used for SIDH key exchange unless the other party is using an ephemeral key. In addition, it requires that all holders of static keys must double their computational costs, recomputing the other participant's message in order to verify the validity of the message.

We now apply the multiple instances model of Section 4.2 to create a k-**KA** scheme based on supersingular isogenies. For the security proof of Theorem 4.2.5 to apply we need to show that SIDH is irreducible as defined in Definition 4.2.3. We first address the case where a malicious Bob scales his public torsion points by some invertible element.

Lemma 4.3.1. *Suppose Alice and Bob participate in an instance of SIDH key agreement and that Bob uses the dishonest public key*

$$p^* = (E_B, [\mu]\phi_B(P_A), [\mu]\phi_B(Q_A))$$

for some μ coprime to order of P_A and Q_A. Then p^* is not a malicious key in the sense of Definition 4.2.3.

Proof. Denote the order of Alice's torsion subgroup by ℓ_A^m and Bob's by ℓ_B^n. The verification phase of SIDH consists of checking that the two torsion points are independent, have the correct order, satisfy the Weil pairing condition, and that both parties compute the same shared secret key. The order and independence conditions follow immediately from the assumption that ℓ_A and μ are coprime. By Proposition 2.1.25 and the bilinearity of the Weil pairing,

$$e_{\ell_A^m}([\mu]\phi_B(P_A), [\mu]\phi_B(Q_A)) = e_{\ell_A^m}(P_A, Q_A)^{\mu^2 \ell_B^n}.$$

Therefore p^* passes the Weil pairing test if and only if $\mu^2 \equiv 1 \mod \ell_A^m$. Lastly, if we denote Alice's private key by α, then

$$\begin{aligned}\langle [\mu]\phi_B(P_A) + [\alpha]\left([\mu]\phi_B(Q_A)\right)\rangle &= \langle [\mu]\left(\phi_B(P_A) + [\alpha]\phi_B(Q_A)\right)\rangle \\ &= \langle \phi_B(P_A) + [\alpha]\phi_B(Q_A)\rangle,\end{aligned}$$

where the second equality follows from μ being coprime to ℓ_A.

This shows that if Bob modifies his public key in this way, then Alice will compute the same shared secret independent of her private key. Therefore no more information about her private key can be leaked by Alice accepting (or rejecting if $\mu^2 \not\equiv 1$) than is already leaked when Bob performs honestly. Hence, $\kappa(p^*) = 0$ and this modification does not result in a malicious public key. □

It is worth noting that if Bob scales his two torsion points by different scalers, say μ_1 and μ_2, then they will no longer generate the same subgroup under Alice's private key by the independence of $\phi_B(P_A)$ and $\phi_B(Q_A)$, again resulting in a public key which is not malicious. Now we can prove that isogenies lend themselves to the transform of Aection 4.2.

Theorem 4.3.2. *Under the assumption that the CSSI problem is intractable, it is computationally infeasible for a malicious Bob with non-negligible probability to modify his public key $(E_B, \phi_B(P_A), \phi_B(Q_A))$ to some $p^* = (E_B, R, S)$ which is malicious for SIDH.*

Proof. Let $p = \ell_A^m \ell_B^n f \pm 1$ be prime, let E be an elliptic curve defined over $\mathbb{F}_{p^2}$, and let P_A, Q_A, P_B and Q_B be points on $E(\mathbb{F}_{p^2})$ such that $\langle P_A, Q_A \rangle = E[\ell_A^m]$ and $\langle P_B, Q_B \rangle = E[\ell_B^n]$. Alice has some public key/secret key pair

$$\phi_{A1} : E \to E_{A1} = E/\langle P_A + [\alpha_1]Q_A \rangle, \ \alpha_1 \in \mathbb{Z}/\ell_A^m\mathbb{Z}.$$

Bob knows the global parameters p, P_A, Q_A, P_B and Q_B, and receives the public key $(E_{A1}, \phi_{A1}(P_B), \phi_{A1}(Q_B))$ from Alice. By the assumption of intractability of the CSSI problem, it should be infeasible for Bob to compute α_1. The goal of our proof is to show that if Bob can violate the definition of irreducibility by computing p^* in the statement of the theorem, then he can compute α_1 efficiently which violates the CSSI assumption.

Bob uses the SIDH key generation algorithm twice, to generate

$$\alpha_2 \in \mathbb{Z}/\ell_A^m\mathbb{Z}, \ \phi_{A2} : E \to E_{A2} = E/\langle P_A + [\alpha_2]Q_A \rangle, \text{ and}$$

$$\beta \in \mathbb{Z}/\ell_B^n\mathbb{Z}, \ \phi_B : E \to E_B = E/\langle P_B + [\beta]Q_B \rangle.$$

Suppose for contradiction that Bob is able to modify $(E_B, \phi_B(P_A), \phi_B(Q_A))$ to some malicious public key (E_B, R, S), violating irreducibility as stated in Definition 4.2.3. That is:

- (E_B, R, S) passes all validation tests,
- $j(E_B/\langle R + [\alpha_1]S \rangle) = j(E_B/\langle \phi_B(P_A) + [\alpha_1]\phi_B(Q_A) \rangle)$,
- $j(E_B/\langle R + [\alpha_2]S \rangle) = j(E_B/\langle \phi_B(P_A) + [\alpha_2]\phi_B(Q_A) \rangle)$, and
- $\kappa(E_B, R, S) > 0$.

Since we cannot fully characterize public keys with $\kappa(p^*) > 0$ in this setting, we instead use the condition that $(R, S) \neq ([\mu]\phi_B(P_A), [\mu]\phi_B(Q_A))$ for some μ coprime to ℓ_A. By Lemma 4.3.1 these public keys satisfy $\kappa(p^*) = 0$, so we are assuming a potentially weaker condition than $\kappa(p^*) > 0$ by excluding only public keys of this type.

To simplify notation for the remainder of this proof we set $\ell = \ell_A$. The subgroups $\langle R + [\alpha_1]S \rangle$ and $\langle R + [\alpha_2]S \rangle$ are guaranteed to be kernels of isogenies from E to elliptic curves isomorphic to E_{A1} and E_{A2} respectively by the j-invariant requirements. For the first subgroup one of two cases is true:

i The isogeny with kernel $\langle R+[\alpha_1]S \rangle$ is isomorphic to the isogeny with kernel $\langle \phi_B(P_A)+ [\alpha_1]\phi_B(Q_A) \rangle$, ii The isogeny with kernel $\langle R + [\alpha_1]S \rangle$ is not isomorphic to the isogeny with kernel $\langle \phi_B(P_A) + [\alpha_1]\phi_B(Q_A) \rangle$.

Likewise, there are two cases for α_2 and the isogeny to E_{A2}. For the remainder of the proof we assume that both isogenies fall into case (i) as our reduction only applies in this situation. This point will be examined in greater detail in the runtime analysis at the end of the proof. This distinction of cases must be made as it is possible for the two isogenies to be non-isomorphic and yet the torsion points R and S (or some scaling of them) still satisfy all the requirements of the verification phase, including the Weil pairing test that $e_{\ell^m}(R, S) = e_{\ell^m}(P_A, Q_A)^{\ell_B^n}$ (see [53, §3.2] for details).

Suppose the isogeny with kernel $\langle \phi_B(P_A) + [\alpha_i]\phi_B(Q_A) \rangle$ is isomorphic to that of $\langle R + [\alpha_i]S \rangle$ for both $i \in \{1, 2\}$. Then the two subgroups themselves are equal for each i. It follows that their generators must then differ by a scalar multiple coprime to the order of the subgroup. We can then write

$$[\lambda_i](\phi_B(P_A) + [\alpha_i]\phi_B(Q_A)) = R + [\alpha_i]S, \tag{4.1}$$

for some $\lambda_i \in \mathbb{Z}/\ell^m\mathbb{Z}$ coprime to ℓ^m (i.e. coprime with ℓ), for both $i \in \{1, 2\}$.

Since ℓ is a small prime, the elliptic curve discrete log problem is tractable on $E_B[\ell^m]$ using Pohlig-Hellman [79] and the Weil or Tate pairing (see [5, §3.2] and optimization [33, §4-5]). Solving two instances of the two-dimensional ECDLP provides $a, b, c, d \in \mathbb{Z}/\ell^m\mathbb{Z}$ such that

$$R = [a]\phi_B(P_A) + [b]\phi_B(Q_A), \text{ and } S = [c]\phi_B(P_A) + [d]\phi_B(Q_A). \tag{4.2}$$

Substituting these decompositions into (4.1) and rearranging we obtain

$$[\lambda_1](\phi_B(P_A) + [\alpha_1]\phi_B(Q_A)) = [a + \alpha_1 c]\phi_B(P_A) + [b + \alpha_1 d]\phi_B(Q_A).$$

The points P_A and Q_A are independent—there does not exist $t \in \mathbb{Z}/\ell^m\mathbb{Z}$ such that $P_A = [t]Q_A$. Therefore $\phi_B(P_A)$ and $\phi_B(Q_A)$ are independent as well. Comparing coefficients of $\phi_B(P_A)$ implies that $\lambda_1 \equiv a + \alpha_1 c \bmod \ell^m$. Comparing coefficients of $\phi_B(Q_A)$ then gives the congruence

$$b + \alpha_1 d \equiv \lambda_1\alpha_1 \equiv (a + \alpha_1 c)\alpha_1 \bmod \ell^m. \tag{4.3}$$

Similar analysis of the subgroups associated with α_2 result in the congruence

$$b + \alpha_2 d \equiv (a + \alpha_2 c)\alpha_2 \bmod \ell^m. \tag{4.4}$$

Rearranging (4.3) and (4.4) gives

$$c\alpha_1^2 + (a - d)\alpha_1 - b \equiv 0 \bmod \ell^m, \text{ and } c\alpha_2^2 + (a - d)\alpha_2 - b \equiv 0 \bmod \ell^m.$$

Therefore α_1 and α_2 are solutions to the quadratic congruence relation

$$cx^2 + (a-d)x - b \equiv 0 \bmod \ell^m. \tag{4.5}$$

Bob has the ability to construct this polynomial. One approach to solving this equation comes from the assumption that α_1 and α_2 are simple roots modulo ℓ (this is the same assumption required in Hensel's lemma) as it implies $\alpha_1 - \alpha_2$ is invertible modulo ℓ^m. By subtracting (4.4) from (4.3) and multiplying the result by $(\alpha_1 - \alpha_2)^{-1} \mod \ell^m$ we obtain

$$d \equiv a + c(\alpha_1 + \alpha_2) \mod \ell^m, \tag{4.6}$$

and it follows that

$$b \equiv -c\alpha_1\alpha_2 \mod \ell^m. \tag{4.7}$$

Therefore, if $\ell^r \mid c$, then $\ell^r \mid a - d$ and $\ell^r \mid b$ too. If $c \equiv 0 \mod \ell^m$, then $b \equiv 0$ and $a \equiv d \bmod \ell^m$, which contradicts the assumption that (E_B, R, S) is malicious by Lemma 4.3.1.

From the malicious public key (E_B, R, S), Bob can now efficiently solve for α_1 and α_2 using the following process:

1 Compute the discrete log coefficients $a, b, c, d \in \mathbb{Z}/\ell^m\mathbb{Z}$ as above.

2 Write $c = \ell^r g$ for some g indivisible by ℓ and $0 \leq r < m$.

3 Let $K = g^{-1}\dfrac{a-d}{\ell^r} \mod \ell^{m-r}$ and $L = -g^{-1}\dfrac{b}{\ell^r} \mod \ell^{m-r}$, where the inverse of g is computed modulo ℓ^{m-r}.

4 α_1 and α_2 are roots of the quadratic $x^2 + Kx + L \equiv 0 \bmod \ell^{m-r}$ by (4.5). Solve for all roots of this polynomial modulo ℓ^{m-r}.

5 For each root, u, extend it to an integer mod ℓ^m, say u', and test if it is equal to α_1. This test can be performed by computing the image curve of the isogeny with $\langle P_A + [u']Q_A \rangle \subset E(\mathbb{F}_{p^2})$ as its kernel and comparing its j-invariant with $j(E_{A1})$ (the image curve of the isogeny with $\langle P_A + [\alpha_1]Q_A \rangle$ as its kernel).

What remains is to analyze the computational cost of this reduction and the probability of success. For this analysis, we need to know the likelihood of our assumptions, the probable size of the value r, and the number of roots of the quadratic congruence.

The first assumption is that the subgroup associated to the points R and S is the same as the isogeny kernel in the SIDH instance. The existence of multiple isogenies of degree ℓ^m between two fixed supersingular elliptic curves is possible, but unlikely under the Galbraith et al. heuristic of [53, §4.2]. For instance there can exist multiple isogenies of degree ℓ from one j-invariant, j_0, to another and this occurs exactly when the classical modular polynomial $\psi_\ell(j_0, x)$ has repeated roots in x. The set of possible roots grows with p and yet its degree in x is fixed by $\ell + 1$, so this situation unlikely for large p.

Next we examine the value r when $\alpha_1 \equiv \alpha_2 \mod \ell$. When $\ell = 2$, we have that $r \geq 3$ whenever $m > 3$. From the distribution of multiples of ℓ in $\mathbb{Z}/2^m\mathbb{Z}$, we have $r = j$ for $3 \leq j < m$ with probability $\frac{1}{2^{j-2}}$, and the probability that $r = m$ (i.e. $c = 0$) is $\frac{1}{2^{m-3}}$. When ℓ is odd, only $r \geq 1$ is guaranteed. For $\mathbb{Z}/\ell^m\mathbb{Z}$ with odd ℓ, we have $r = j$ for $1 \leq j < m$ with probability $\frac{1}{\ell^j}$, and the probability that $r = m$ is $\frac{1}{2^{m-1}}$. Hence, it is most likely that $r = 3$ or 4 when $\ell = 2$, and $r = 1$ or 2 when ℓ is an odd prime.

Lastly, we look at the number of solutions to (4.5). If $\alpha_1 \not\equiv \alpha_2 \bmod \ell$ and $\ell \nmid c$, then there are exactly two solutions modulo ℓ^m, namely α_1 and α_2. Letting r be the ℓ-adic valuation of c as above, the number of solutions to this quadratic congruence is $2\ell^r$, namely

$$\alpha_i + z\ell^{m-r-1},\ 0 \leq z \leq \ell^r - 1,\ i \in \{1, 2\}.$$

Even though the number of roots to check grows exponentially in r, the probability of each successive value of r occurring decreases exponentially (see the previous paragraph).

When α_2 is chosen to be congruent to α_1 modulo ℓ, b and d are not necessarily of the form (4.6) and (4.7). This makes solving for α_1 much harder, and in some cases, impossible. However, this only happens with probability $\frac{1}{\ell}$. By the previous paragraph we see that if Bob counts the number of roots of (4.5) modulo ℓ^{m-r} before solving for α_1, then verifying there are less than ℓ^{r+1} of them can serve to test for when $\alpha_1 \equiv \alpha_2 \mod \ell$. If the test fails then Bob can reuse the key generation algorithm until the private key provided is incongruent to the initial α_2, and then repeat the process above (he never has to run this process more than twice).

We conclude that if Bob can violate this irreducibility condition, then he can efficiently solve the CSSI problem. □

Combining Theorem 4.3.2 with the fact that there are currently no know attacks on SIDH that involve modified elliptic curves (as opposed to modified torsion points) we conclude that SIDH is irreducible for all known modified public keys. We now give an explicit statement of the k-SIDH protocol.

Setup: A preimage resistant hash function H, a prime number $p = 2^m 3^n f \pm 1$, a supersingular elliptic curve $E/\mathbb{F}_{p^2}$, and four points $P_A, Q_A, P_B, Q_B \in E(\mathbb{F}_{p^2})$ such that $\langle P_A, Q_A \rangle = E[2^m]$ and $\langle P_B, Q_B \rangle = E[3^n]$.

Key Generation: Upon input of 0, the key generation function computes, for $1 \leq i \leq k$:

$$\begin{aligned} \alpha_i &\leftarrow_R \mathbb{Z}/2^m\mathbb{Z}, \\ \phi_{Ai} \colon E \to E_{Ai} &= E/\langle P_A + [\alpha_i]Q_A \rangle, \\ (R_i, S_i) &\leftarrow (\phi_{Ai}(P_B), \phi_{Ai}(Q_B)). \end{aligned}$$

The key generation function then outputs the private key $(\alpha_1, \ldots, \alpha_k)$ and the public key $(E_{A1}, R_1, S_1), \ldots, (E_{Ak}, R_k, S_k)$. The recipient checks that the order of each R_i and S_i is 3^n to ensure no faults were induced.

Upon input of 1 the key generation function computes, for $1 \leq j \leq k$:

$$\begin{aligned} \beta_j &\leftarrow_R \mathbb{Z}/3^n\mathbb{Z}, \\ \phi_{Bj} : E \to E_{Bi} &= E/\langle P_B + [\beta_j]Q_B \rangle, \\ (U_j, V_j) &\leftarrow (\phi_{Bj}(P_A), \phi_{Bj}(Q_A)). \end{aligned}$$

The key generation function then outputs the private key $(\beta_1, \ldots, \beta_k)$ and the public key $(E_{B_1}, U_1, V_1), \ldots, (E_{B_k}, U_k, V_k)$. The recipient checks that the order of each U_j and V_j is 2^m to ensure no faults were induced.

Communication: Bob initiates conversation and sends his public key to Alice. Alice responds with her public key.

Key Establishment: For each $1 \leq i, j \leq k$, Alice computes

$$z_{i,j} = j(E_{Bj}/\langle U_j + [\alpha_i]V_j \rangle),$$

and then she calculates the hash

$$h = H(z_{1,1}, \ldots, z_{1,k}, z_{2,1}, \ldots, z_{2,k}, \ldots, z_{k,1}, \ldots, z_{k,k}).$$

Similarly, for each $1 \leq i, j \leq k$, Bob computes

$$z'_{i,j} = j(E_{Ai}/\langle R_i + [\beta_j]S_i \rangle),$$

and calculates the hash

$$h' = H(z'_{1,1}, \ldots, z'_{1,k}, z'_{2,1}, \ldots, z'_{2,k}, \ldots, z'_{k,1}, \ldots, z'_{k,k}).$$

Verification: Alice verifies that for each $1 \leq j \leq k$ the pair U_j and V_j are independent points of order 2^m on the curve E_{Bj} [34, §9]. Additionally Alice verifies that $e_{2^m}(U_j, V_j) =$

$e_{2^m}(P_A, Q_A)^{3^n}$. Likewise Bob verifies that each pair R_i and S_i are independent points of order 3^n on the curve E_{Ai} and that $e_{3^n}(R_i, S_i) = e_{3^n}(P_B, Q_B)^{2^m}$. Alice sends $H(H(h))$ to Bob who verifies it is equal to $H(H(h'))$. Bob sends $H(h')$ to Alice who verifies it is equal to $H(h)$. If they have different secret keys, or any of the public key pairs fail the verification, then the session is terminated. Otherwise they continue communication with $h = h'$ as their shared secret key.

4.4 Security Analysis and Key Size

Before the security of k-SIDH can be properly analyzed we need the following simple result. As before, let $\ell = \ell_A$ denote the prime defining Alice's torsion subgroup. Recall that an ℓ^m-degree isogeny can be expressed uniquely as a composition of m ℓ-degree isogenies. The following result tells us that, given the shared j-invariant and Alice's public key, Bob is unable to determine the final ℓ-isogeny in the composition of Alice's isogeny (under the CSSI assumption).

Theorem 4.4.1. *Suppose Alice and Bob perform the standard SIDH key agreement protocol as described Section 2.2.2. In the key establishment phase Alice computes a secret isogeny $\phi_A : E_B \to E_{BA}$ of degree ℓ^m $(\ell \in \{2, 3\})$ as the composition of m isogenies of degree ℓ, say $\phi_A = \phi_m \circ \cdots \circ \phi_1$. Let $\phi' = \phi_{m-1} \circ \cdots \circ \phi_1$ be the isogeny whose image curve is ℓ isogenous to E_{AB}, say $\phi' : E_B \to E'$. Bob also knows the curve E_{BA} by performing his half of the key establishment. If Bob has access to an efficient, deterministic algorithm which produces E' from E, E_A, E_B, E_{BA} and ℓ^m, then Bob can efficiently solve the CSSI problem.*

Proof. The elliptic curve E' is ℓ isogenous to E_{BA}. Given E' Bob can then determine ϕ_m as there are only $\ell + 1$ choices which he can test exhaustively. Repeated iterations of the procedure, replacing the target curve adaptively and decreasing the exponent of the degree iteratively by 1, will return each ℓ-isogeny in the composition. This procedure will reveal ϕ_A, breaking CSSI. □

The security of this scheme is based on the amount of work Bob must do in the proof of Theorem 4.2.5 to compute the preimage of h. There are two benchmarks that we could use when choosing k: the expected number of hashes Bob will compute before correctly hashing the preimage, or the number of hashes before the solution is found with probability $\frac{1}{2}$. The former is asymptotically greater in our case, and so it is irrelevant when setting a security level.

The runtime depends on the order Bob guesses at solutions, so we always assume he does so optimally. We index Bob's guesses by i, and denote the associated probability of success by P_i. The proof of Theorem 4.2.5 shows that for Bob to determine the preimage of h he must correctly guess at least $k-1$ independent samples from some distribution. We now determine that distribution for SIDH.

In the attack of Galbraith et al. [53], the public key with the greatest ratio of revealed bits of Alice's private key to probability of success that Bob could use is $p^* = (E_B, \phi_B(P_A), \phi_B(Q_A) + [\ell^{m-1}]\phi_B(P_A))$. Bob knows the shared key that Alice computes were he to participate honestly,

$$j_0 = j(E_A/\langle \phi_A(P_B) + [\beta]\phi_B(Q_A)\rangle),$$

and when using this dishonest p^* he knows Alice will compute either j_0 or some other j-invariant which is ℓ^2-isogenous to j_0. With overwhelming probability there are $\ell(\ell+1)$ distinct isomorphism families which are ℓ^2-isogenous to any isomorphism family (not represented by the j-invariant 0 or 1728). Combining this fact with the Theorem 4.4.1 shows that k-SIDH exhibits the following probability distribution for each of Bob's k guesses:

$$\left\{\frac{1}{2}, \frac{1}{2\ell(\ell+1)}, \dots, \frac{1}{2\ell(\ell+1)}\right\},$$

where $\dfrac{1}{2\ell(\ell+1)}$ occurs $\ell(\ell+1)$ times. For example, if $\ell = 2$, then the honestly computed j-invariant, j_0, occurs with probability $\frac{1}{2}$, and there are six j-invariants which are 4-isogenous to j_0 each occurring with probability $\frac{1}{12}$.

The guess that maximizes Bob's probability of success is j_0 for each of the $k-1$ unknown values, resulting in $P_1 = \dfrac{1}{2^{k-1}}$. The next most likely outcomes are those with j_0 for $k-2$ of the values and one of the other $\ell(\ell+1)$ j-invariants, each occurring with probability

$$P_i = \frac{1}{2^{k-2} \cdot (2\ell(\ell+1))} \text{ for } 2 \leq i \leq (k-1)(\ell(\ell+1)) + 1.$$

From this we calculate r, the number of hashes that Bob computes before his probability of success is $\frac{1}{2}$ by solving $\dfrac{1}{2} = \sum_{i=1}^{r} P_i$.

The first step is to collect all guesses with the same probability of success, that is, those which select the same number of j_0. To achieve this we change from the variable r, the total number of guesses Bob makes, to t which represents the quantity of non-j_0 elements in Bob's choice. They are related by

$$r = \sum_{i=0}^{t} \binom{k-1}{i} (\ell(\ell+1))^i$$

as each term in the summand is the number of possibilities with i non-j_0 elements. Therefore,

$$\sum_{i=1}^{r} P_i = \sum_{i=0}^{t} \frac{1}{2^{k-1-i}(2\ell(\ell+1))^i} \binom{k-1}{i} (\ell(\ell+1))^i = \frac{1}{2^{k-1}} \sum_{i=0}^{t} \binom{k-1}{i},$$

and this final sum equals $\frac{1}{2}$ exactly when $t = \frac{k-2}{2}$ (if k is odd then the sum needs one half times the $\binom{k-1}{\frac{k-1}{2}}$ term) by the symmetry of the binomial coefficient. This implies that the number of hashes required by Bob to learn the first bit of each of Alice's k secret keys is

$$r = \sum_{i=0}^{\frac{k-2}{2}} \binom{k-1}{i} (\ell(\ell+1))^i. \tag{4.8}$$

If $\ell = 2$, then $k = 60$ gives $r = 2^{130}$; for $\ell = 3$, 2^{131} hashes is achieved by $k = 50$.

When considering security against a quantum enabled adversary, one would expect a quadratic speedup because the runtime of Grover's algorithm [56] on a non-uniform distribution is still $O(\sqrt{N})$ when searching for one item, where N is the size of the domain [13]. The domain for k-SIDH when Bob uses a malicious public key is all possible preimages to Alice's hash. Considering an initial state of each possible preimage (where each preimage is a collection of qubits representing the associated j-invariants) all with amplitude $\frac{1}{\sqrt{(2(\ell(\ell+1))^{k-1})}}$, and searching for an element of the marked set (the collection of qubits corresponding to the correct preimage) gives a quantum algorithm with runtime $\frac{\pi}{4} 2^{\frac{k-1}{2}} (\ell(\ell+1))^{\frac{k-1}{4}}$ and requires at least $(2(\ell(\ell+1))^{k-1}$ qubits. We then calculate the minimal k such that Bob is required to compute 2^{128} quantum operations before successfully calculating the preimage of Alice's hash, which would reveal k bits of her secret key. Setting $k = 113$ is required when $\ell = 2$, and $k = 94$ suffices when $\ell = 3$.

These choices of k are based on the currently best known attack that satisfy Definition 4.2.2. There is the possibility that other attacks will be discovered such as modifying the elliptic curve in a public key instead of the torsion points, or perhaps stronger attacks using modified torsion points. However, if such attacks are discovered, the generality of the Theorem 4.2.5 shows that only a recalculation of k is needed to adapt as these qualify as malicious public key attacks.

To achieve a specified level of security for k-SIDH each individual SIDH instance must also meet that security level. Using the compression techniques of [33, §7], at the 128-bit quantum security (or 192-bit classical security) level a k-SIDH pubic key requires 37 kb when $\ell = 2$ and 31 kb when $\ell = 3$.

Regarding future work, the proposed k-instances model applies to key agreement schemes in which the resulting shared secret is dependent on input from both parties (not encapsulation methods) where the use of static keys may reveal private keys to a malicious participant. We have seen that this transformation applies to SIDH [53], but one may ask if it also applies to lattice based schemes. The ring-LWE key agreement protocol by Ding et al. [43] satisfies the criterion of being susceptible to such an active attack [48]. However, the scheme of Ding et al. is interactive and so it is not immediately clear how to model it as the key-establishment protocols of Definition 4.2.1.

The computational costs of k-SIDH are naively k^2 that of standard SIDH, in which parties simply perform k^2 independent SIDH operations. Economies of scale could be realized in an optimized implementation using (for example) SIMD, since the key establishments can be organized into k groups of k such that all SIDH operations in a group have one half in common.

k-SIDH only addresses the problem of dishonest users who manipulate elliptic curve points. It does not address the case where the curves themselves are manipulated. It may be worth examining whether approaches like k-SIDH can help protect against attacks involving manipulated curves. Another interesting problem comes from the heuristic assumption from [53, §4.2] which was used in the proof of Theorem 4.3.2. Although this assumption seems plausible in light of the Ramanujan property of the supersingular ℓ-isogeny graph, a proof would be preferable, perhaps under a standard assumption such as GRH. Similar results have been achieved in the ordinary case [60].

We conclude with a summary of the work in this chapter. We presented a new key agreement model which performs k^2 simultaneous key agreements and defends against a specific class of active adversaries when certain assumptions about the underlying key agreement protocol are satisfied. We showed that supersingular isogeny key agreement satisfies these assumptions provided its computational problem is intractable. Using this new model, we determined that performing $60 \cdot 50 = 3000$ simultaneous instances of SIDH will protect both participants from leaking any information of their secret key against these active adversaries with classical capabilities, and $113 \cdot 94 = 10622$ suffices for protection against quantum adversaries.

Chapter 5

Cryptanalysis from Constructing Endomorphisms

Portions of this chapter were published in Eisentraeger et al. [46]. The contents of this chapter represent my contribution to that work.

We present an algorithm for computing non-trivial endomorphisms of a supersingular elliptic curve over $\mathbb{F}_{p^2}$. We then analyze its expected runtime and find it to be $\tilde{O}(\sqrt{p})$. We then discuss how to use this algorithm as a subroutine to compute the full endomorphism ring of elliptic curves again with runtime in $\tilde{O}(\sqrt{p})$, and compare it to the other endomorphism computing algorithms.

5.1 Introduction

Let $\mathbb{B}_{p,\infty}$ be a quaternion algebra ramified exactly at p and ∞, and let $\mathcal{O}$ be a maximal order of $\mathbb{B}_{p,\infty}$. For any supersingular elliptic curve $E/\overline{\mathbb{F}}_p$, the endomorphism ring $\mathrm{End}(E)$ is isomorphic to some maximal order $\mathcal{O}$ of $\mathbb{B}_{p,\infty}$. Deuring gave a correspondence [42] between left-ideal classes of $\mathcal{O}$ and the endomorphism rings of supersingular elliptic curves over $\overline{\mathbb{F}}_p$ up to Galois conjugacy of their j-invariants. As all supersingular j-invariants are defined over $\mathbb{F}_{p^2}$, there are no more than 2 isomorphism classes of supersingular elliptic curves in each Galois conjugacy class.

The security of supersingular isogeny-based schemes depends on the intractibility of the ℓ-isogeny problem, which is the problem of determining an isogeny $\phi : E \to E'$ with ℓ-power degree when given only E and E'. The intractibility of the ℓ-isogeny problem in turn depends on the intractability of computing $\mathrm{End}(E)$ from a random supersingular elliptic curve E. This is because the ℓ-isogeny problem can heuristically be solved in polynomial time when the endomorphism rings of both the domain and codomain curves are known [65]. As the ℓ-power degree in SIDH is small compared to p, the isogeny output by the algoritm of [65] will also be the solution to the corresponding CSSI problem with high probability [53, Algorithm 2].

The first to study the problem of computing $\mathrm{End}(E)$ for a supersingular elliptic curve was Kohel in his PhD thesis [64]. The algorithms in his work find two independent cycles containing $j(E)$ in the ℓ-isogeny graph of supersingular elliptic curves, which correspond to endomorphisms of E of ℓ-power degrees. These independent endomorphisms generate a finite index subring of $\mathrm{End}(E)$. This algorithm performs a breadth-first search of the graph to find cycles, and so it has an expected running time of $O(p^{1+\epsilon})$ and requires exponential storage. Heuristically, one expects this algorithm to be called $O(\log p)$ many times before the entire endomorphism ring of E can be determined [45].

An improvement to the problem of computing endomorphisms was presented in [41], by taking a depth-first search in a graph with many prime degree isogenies (instead of only the ℓ-graph), which ran in $O(p^{1/2}(\log p)^2)$ time and required polynomial space. Again, $O(\log p)$ many calls could be used to then determine $\mathrm{End}(E)$ entirely.

We present a probabilistic algorithm for finding cyclic endomorphisms of supersingular elliptic curves with prime power degrees which takes advantage of a symmetry in the ℓ-isogeny graph. We prove a runtime of $O(p^{1/2}(\log p)^2)$ with probability $1 - O(1/p)$, and $\mathrm{polylog}\, p$ space requirements. While this work does not represent the fastest currently known attack on supersingular isogeny-based cryptosystems, it is the fastest known algorithm for attacking such schemes by way of computing endomorphism rings.

5.2 Quaternion Preliminaries

Assume throughout that p is a prime, and $p \neq 2,\ 3$.

$\mathbb{A}$ is a quaternion algebra over $\mathbb{Q}$ when it is a central, simple algebra of dimension 4 over $\mathbb{Q}$. That is, there exists a ring homomorphism from $\mathbb{Q} \to \mathbb{A}$ which is an isomorphism to the centre of $\mathbb{A}$, and gives $\mathbb{A}$ the structure of a 4-dimensional vector space over $\mathbb{Q}$, and there are no non-trivial two-sided ideals of $\mathbb{A}$. A lattice, Λ, of $\mathbb{A}$ is a finitely generated $\mathbb{Z}$-module which contains a basis for $\mathbb{A}$ over $\mathbb{Q}$, i.e., $\Lambda \otimes \mathbb{Q} = \mathbb{A}$. An order of $\mathbb{A}$ is a lattice containing 1. An ideal of an order of $\mathbb{A}$ is defined in the usual way, and they are therefore lattices not necessarily containing 1. The norm of an ideal is defined as $N(I) = \gcd(\alpha : \alpha \in I)$.

For any prime q, define $\mathbb{A}_q = \mathbb{A} \otimes_{\mathbb{Q}} \mathbb{Q}_q$, and $\mathbb{A}_\infty = \mathbb{A} \otimes_{\mathbb{Q}} \mathbb{R}$. If $\mathbb{A}_q$ is a division algebra, then q is said to ramify; otherwise $\mathbb{A}_q \cong \mathbb{M}_2(\mathbb{Q}_q)$ and q is said to split. Given a finite set of places where $\mathbb{A}$ ramifies, there exists a unique quaternion algebra up to isomorphism of algebras.

There exists an equivalence relation on maximal orders of a quaternion algebra.

Definition 5.2.1. *Two maximal orders, $\mathcal{O}$ and $\mathcal{O}'$, have the same* ***type*** *if there exists some quaternion $\alpha \in \mathbb{A}^*$ such that $\alpha\mathcal{O}\alpha^{-1} = \mathcal{O}'$. Similarly, two left-ideals, I and J, of a maximal order are* ***equivalent*** *if there exists some quaternion $\alpha \in \mathbb{A}^*$ such that $I\alpha = J$.*

Let $\mathbb{B}_{p,\infty}$ be the unique quaternion algebra over $\mathbb{Q}$ which is ramified only at p and ∞. Deuring [42, §10.2] proved a bijection between the two-sided ideal classes for each type of maximal order in $\mathbb{B}_{p,\infty}$ and the supersingular j-invariants over $\bar{\mathbb{F}}_p$. Additionally,

Theorem 5.2.2. *[42, §10.2] Let $\mathcal{O}$ be a maximal order of $\mathbb{B}_{p,\infty}$. There exists one or two supersingular j-invariants such that the elliptic curves it represents have endomorphism ring isomorphic to $\mathcal{O}$. If the prime ideal over p in $\mathcal{O}$ is principal, then the associated j is $\mathbb{F}_p$-rational; otherwise the two associated j's are $\mathbb{F}_p$ conjugates in $\mathbb{F}_{p^2}$.*

Kohel showed a categorical equivalence between the category whose objects are left ideal classes of a maximal quaternion order and morphisms are ideal homomorphisms, and the category whose objects are supersingular j-invariants and morphisms are isogenies [64, §5.3]. Specifically, let k be a field of characteristic p with q element, let $\mathcal{O}$ be a maximal order in $\mathbb{B}_{p,\infty}$ with an element of reduced norm q, and let E_0/k be a supersingular elliptic curve with endomorphism ring isomorphic to $\mathcal{O}$. Let S_k be the category with objects (E, π) where E is a supersingular elliptic curve, π is the Frobenius endomorphism relative to k, and morphisms

$$(E_1, \pi_1) \to (E_2, \pi_2)$$

are homomorphisms $\psi : E_1 \to E_2$ such that $\psi \circ \pi_1 = \pi_2 \circ \psi$. Let $M_{\mathcal{O},q}$ be the category with objects consisting of (I, ϕ) where I is a projective right module of rank one over $\mathcal{O}$, ϕ an endomorphism of I of reduced norm q, and morphisms

$$(I_1, \phi_1) \to (I_2, \phi_2)$$

consisting of homomorphisms $\psi : I_1 \to I_2$ such that $\psi \circ \phi_1 = \phi_2 \circ \psi$. Then the functor I from S_k to $M_{\mathcal{O},q}$, taking (E, π) to $(\mathrm{Hom}(E_0, E), \tau_\pi)$ where τ_π is the homomorphism of $\mathrm{Hom}(E_0, E)$ to itself given by left composition by π is a full and faithful equivalence from S_k to $M_{\mathcal{O},q}$ [64, Theorem 45, Proposition 49].

This correspondence leads to a graph similar to the isogeny graph; the **graph of maximal orders** in $\mathbb{A}$ with a vertex set of maximal orders up to type, and an edge set of ideals with norm ℓ up to equivalence. When the correspondence between an ideal and an isogeny is known, the ideal norm can be computed as the degree of the isogeny. A recent result has shown that given two maximal orders, $\mathcal{O}_0$ and $\mathcal{O}_1$, one can efficiently (assuming GRH) find a smooth ideal I whose left order is $\mathcal{O}_0$ and whose right order is $\mathcal{O}_1$ [65]. Further analysis of this algorithm [53] has shown that if the endomorphism rings of the supersingular elliptic curves E_0 and E_1 are known and embedded as maximal orders into the same quaternion algebra, then the previously mentioned algorithm is likely to solve the isogeny problem underlying the supersingular isogeny-based key agreement protocol SIDH. Therefore, improvements to endomorphism ring construction algorithms lead to faster attacks on SIDH. An even more recent result showed that computing the endomorphism ring leads to breaking the CSIDH cryptosystem as well [24].

5.3 Algorithms for Computing Endomorphisms

There have been few results on computing endomorphisms of a fixed supersingular j-invariant. The first was the PhD thesis of Kohel [64, §7]. As there is an isomorphism between $\mathrm{End}(E)$ and some maximal quaternion order, an endomorphism α can be viewed as a quaternion. Kohel then proves that the class number of $\mathbb{Z}[\alpha]$ is equal to the number of isomorphism families of supersingular elliptic curves (counted with multiplicity) containing an endomorphism with both trace and norm equal to that of α. This motivates the following theorem.

Theorem 5.3.1. *[64, 84] Let α and β be two endomorphisms of a supersingular E that intersect only at E. Suppose that their norms are ℓ^{h_1} and ℓ^{h_2} respectively, where h_1 is the class number of $\mathbb{Z}[\alpha]$ and h_2 is the class number of $\mathbb{Z}[\beta]$. Then End(E) is determined uniquely by the embedding of $\mathbb{Z}\langle\alpha, \beta\rangle$.*

This theorem in turn motivates an algorithm for computing four endomorphisms which are linearly independent over $\mathbb{Z}$. The idea is to construct the entire supersingular ℓ-isogeny graph for some small integer ℓ and then to use brute force to find two distinct cycles, both containing the vertex $j(E)$, corresponding to endomorphisms α and β that satisfy the class group conditions of the above theorem. The runtime of this algorithm is $O(p^{3/2+\epsilon})$ and $O(p^{1+\epsilon})$, for any $\epsilon > 0$, for the deterministic and the probabilistic versions respectively.

In Eisenträger et al. [45, Proposition 8] a heuristic transformation was presented for turning an algorithm which finds endomorphisms into an algorithm which determines the entire endomorphism ring. The analysis states that $\mathrm{End}(E)$ can be output after $O(\log p)$ calls to the endomorphism finding algorithm. This implies that the algorithms of Kohel will determine $\mathrm{End}(E)$ completely after $O(\log p)$ iterations.

Cervino [25] applied the theory of ternary quadratic forms associated to the norm of a maximal order to construct a $\Omega(p^{2+\epsilon})$ time algorithm for constructing a correspondence between *every* supersingular j-invariant and *every* maximal order type in characteristic p.

McMurdy [71] gave an algorithm which computes an isogeny ϕ from E to an elliptic curve E' with a known endomorphism ring. This can be used to determine $\mathrm{End}(E)$ via Kohel or Deuring's correspondence. The ring $\mathrm{End}(E')$ will correspond to a maximal order $\mathcal{O}' \subset \mathbb{B}_{p,\infty}$, and ϕ will correspond to a right-ideal of $\mathcal{O}'$, some I. The maximal order $\mathcal{O}$ corresponding to $\mathrm{End}(E)$ can then be computed as the left order of I, a computation which can be done efficiently with linear algebra. The runtime of this algorithm has not been publicly examined.

Delfs and Galbraith [41] gave a $O(p^{1/4})$ algorithm for constructing an isogeny between two supersingular elliptic curves defined over $\mathbb{F}_p$, the base field, with logarithmic space required. This can be turned into a $O(p^{1/2})$ algorithm for constructing endomorphisms of an elliptic curve E by finding two paths from E to elliptic curves with j-invariants defined in $\mathbb{F}_p$, and then solving the isogeny problem between them with the algorithm of Delfs and Galbraith. The degree of the output endomorphism will be smooth, but not a prime power. The transformation of [45] can then be used to determine $\mathrm{End}(E)$ with $O(\log p)$ calls to the above endomorphism constructing algorithm.

A second application of the Delfs and Glabraith algorithm for constructing an isogeny between two elliptic curves defined over $\mathbb{F}_p$ is as follows: find one path from E to an elliptic curve E_0 with j-invariant in $\mathbb{F}_p$, then solve for an isogeny between E_0 and one of the elliptic curves with a known endomorphism ring which also has j-invariant in $\mathbb{F}_p$. The composition of the two isogenies gives one from E to one with a known endomorphism ring. Then, similar to McMurdy, $\mathrm{End}(E)$ can be computed from this data. Algorithms have been presented for this task when the paths are smooth [54] or a power of ℓ [40]. The runtime analysis for this combination of ideas has not been yet examined to the best of my knowledge.

The following theorem from Bank et al. [7] is also relevant to the study of constructing $\mathrm{End}(E)$ from finding cycles. Let $E(j)$ denote an elliptic curve with j-invariant j.

Theorem 5.3.2. *[7, 5.1] Suppose two cycles in the ℓ-isogeny graph $G(p,\ell)$ both contain the same path between vertices $E(j_1)$ and $E(j_2)$. Let α and β be the corresponding endomorphisms of $E(j_1)$. If the path between $E(j_1)$ and $E(j_2)$ passes through additional vertices, or if $j_1^p \neq j_2$, then $\{1, \alpha, \beta, \alpha\beta\}$ is not a basis for* $\mathrm{End}(E(j_1))$.

5.3.1 Proposed Algorithm

From Theorem 5.3.1 we see that if a j-invariant j_1 is a part of a cycle in the ℓ-isogeny graph corresponding to an endomorphism α, but the Galois conjugate of j_1 is not in the same cycle, then $\log_\ell \deg \alpha$ cannot be equal to the class number of $\mathbb{Z}[\alpha]$. We prove this in Theorem 5.3.5. This motivates an algorithm for finding cycles such that for all j-invariants in the output cycle, the Galois conjugate of that j-invariant is also in the cycle. We describe such an algorithm in this subsection.

Let $\sigma \in \mathrm{Gal}(\mathbb{F}_{p^2}/\mathbb{F}_p)$ be non-trivial and denote $\sigma(j)$ by $\overline{j}$.

Lemma 5.3.3. *Let $j_1,\ j_2$ be ℓ-isogenous j-invariants of two supersingular elliptic curves. Then $\overline{j_1}$ and $\overline{j_2}$ are also ℓ-isogenous.*

Proof. Suppose E_i has j-invariant j_i. Let $G \subset E_1$ such that $\phi : E_1 \to E_2$ with $\ker \phi = G$, $|G| = \ell$. Denote by π the p^{th} Frobenius map of E_1. Then $\pi(G) \subset E_1^{(p)}$, and $\phi^{(p)} : E_1^{(p)} \to E_2^{(p)}$ with degree ℓ and kernel $\pi(G)$. Noting that $j(E_i^{(p)}) = \overline{j(E_i)}$ concludes the proof. □

Note that the j_i's above can be either in $\mathbb{F}_p$ or $\mathbb{F}_{p^2}$. This leads to two non-trivial cases. If both j_i's are in $\mathbb{F}_{p^2}$, then their conjugates are also ℓ-isogenous to one another. If $j_1 \in \mathbb{F}_p$ is ℓ-isogenous to $j_2 \in \mathbb{F}_{p^2}/\mathbb{F}_p$, then j_1 is also ℓ-isogenous to $\overline{j_2}$.

Lemma 5.3.4. *Let C be a cycle in the supersingular ℓ-isogeny graph corresponding to an endomorphism α of a vertex j. There exists a cycle in the supersingular ℓ-isogeny graph $\overline{C}$ whose ordered vertex set is exactly the conjugates of the ordered vertex set of C. The cycle $\overline{C}$ corresponds to an endomorphism β of the vertex $\overline{j}$. Further, the degree and trace of β are equal to the degree and trace of α, respectively.*

Proof. Let $\alpha \in \mathrm{End}(E)$ where $j(E) = j$. The existence of $\overline{C}$ follows from Lemma 5.3.3, and by that proof it follows that $\ker \beta = \pi_E(\ker \alpha)$. We can identify β with an element of the endomorphism algebra $\mathrm{End}(E^{(p)}) \otimes \mathbb{Q}$. We claim that in this algebra, $\beta = \dfrac{\pi_E \circ \alpha \circ \pi_{E^{(p)}}}{p}$. As $\ker\left(\dfrac{\pi_E \circ \alpha \circ \pi_{E^{(p)}}}{p}\right) = \pi_E(\ker \alpha)$ and $\deg \beta = \deg \dfrac{\pi_E \circ \alpha \circ \pi_{E^{(p)}}}{p} = \deg \alpha$, this claim follows, as does this Lemma's claim on the degree of β. It remains to show that $Tr(\beta) = Tr(\alpha)$. Observe,

$$\begin{aligned} Tr(\beta) = Tr\left(\frac{\pi_E \circ \alpha \circ \pi_{E^{(p)}}}{p}\right) &= \frac{1}{p}Tr(\pi_E \circ \alpha \circ \pi_{E^{(p)}}) = \frac{1}{p}(\pi_E \circ \alpha \circ \pi_{E^{(p)}} + \pi_E \circ \hat{\alpha} \circ \pi_{E^{(p)}}) \\ &= \frac{1}{p}\pi_E \circ Tr(\alpha) \circ \pi_{E^{(p)}} = \frac{Tr(\alpha)}{p}\pi_E \circ \pi_{E^{(p)}} = Tr(\alpha). \end{aligned}$$

□

Theorem 5.3.5. *Let α be an endomorphism of a supersingular j-invariant j_0, corresponding to a cycle in the ℓ-isogeny graph. Suppose the cycle of α contains some j-invariant j', but does not contain $\overline{j'}$. Then, $|Cl(\mathbb{Z}[\alpha])| > \log_\ell(\deg\alpha)$.*

Proof. Let C be the cycle in the ℓ-isogeny graph corresponding to α. We know that C contains at least one vertex in $\mathbb{F}_{p^2}/\mathbb{F}_p$, namely j'. Let $\overline{\alpha}$ be the endomorphism from Lemma 5.3.4, with corresponding cycle $\overline{C}$. As C contains some vertex in $\mathbb{F}_{p^2}/\mathbb{F}_p$, it follows that $C \neq \overline{C}$.

The class number of $\mathbb{Z}[\alpha]$ counts the number of vertices in the ℓ-isogeny graph which are contained (with multiplicity) on a cycle with degree and trace equal to that of α. We have exhibited above a cycle, $\overline{C}$, which shares these properties with C, but is not equal to C. Therefore, $|\mathrm{Cl}\,(\mathbb{Z}[\alpha])\,| \geq 2\log_\ell(\deg\alpha)$. □

Corollary 5.3.6. *Let α be an endomorphism of a supersingular elliptic curve E with j-invariant j_0, corresponding to a cycle in the ℓ-isogeny graph. Suppose the cycle of α contains some j-invariant j', but does not contain $\overline{j'}$. Then, $\forall\beta \in \mathrm{End}(E)$ there exist at least two maximal orders of $\mathbb{B}_{p,\infty}$, one of which is isomorphic to* End(E)*, which contain an embedding of $\mathbb{Z}\langle\alpha,\beta\rangle$.*

The proposed algorithm of this section uses the following idea. Let $E(\mathbb{F}_{p^2})$ be an arbitrary supersingular elliptic curve, with j-invariant j_0, and let ℓ be any prime other than p. We define the *terminating vertices* of the supersingular ℓ-isogeny graph to be those j-invariants that satisfy either

i) $j \in \mathbb{F}_p$, or

ii) $j \in \mathbb{F}_{p^2}/\mathbb{F}_p$, and j is adjacent to $\overline{j}$ in the ℓ-isogeny graph.

Let $P_1 = [j_0, j_1, \ldots, j_k]$ be a non-backtracking path in the ℓ-isogeny graph starting from j_0 and ending at some terminating vertex j_k and containing no other terminating vertices. If $j_k \in \mathbb{F}_p$, then define $P_2 = [\overline{j}_{k-1}, \ldots, \overline{j}_1, \overline{j}_0]$. By Lemma 5.3.3 P_1 concatenated with P_2 still represents a walk in the isogeny-graph, and since there are no terminating vertices in P_1 other than j_k we see the concatenation $P_1\|P_2$ is in fact a path. If $j_k \notin \mathbb{F}_p$, then define $P_2 = [\overline{j}_k, \overline{j}_{k-1}, \ldots, \overline{j}_1, \overline{j}_0]$. Similarly, the concatenation $P_1\|P_2$ is a non-backtracking path from j_0 to its conjugate.

One can repeat this process to find a vertex disjoint path $P_3\|P_4$ from $P_1\|P_2$. Then the path $P_1\|P_2\|\widehat{P_4}\|\widehat{P_3}$ represents an endomorphism α of E, where $\widehat{P_i}$ denotes the vertices of P_i

taken in reverse order. See Algorithm 5.3.7 for the formal treatment. If the paths P_1 and P_3 are obtained using depth-first search, the storage cost can be kept low, so we examine the expected length of a path before a terminating vertex is found so that depth-first seach can be used (see Section 5.4). The expected length (by Theorem 5.4.4) of the cycles will also be examined in Section 5.4.

Algorithm 5.3.7.
Input: *Primes p and $\ell \neq p$, a bound $B > 0$, and a supersingular $j \in \mathbb{F}_{p^2}$.*

Output: *a simple cycle in the ℓ-isogeny graph containing j.*

```
$P_1 \leftarrow$ Depth-first search of depth $B$ terminating at a vertex of Type (i) or (ii)
If $P_1[\#P_1] \in \mathbb{F}_p$, then
    $P_2 \leftarrow [\overline{P_1[\#P_1 - 1]}, \dots, \overline{P_1[1]}, \overline{j}]$
Else
    $P_2 \leftarrow [\overline{P_1[\#P_1]}, \dots, \overline{P_1[1]}, \overline{j}]$
$P_3 \leftarrow$ Depth-first search of depth $B$, avoiding $P_1 \cup P_2$, terminating at a vertex of Type (i) or (ii)
If $P_3[\#P_3] \in \mathbb{F}_p$, then
    $P_4 \leftarrow [\overline{P_3[\#P_3 - 1]}, \dots, \overline{P_3[1]}, j]$
Else
    $P_4 \leftarrow [\overline{P_3[\#P_3]}, \dots, \overline{P_3[1]}, j]$
Return $P_1 \| P_2 \| P_3 \| P_4$
```

Algorithm 5.3.7 can easily be adapted to exclude certain vertices. For instance, when called a second time to find an endomorphism independent from the first output, a list of j-invariants can be passed and ignored in the two steps which perform depth-first searches.

As proposed, the cycles output by the above algorithm do not backtrack and ensures that if any j-invariant is in the cycle, then so is its complex conjugate. This suggests that fewer endomorphisms output from this algorithm are needed to construct the full endomorphism ring than generic endomorphisms.

5.4 Runtime Analysis

We now attempt to analyze the runtime of Algorithm 5.3.7. In particular, we need the quantity of terminating vertices. Then, under the assumption that they are uniformly distributed, we can estimate the value of the bound B for the depth-first searches in Steps 1 and 6, and we can calculate the expected degree of the output endomorphisms. In fact, by the recent analysis of Arpin et al. [3, §4.3], the j-invariants in $\mathbb{F}_p$ will be discovered by a depth-first search in fewer steps than expected for a random subgraph due to their additional structure.

First, we recall the number of supersingular elliptic curves with j-invariants defined over $\mathbb{F}_p$, as these are the first type of terminating vertex. Let S_p be this set of supersingular j-invariants defined over $\mathbb{F}_p$.

Lemma 5.4.1. *[41, Eq.1] [37, 14.18] Let $h(d)$ be the class number of the imaginary quadratic field $\mathbb{Q}(\sqrt{d})$. The number of supersingular j-invariants in $\mathbb{F}_p$ is*

$$\begin{cases} \frac{1}{2}h(-4p) & p \equiv 1 \mod 4 \\ h(-p) & p \equiv 7 \mod 8 \\ 2h(-p) & p \equiv 3 \mod 8 \end{cases}.$$

The class number can be bounded [30, Exercise 5.27 (b)] by $h(d) < \frac{1}{\pi}\sqrt{|D|}\ln|D|$, where D is the discriminant of the imaginary quadratic field. Stricter asymptotic bounds are known assumng a generalized Riemann hypothesis on Dirichlet L-functions [70], with unknown constant c_2, where γ is the Euler–Mascheroni constant:

$$\left(\frac{(1+o(1))\pi}{12e^{\gamma}}\right)\frac{\sqrt{|D|}}{\log\log|D|} \leq h(D) \leq c_2\sqrt{|D|}\log\log|D|.$$

From this, we see that $|S_p| = O\left(\sqrt{p}\log(\log(p))\right)$, and $|S_p| = \Omega\left(\frac{\sqrt{p}}{\log(\log(p))}\right)$. See Figures 5.1, 5.2 for experimental evidence. Either $h(-p)$ or $h(-4p)$ is used depending on the congruence of $p \mod 4$.

The next value we need to bound is the number of supersingular j-invariants which are ℓ-isogenous to their own conjugate j-invariant. This is related to the questions posed in Arpin et al. [3]; specifically Theorem 5.4.2 answers the lower-bound part of [3, Question 3].

Figure 5.1: Imaginary quadratic class numbers with lower bound

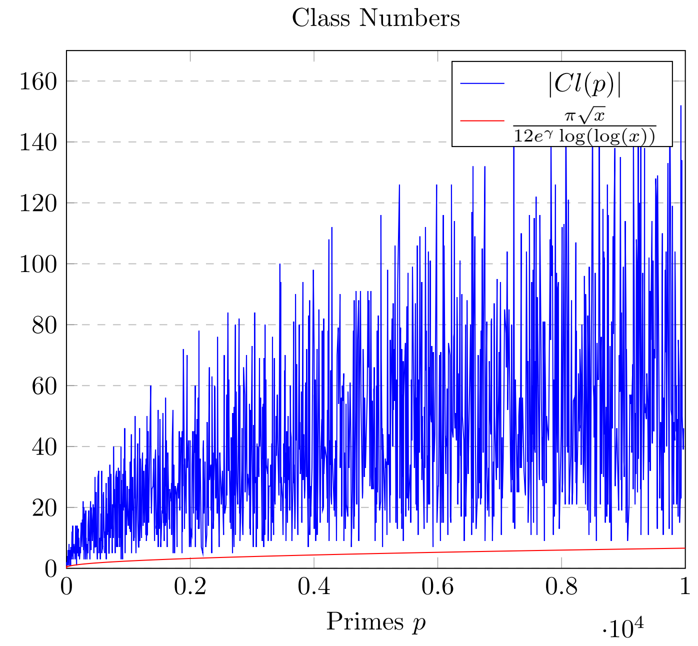
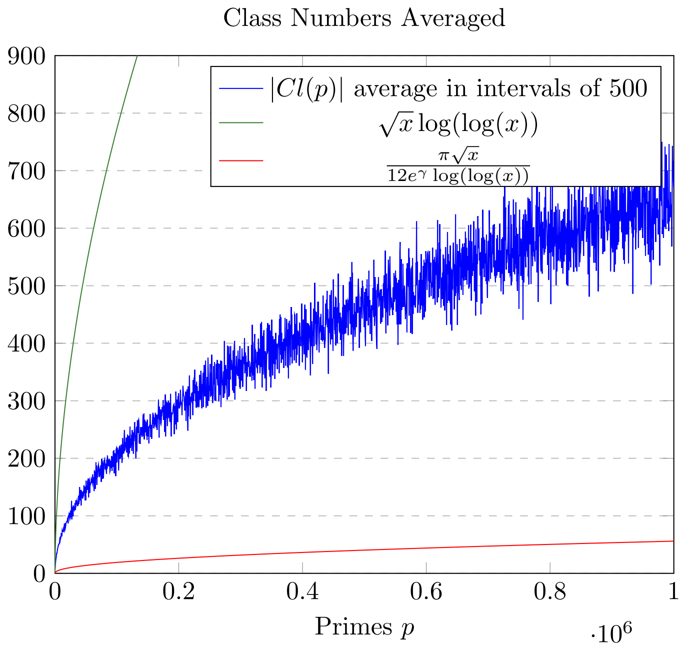


Figure 5.2: Averages of imaginary quadratic class numbers



Theorem 5.4.2. *Assume $\ell < p/4$. Let*

$$\overline{S} = \{j \in \mathbb{F}_{p^2}/\mathbb{F}_p : j \textit{ is supersingular and } \ell\textit{-isogenous to } \overline{j}\}.$$

Under GRH, there exists a constant $C > 0$ (depending on ℓ) such that $|\overline{S}| > C\dfrac{\sqrt{p}}{\log(\log(p))}$.

The proof of this Theorem is not my work, and therefore is omitted. See the proof of Theorem 3.9 in Eisentraeger et al. [46].

An upper bound on $|\overline{S}|$ is given in Lemma 6 of Charles, Goren, and Lauter [26] as $\sqrt{\ell} \cdot \tilde{O}\left(\sqrt{p}\right)$, or more specifically as $\sqrt{\ell}|\mathcal{CL}(p)|$. From this, we see that $|\overline{S}| = O\left(\sqrt{\ell p}\log(\log(p))\right)$, and $|\overline{S}| = \Omega\left(\frac{\sqrt{p}}{\log(\log(p))}\right)$. Experimental data can be found in §5 of Arpin et al. [3].

By combining Theorem 5.4.2 and Lemma 5.4.1 we can compute the asysmptotic growth of the set of terminating vertices. Let $S = S_p \cup \overline{S}$ be the set of terminating vertices. Then

$$\begin{aligned} |S| &= O\left(\sqrt{\ell p}\log(\log(p))\right), \\ |S| &= \Omega\left(\frac{\sqrt{p}}{\log(\log(p))}\right). \end{aligned}$$

We appeal to the following result on Ramanujan graphs to determine the expected depth of a walk in the ℓ-isogeny graph before a terminating vertex is found by depth-first search, as this is the value of B in Algorithm 5.3.7.

Proposition 5.4.3. *Let $p > 3$ be prime, and let $\ell \neq p$ also be prime. Let S be any subset of vertices of the ℓ-isogeny graph over $\overline{\mathbb{F}}_p$ not containing 0 or 1728. Then a random walk of length at least*

$$\frac{\log\left(\dfrac{p}{6\sqrt{|S|}}\right)}{\log\left(\dfrac{\ell+1}{2\sqrt{\ell}}\right)}$$

lands in S with probability at least $\frac{6|S|}{p}$.

Proof. See Lemma 2.1 of Jao, Miller, Venkatesan [60]. □

A quick calculation with the set S of terminating vertices shows that, assuming ℓ is a constant, a walk of length approximately

$$\log\left(\frac{p^{3/4}}{\sqrt{\log(\log(p))}}\right)$$

will end in the set S with probabilty at least

$$\left(\sqrt{p}\log(\log(p))\right)^{-1}.$$

We can now prove the runtime of, and expected degree of endomorphisms output by, Algorithm 5.3.7.

Theorem 5.4.4. *Let j be the j-invariant of a supersingular elliptic curve $E/\overline{\mathbb{F}}_p$, and set*

$$B = \Theta\left(\log\left(\frac{p^{3/4}}{\sqrt{\log(\log(p))}}\right)\right).$$

Under GRH, repeating Algorithm 5.3.7 with an excluded list of j-invariants, and with input B above computes two cycles in the ℓ-isogeny graph through j that generate an order in the endomorphism ring of E in time $O\left(\sqrt{p}(\log p)^2\right)$, as long as the two cycles do not pass through the vertices 0 or 1728, which occurs with probability $1 - O(1/p)$. The algorithm requires polylog p space.

The proof of this Theorem is not my work, and therefore is omitted. See the proof of Theorem 3.7 in Eisentraeger et al. [46].

In light of Theorem 5.4.4, an important question to ask is: once an order of $\mathrm{End}(E)$ is found, how can the entirety of $\mathrm{End}(E)$ be computed? In [45, Proposition 8], they give a heuristic argument as to why $O(\log p)$ endomorphisms are sufficient to determine the whole of $\mathrm{End}(E)$. In [46], the authors examine the constructed order locally, determine a finite list of maximal orders which may be $\mathrm{End}(E)$, and then exhaustively check which is the correct endomorphism ring of E using the methods of Galbraith, Petit, and Silva [54]. The result of [46] is that heuristically only a constant number of calls to Algorithm 5.3.7 is sufficient to then construct $\mathrm{End}(E)$. This reduces the runtime of computing the endomorphism ring from a subring by a factor of $\log p$.

To conclude this chapter we present Table 5.1 comparing the current known algorithms for computing endomorphisms.

Table 5.1: Algorithms computing End(E)

Algorithm	Output	Runtime	Space Required
Kohel [64]	a suborder of End(E)	$O(p^{1+\epsilon})$	$O(p)$
Cervino [25]	End(E) for all E	$\Omega(p^{2+\epsilon})$	?
McMurdy [71]	End(E)	?	?
Delfs, Galbraith [41]	some $\alpha \in$ End(E)	$O(p^{1/2}(\log p)^2)$	polylog p
[41] and [45]	End(E)	$O(p^{1/2}(\log p)^3)$	polylog p
This work	a suborder of End(E)	$O(p^{1/2}(\log p)^2)$	polylog p
Eisenträger et al. [46]	End(E)	$O(p^{1/2}(\log p)^2)$	polylog p

Chapter 6

The Security of Endomorphism-Based Key Establishment

Portions of this chapter were presented in the poster session of MathCrypt 2018. The contents of this chapter and that work are entirely my own.

Recently three Diffie–Hellman type key-exchange protocols based on elliptic curve endomorphism rings were proposed by Daghigh, Khodakaramian Gilan, and Seifi Shahpar [38]. The authors show how their novel idea can be applied to either ordinary or supersingular elliptic curves over a finite field, which is what differentiates the first two methods.

We show that each of these proposed schemes are either insecure (classically in two cases, and using quantum computations in the third) or impractical, depending on parameter selection. In particular, the protocols use a torsion point $P \in E[n]$, and we show that the shared key can be efficiently computed when n is a smooth integer and that the key-exchange protocols are infeasible otherwise.

6.1 Introduction

Building off the idea of supersingular isogeny-based key-exchange [47], the creators of these new schemes [38] propose using the image of elliptic curve torsion points under random endomorphisms to create a random walk in the isogeny graph, with the goal of creating a key-exchange resistant to quantum attacks. The first method uses endomorphisms of an ordinary elliptic curve, the second uses endomorphisms of a supersingular elliptic curve, and the third uses isogenies between two supersingular elliptic curves.

Previously known cryptanalysis did not appear to immediately apply to these schemes. The security appeared to be based on the weaker assumption of computing the endomorphism ring of supersingular or ordinary elliptic curves over a finite field. While they were never implemented publicly, a preliminary assessment would indicate no worse runtime or key size than SIDH for the second and third methods, which would suggest that these schemes may be valid post-quantum candidates.

Section 6.2 describes the protocols and the implementation details as stated in the original work. Section 6.3 then cryptanalyzes the three proposed methods. The work is divided into two cases, depending on whether or not the order of a globally published elliptic curve point is smooth.

We show that the second and third proposed methods can be broken by classical attacks when the torsion point has smooth order, and otherwise it is computationally infeasible for the honest participants to determine the shared key. For the remaining method, again depending on the smoothness of the torsion points order, there is either a classical attack with a polynomial running time or a quantum polynomial-time attack (both running times are in terms of the logarithm of the characteristic of the base field).

6.2 Protocol Overview

In this section we briefly review the three key-exchange methods [38] that will later be cryptanalyzed. As stated in the original work, the first and second protocol are essentially the same but for ordinary and supersingular elliptic curves respectively, while the third generalizes the second.

Method 1: Let E be an ordinary elliptic curve over a finite field $\mathbb{F}_q$, π_q the q^{th} Frobenius endomorphism, and $P \in E[n]$ for some integer n. Alice's private key is $m_0, m_1 \in \mathbb{Z}$, and Bob's is $n_0, n_1 \in \mathbb{Z}$. Their corresponding public keys are $\phi_A(P) = ([m_0] + [m_1]\pi_q)(P)$ and $\phi_B(P) = ([n_0] + [n_1]\pi_q)(P)$, respectively. The points $\phi_A(\phi_B(P))$ and $\phi_B(\phi_A(P))$ are equal because E is ordinary, and so this point is used as the shared key:

$$s = \phi_A(\phi_B(P)) = \phi_B(\phi_A(P)).$$

Method 2: Let E be a supersingular elliptic curve over a finite field $\mathbb{F}_q$, $\{1, \alpha_1, \alpha_2, \alpha_3\}$ a set of generators of $\text{End}(E)$, and $P \in E[n]$ for some integer n. Alice's private key is $m_0, m_1, m_2, m_3 \in \mathbb{Z}$, and Bob's is $n_0, n_1, n_2, n_3 \in \mathbb{Z}$. Their corresponding public keys are $\phi_A(P) = ([m_0] + [m_1]\alpha_1 + [m_2]\alpha_2 + [m_3]\alpha_3)(P)$ and $\phi_B(P) = ([n_0] + [n_1]\alpha_1 + [n_2]\alpha_2 + [n_3]\alpha_3)(P)$, respectively. The points $\phi_A(\phi_B(P))$ and $\phi_B(\phi_A(P))$ are no longer equal because E is supersingular, so the endomorphisms do not commute in general. However

$$E/\langle P,\ \widehat{\phi}_A(\phi_B(P))\rangle \cong E/\langle P,\ \widehat{\phi}_B(\phi_A(P))\rangle$$

by [38, Prop. 3.1]. Alice and Bob can then use the j-invariant as their shared key:

$$s = j(E/\langle P,\ \widehat{\phi}_A(\phi_B(P))\rangle) = j(E/\langle P,\ \widehat{\phi}_B(\phi_A(P))\rangle).$$

Method 3: Let E be a supersingular elliptic curve over a finite field $\mathbb{F}_q$, $\{\phi_0, \phi_1, \phi_2, \phi_3\}$ a set of isogenies in $\mathrm{Hom}(E, E')$, and $P \in E[n]$ for some integer n (see below for details). Alice's private key is $m_0, m_1, m_2, m_3 \in \mathbb{Z}$, and Bob's is $n_0, n_1, n_2, n_3 \in \mathbb{Z}$. Their corresponding public keys are $\phi_A(P) = ([m_0]\phi_0 + [m_1]\phi_1 + [m_2]\phi_2 + [m_3]\phi_3)(P)$ and $\phi_B(P) = ([n_0]\phi_0 + [n_1]\phi_1 + [n_2]\phi_2 + [n_3]\phi_3)(P)$, respectively. Then once again,

$$E/\langle P,\ \widehat{\phi}_A(\phi_B(P))\rangle = E/\langle P,\ \widehat{\phi}_B(\phi_A(P))\rangle,$$

by [38, Prop. 3.1]. Therefore both Alice and Bob can compute the shared key

$$s = j(E/\langle P,\ \widehat{\phi}_A(\phi_B(P))\rangle) = j(E/\langle P,\ \widehat{\phi}_B(\phi_A(P))\rangle).$$

To construct the four isogenies in the setup of method 3 the authors propose using any isogeny $\psi : E \to E'$ and the set $\{1, \alpha_1, \alpha_2, \alpha_3\}$ of generators for $\mathrm{End}(E)$, then publishing

$$\phi_0 = \psi,\ \phi_1 = \psi \circ \alpha_1,\ \phi_2 = \psi \circ \alpha_2,\ \phi_3 = \psi \circ \alpha_3.$$

The integer n is given no restrictions in any of the methods. The private keys in each method are integers, but only their values modulo n are relevant, and therefore can be sampled from $\mathbb{Z}/n\mathbb{Z}$. In Section 6.3 we show these protocols are either insecure or infeasible in practice.

6.3 Key-Only Cryptanalysis

We divide the cryptanalysis of the three protocols into two parts depending on whether the parameter n is a smooth integer or not. In each case, we only need the public-key in each scheme to determine the private key.

Note that in methods 2 and 3 we can assume the factorization of n is public. This is because in these two schemes the shared key requires Alice to compute an isogeny of degree n:

$$E/\langle P\rangle \to (E/\langle P\rangle)\,/\langle\widehat{\phi}_A \circ \phi_B(P)\rangle.$$

Regardless, it is easy to determine if n is smooth.

6.3.1 Smooth n

We first assume the integer n is a smooth integer. Write $n = \prod_{i=1}^{k} \ell_i^{e_i}$ for prime numbers ℓ_i less than some bound B. We show how, given two public keys, one can efficiently compute the shared key in each of the three key-exchange methods. First, we look at method 2.

Recall that in method 2 Alice's private key is $m_0,\ m_1,\ m_2,\ m_3 \in \mathbb{Z}$ and her public key is $\phi_A(P) = ([m_0] + [m_1]\alpha_1 + [m_2]\alpha_2 + [m_3]\alpha_3)(P)$. The endomorphisms α_i are given in the protocol setup, $0 \leq i \leq 3$. A passive adversary therefore knows the points $P,\ \alpha_1(P),\ \alpha_2(P),\ \alpha_3(P)$, and Alice's public key $\phi_A(P)$. Note all of these points still have smooth order. Then for each $1 \leq i \leq k$ the adversary can solve for each $z_0,\ z_1,\ z_2,\ z_3$ modulo $\ell_i^{e_i}$ by applying the Pohlig-Hellman algorithm [79] using the target point $\phi_A(P)$ and the base points $P,\ \alpha_1(P),\ \alpha_2(P),\ \alpha_3(P)$. Once this is solved for each $\ell_i^{e_i}$, the Chinese remainder theorem can be used to determine values $z_0,\ z_1,\ z_2,\ z_3 \in \mathbb{Z}/n\mathbb{Z}$.

Denote by $\mathrm{CRT}(L_1, L_2)$ the Chinese remainder theorem which takes as input two lists of non-negative integers of equal length k, $L_1 = [r_1, \dots, r_k]$ and $L_2 = [q_1, \dots, q_k]$, and outputs the unique non-negative integer z less than $\prod_{i=1}^{k} q_i$ which satisfies $z \equiv r_i \mod q_i$ for each $i = 1, \dots, k$.

Algorithm 6.3.1.

Input: *$P \in E[n],\ \phi_A(P),$ generators $\{1, \alpha_1, \alpha_2, \alpha_3\}$ for End(E), $n = \prod_{i=1}^{k} \ell_i^{e_i}$.*

Output: *$z_0,\ z_1,\ z_2,\ z_3 \in \mathbb{Z}/n\mathbb{Z}$ such that $\phi_A(P) = ([z_0] + [z_1]\alpha_1 + [z_2]\alpha_2 + [z_3]\alpha_3)(P)$.*

```
1. For $i = 1, \dots, k$ :
2.     $R \leftarrow \left[\prod_{r \neq i} \ell_r^{e_r}\right] \phi_A(P)$
3.     $a_i,\ b_i,\ c_i,\ d_i \leftarrow 0$
```

```
    For $j = 1, \dots, e_i$ :
        $Q \leftarrow [\ell_i^{e_i - j}]\,(R - [a_i]P - [b_i]\alpha_1(P) - [c_i]\alpha_2(P) - [d_i]\alpha_3(P))$
        Try $s_j,\ t_j,\ u_j,\ v_j \in \mathbb{Z}/\ell_i\mathbb{Z}$ until:
            $Q = [s_j]P + [t_j]\alpha_1(P) + [u_j]\alpha_2(P) + [v_j]\alpha_3(P)$
        $a_i \leftarrow a_i + s_j \ell_i^{j-1}$
        $b_i \leftarrow b_i + t_j \ell_i^{j-1}$
        $c_i \leftarrow c_i + u_j \ell_i^{j-1}$
        $d_i \leftarrow d_i + v_j \ell_i^{j-1}$
$z_0 \leftarrow CRT(\{a_i\}_{i=1}^{k}, \{\ell_i^{e_i}\}_{i=1}^{k})$
$z_1 \leftarrow CRT(\{b_i\}_{i=1}^{k}, \{\ell_i^{e_i}\}_{i=1}^{k})$
$z_2 \leftarrow CRT(\{c_i\}_{i=1}^{k}, \{\ell_i^{e_i}\}_{i=1}^{k})$
$z_3 \leftarrow CRT(\{d_i\}_{i=1}^{k}, \{\ell_i^{e_i}\}_{i=1}^{k})$
Return $z_0, z_1, z_2, z_3$
```

There is no guarantee that $z_i = m_i$ for any of the $0 \leq i \leq 3$. However, what we have found is some endomorphism

$$\phi_Z = [z_0] + [z_1]\alpha_1 + [z_2]\alpha_2 + [z_3]\alpha_3$$

such that $\phi_Z(P) = \phi_A(P)$. It follows from the definition of a key exchange protocol that if two distinct private keys generate the same public key, then they will agree on every possible shared key. We make this explicit in our case anyway with the following result, which shows that the endomorphism ϕ_Z is enough to determine the key Alice shares with anyone else, say Bob.

Proposition 6.3.2. *Let E be an elliptic curve over a finite field K, $P \in E(K)$, and let ϕ, ϕ_A, ϕ_B be three endomorphisms of E. If $\phi(P) = \phi_A(P)$, then*

$$\langle P,\ \widehat{\phi}(\phi_B(P))\rangle = \langle P,\ \widehat{\phi}_A(\phi_B(P))\rangle.$$

The proof is similar to that of [38, Prop. 3.1].

Proof. Let $\alpha \in \mathrm{End}(E)$ be arbitrary Let $k = \mathrm{Tr}(\alpha) = \alpha + \widehat{\alpha} \in \mathbb{Z}$. Then

$$\langle P,\ \widehat{\alpha}(P)\rangle = \langle P,\ [k]P - \alpha(P)\rangle = \langle P,\ \alpha(P)\rangle.$$

Applying this twice gives

$$\langle P,\ \widehat{\phi}(\phi_B(P))\rangle = \langle P,\ \widehat{\phi}_B(\phi(P))\rangle = \langle P,\ \widehat{\phi}_B(\phi_A(P))\rangle = \langle P,\ \widehat{\phi}_A(\phi_B(P))\rangle. \qquad \square$$

This procedure consists the of Pohlig-Hellman algorithm, the Chinese remainder theorem, and then Velu's formula [95] on a point of order n. All three can be performed in polynomial time in B. This removes the possibility of n being a smooth integer in method 2.

This process can be easily modified to attack method 3 as well. In this case Alice's public key is $\phi_A(P) = ([m_0]\phi_0 + [m_1]\phi_1 + [m_2]\phi_2 + [m_3]\phi_3)(P)$. Since P and each ϕ_i are given in the protocol setup, we simply change the base points when applying the Pohlig-Hellman algorithm to $\phi_i(P) = (\psi \circ \alpha_i)(P)$, $0 \leq i \leq 3$. Similarly, in method 1, the base points are P and $\pi_q(P)$, and we are only looking for two coefficients instead of four. Therefore n should not be smooth in any of the three schemes.

6.3.2 Non-smooth n

We now assume that n is not smooth. Let ℓ be the largest prime factor of n. The order of the point $\phi(P)$ divides n, since $P \in E[n]$. Computing the isogeny, or even the coefficients of the codomain curve for

$$E/\langle P\rangle \to (E/\langle P\rangle)\,/\langle\phi(P)\rangle,$$

requires a summation over $O(\ell)$ many elements of $\mathbb{F}_q$, when using Velu's formula [34], and $O(\sqrt{\ell})$ using the work of Bernstein et al. [9] (see Section 2.1.6).

The shared key must be feasibly computable for honest participants. Therefore $O(\sqrt{\ell})$ is a feasible amount of work in methods 2 and 3, and hence $O(n)$ is feasible as well because by assumption $n = \mathit{poly}(\ell)$. In this case, we may brute force the private key $m_0, m_1, m_2, m_3 \in \mathbb{Z}/n\mathbb{Z}$, searching for the combination that generates Alice's public key. As shown by Proposition 6.3.2 knowledge of the integers modulo n is sufficient to recreate Alice's side of the shared key computation. Brute force of this integer requires $O(n^4)$ operations, and so there is only a polynomial gap between the work done by honest participants and an eavesdropper.

Combining this with the previous subsection shows that methods 1, 2 and 3 are insecure when n is smooth, and Alice and Bob cannot efficiently compute the shared key when n is not smooth in methods 2 and 3. Method 1 however does not rely on an isogeny computation, so we have not yet described a flaw in the case with ordinary elliptic curves and a non-smooth choice for n.

Recall, method 1 has Alice using the private key $m_0, m_1 \in \mathbb{Z}$ and public key $\phi_A(P) = ([m_0]+[m_1]\pi_q)(P) \in E(\mathbb{F}_q)$ for some point $P \in E[n]$. Since the q^{th} Frobenius endomorphism can be evaluated by anyone, the inversion of key-generation can be interpreted as solving a two-dimensional elliptic curve discrete logarithm problem: given $P \in E[n]$ find some integers α_0, α_1 such that

$$\phi_A(P) = [\alpha_0]P + [\alpha_1]\pi_q(P).$$

This discrete log problem is over an Abelian group and so it can be solved in polynomial runtime by a quantum computer using Shor's algorithm [83]. Note that we have no guarantee that $\alpha_i = m_i$ for either $i \in \{0, 1\}$, but we can once again appeal to Proposition 6.3.2 to see this is sufficient information to invalidate the security of this key-exchange scheme.

This work shows that none of the three suggested endomorphism-based key exchange schemes [38] is a candidate for a post-quantum scheme. Further, two of the three methods (2 and 3) are not resistant to classical attacks under the assumptions that the runtime for computing isogenies is asymptotically exponential in the largest prime dividing the size of the kernel (as in Velu's formula), and that honest participants must have the ability to compute the shared key.

Chapter 7

The Security of Supersingular Isogeny Learning with Errors

Portions of this chapter were published by Leonardi and Ruiz-Lopez [68]. The contents of this chapter represent my contribution to that work.

This chapter proposes and subsequently proves the total insecurity of a generalized learning with errors instantiation from supersingular isogenies. We first describe a public-key cryptosystem based on the learning with errors problem, and prove its correctness. We then instantiate the protocol with supersingular isogenies, and prove it to be OW-CPA when it is KP. Lastly, we show that this protocol is not KP against a quantum-enabled adversary.

7.1 Introduction

Lattice-based encryption and digital signature schemes are major candidates in post-quantum cryptography. Regev introduced the Learning with Errors (LWE) problem [80] to cryptography when he described the problem of finding solutions to a system of linear equations with noise, which has become a standard assumption in lattice-based cryptography, along with an IND-CPA public-key cryptosystem. In that work, samples are lattice points, and errors are vectors with small norm.

We generalize the concept of both a lattice and an error. This chapter presents an encryption scheme based on the problem of learning a homomorphism between two algebraic objects when given samples of input with noisy output (that is, the output is modified by some error term). In generalizing the LWE problem, we need a notion of size so that the chosen noise does not completely hide the output, and an efficient way of removing the noise. Our proposal uses finitely generated groups as these have a natural metric (Cayley distance), and we propose using a secret normal subgroup $N \lhd H$ from which to draw noise. A decrypter could then remove noise by computing the quotient H/N.

In the pursuit of examining the security of supersingular isogeny-based cryptosystems, this Chapter presents and cryptanalyzes a new encryption scheme modelled after LWE schemes, but implemented with isogenies. Specifically, we instantiate the idea using supersingular elliptic curves over a finite field with a large characteristic and isogenies with smooth degree as in Jao, De Feo [58]. The isogeny actions then behave like "modding out" by a secret subgroup during decryption. The hope is that the resulting encryption scheme is secure under lattice and isogeny assumptions.

This chapter is organized as follows: Section 7.2 introduces the generic encryption scheme using finitely generated groups and normal subgroups for noise. Section 7.3 then details the supersingular isogeny-based instantiation, along with security proofs and basic analysis. The chapter concludes with Section 7.4 which describes both a classical attack and a quantum attack which totally break the isogeny instantiation.

7.2 Generic Encryption Construction

Let G, H and K be finitely generated groups, and let η and χ be probability distributions over G and H respectively such that both distributions can be sampled from efficiently. The following public-key cryptosystem construction uses integer parameters t and ℓ that will be discussed later.

KeyGen(1^λ) : Choose efficiently computable homomorphisms $\phi : G \to H$ and $\psi : H \to K$. For $i \in \{1, ..., t\}$ compute

$$(g_i, \phi(g_i)h_i) \in G \times H,$$

where g_i is sampled from η and h_i is sampled from χ restricted to $\ker(\psi) \leq H$. The secret key, SK, is a description of ϕ and ψ. The public key, PK, is the set

$$\{(g_i, \phi(g_i)h_i)) : i = 1, ..., t\} \subset G \times H,$$

together with a public element $\tau \in H \setminus \ker(\psi)$.

Enc(μ, PK) : For a message $\mu \in \{0,1\}$ and public key PK, choose a word of length ℓ, $w = w_1 \cdots w_\ell$ uniformly at random from the alphabet $\{1,\ldots,t\}$ and return

$$(g,h) = \left(\prod_{i=1}^{\ell} g_{w_i}, \left(\prod_{i=1}^{\ell} \phi(g_{w_i}) h_{w_i}\right) \tau^{\mu}\right) \in G \times H.$$

Dec(g, h, SK) : Compute $\nu = \psi(\phi(g))^{-1}\psi(h)$ and return 0 if $\nu = 1_K$, and return 1 if $\nu \neq 1_K$.

Correctness: If encryption and decryption are perform correctly, then:

$$\begin{aligned}
\nu &= \psi\left(\phi\left(\prod_{i=1}^{\ell} g_{w_i}\right)\right)^{-1} \psi\left(\left(\prod_{i=1}^{\ell} \phi(g_{w_i}) h_{w_i}\right)\tau^{\mu}\right) \\
&= \left(\prod_{i=1}^{\ell} \psi\left(\phi\left(g_{w_i}\right)\right)\right)^{-1} \left(\prod_{i=1}^{\ell} \psi\left(\phi\left(g_{w_i}\right)\right)\psi\left(h_{w_i}\right)\right)\psi\left(\tau^{\mu}\right) \\
&= \left(\prod_{i=1}^{\ell} \psi\left(\phi\left(g_{w_i}\right)\right)\right)^{-1} \left(\prod_{i=1}^{\ell} \psi\left(\phi\left(g_{w_i}\right)\right)\right)\psi\left(\tau^{\mu}\right) \\
&= \psi\left(\tau^{\mu}\right).
\end{aligned}$$

Since $\tau \notin \ker(\psi)$ and $\mu \in \{0,1\}$, it follows that $\psi(\tau^{\mu}) = 1_K$ if and only if $\mu = 0$.

Observe that since the word w is chosen uniformly at random, the value of ℓ in the interval $[0,\ldots,t]$ will follow a Gaussian distribution.

The underlying computational problems of this encryption scheme are the following.

Problem 7.2.1. *Given a public key of t samples $\{(g_i, \phi(g_i)h_i) : i = 1,\ldots,t\} \subset G \times H$, where $g_i \leftarrow_{\eta} G$ and $h_i \leftarrow_{\chi} \ker(\psi)$, and $\tau \in H \backslash \ker(\psi)$, compute generators for ϕ or ψ.*

Problem 7.2.2. *Given T samples of encryptions of known messages $\mu_j \in \{0,1\}$ with unknown words $w^j = {w^j}_1 \cdots {w^j}_{\ell_j}$ chosen uniformly at random from the alphabet $\{1,\ldots,t\}$:*

$$\left(\prod_{i=1}^{\ell_j} g_{{w^j}_i}, \left(\prod_{i=1}^{\ell_j} \phi(g_{{w^j}_i}) h_{{w^j}_i}\right) \tau^{\mu_j}\right) \in G \times H, \; j = 1,\ldots,T,$$

determine if a given $(g,h) \in G \times H$ is an encryption of some chosen $\mu \in \{0,1\}$ or drawn uniformly at random from $G \times H$ with probability greater than $1/2$.

7.3 Isogeny Encryption Construction

The above scheme can clearly be instantiated with elliptic curves groups over finite fields, and isogenies as the morphisms. That scheme will be presented in this section, along with security proofs, and analysis of key size and computation cost.

7.3.1 Protocol

Some changes have been made when translating the previous framework to that of isogenies with the goal of optimizing the message size/computation cost ratio. To allow for efficient isogeny computations, consider primes of the special form, $p = 2^m 3^n f - 1$, similar to that of supersingular isogeny key exchange [47]. The encrypter uses the 2^m torsion subgroups for efficiency, while the decrypter uses the 3^n torsion subgroups. For convenience, we call the following encryption scheme Supersingular Isogeny Learning with Errors (SILWE).

Setup: Let $p = 2^m 3^n f - 1$ be prime, for any small prime f. Fix a supersingular elliptic curve E_0 with $\#E_0(\mathbb{F}_{p^2}) = (p+1)^2$ and two points $R_0,\ S_0 \in E_0[3^n]$ such that $\langle R_0,\ S_0\rangle = E_0[3^n]$. Fix $t \in \mathbb{N}$.

Key Generation: Choose $(k_1, k_2, k_3, k_4) \in_R (\mathbb{Z}/3^n\mathbb{Z})^4$, where k_1 and k_2 are not both divisible by 3, and the same condition for k_3 and k_4, to be the private key. Compute

$$\phi : E_0 \to E_1, \ker(\phi) = \langle [k_1]R_0 + [k_2]S_0 \rangle,$$

and points $P_1,\ Q_1 \in E_1[2^m],\ R_1,\ S_1 \in E_1[3^n]$ such that $\langle P_1,\ Q_1\rangle = E_1[2^m]$ and $\langle R_1,\ S_1\rangle = E_1[3^n]$. Compute

$$\psi : E_1 \to E_2, \ker(\psi) = \langle [k_3]R_1 + [k_4]S_1 \rangle.$$

Verify that ψ does not backtrack along the path of ϕ. If it does backtrack, then choose new k_3 and k_4 and repeat this last step; otherwise compute $\psi(P_1)$ and $\psi(Q_1)$. Choose $2t$ points at random:

$$X_1, \ldots, X_t \in_R E_0(\mathbb{F}_{p^2}),$$

$$Y_1, \ldots, Y_t \in_R \ker(\psi) \subset E_1[3^n].$$

For each $1 \le i \le t$, compute the image of the X_i's under ϕ. The public key is:

$$P_1,\ Q_1 \text{ and tuples } (X_i,\ \phi(X_i) + Y_i), \quad \text{for } 1 \le i \le t.$$

Encryption: Encode the message M into $(M_1,\ M_2) \in (\mathbb{Z}/2^m\mathbb{Z})^2$, where M_1 and M_2 are not both divisible by 2. Choose a random subset $J \subseteq \{1, \ldots, t\}$, and compute the ciphertext:

$$c = \left(\sum_{i \in J} X_i, \left(\sum_{i \in J} \phi(X_i) + Y_i \right) + [M_1]P_1 + [M_2]Q_1 \right) \in E_0(\mathbb{F}_{p^2}) \times E_1(\mathbb{F}_{p^2}).$$

Decryption: Given a ciphertext $c = (c_1,\ c_2)$, compute

$$Z = \psi(c_2 - \phi(c_1)).$$

Using the knowledge of $\psi(P_1),\ \psi(Q_1)$ solve the two dimensional elliptic curve discrete logarithm problem

$$Z = [M_1']\psi(P_1) + [M_2']\psi(Q_1),$$

for M_1' and M_2'. An efficient algorithm for solving the elliptic curve discrete log problem in two dimension is given in [33, §4, 5]. Then recover the message M from $(M_1',\ M_2')$.

Note that P_1 and Q_1 are independent in $E_1[2^m]$ and ψ has degree 3^n, so $\psi(P_1)$ and $\psi(Q_1)$ are independent in $E_2[2^m]$. Therefore the discrete logarithm exists and is unique, so $(M_1',\ M_2') = (M_1,\ M_2)$.

7.3.2 Security Proofs

The goal of this section is to prove that SILWE is OW-CPA assuming that it is KP. Let λ be the security parameter and select a prime $p = 2^m 3^n f - 1 \approx 2^{6\lambda}$. Let E_0 be as in the setup of SILWE. We start by defining the following distributions.

Definition 7.3.1. *Let $PK = (P_1, Q_1,\ (X_i, \phi(X_i) + Y_i)_i)$ be a public key for SILWE. Let DSI_{PK} be the uniform distribution on the possible encryptions of the message* 0*:*

$$DSI_{PK} = \left\{ \left(\sum_{i \in J} X_i, \sum_{i \in J} \phi(X_i) + Y_i \right) : J \subset \{1, \ldots, t\} \right\}.$$

Definition 7.3.2. *Let $PK = (P_1, Q_1,\ (X_i, \phi(X_i) + Y_i)_i)$ be a public key for SILWE, and let ϕ, ψ be the isogenies derived from the corresponding private key. Let $\overline{DSI}_{PK}$ be the uniform distribution on the following set:*

$$\overline{DSI}_{PK} = \{(X, \phi(X) + Y) : X \in E_0(\mathbb{F}_{p^2}), Y \in \ker(\psi)\}.$$

The following proposition tells us that SILWE is OW-CPA if DSI_{PK} is difficult to distinguish from random. We then prove that DSI_{PK} looks random by reducing the problem to showing $\overline{\mathrm{DSI}}_{PK}$ looks random, which we achieve by reducing the problem to the KP problem.

Proposition 7.3.3. *Let $PK = (P_1, Q_1, (X_i, \phi(X_i) + Y_i)_{i=1}^t)$ be a public key for SILWE. If DSI_{PK} is indistinguishable from the uniform distribution on $E_0(\mathbb{F}_{p^2}) \times E_1(\mathbb{F}_{p^2})$, then encryption is OW-CPA.*

Proof. Suppose the algorithm $\mathcal{A}_{PK}$ can decrypt messages successfully with probability at least $\epsilon > \frac{1}{(\log p)^e}$, for some integer e, and runs in time bounded above by T. Now we define $\mathcal{B}_{PK}$ to be the following algorithm for distinguishing DSI_{PK} from the uniformly random distribution.

Let (c_1, c_2) be a given sample from either DSI_{PK} or uniformly random on $E_0(\mathbb{F}_{p^2}) \times E_1(\mathbb{F}_{p^2})$. Let $r = O((\log p)^\gamma)$, for some $\gamma > e$. Choose r pairs $(u_{1,i}, u_{2,i}) \in_R (\mathbb{Z}/2^m\mathbb{Z})^2$ for $i = 1, \dots, r$ and compute

$$z = (c_1, c_2 + [u_{1,i}]P_1 + [u_{2,i}]Q_1).$$

Then run $\mathcal{A}_{PK}$ on all r pairs. If $\mathcal{A}_{PK}(z) = (u_{1,i}, u_{2,i})$, then $\mathcal{B}_{PK}(c_1, c_2)$ returns DSI_{PK}. Let $\mathcal{B}_{PK}(c_1, c_2)$ return the uniform distribution on $E_0(\mathbb{F}_{p^2}) \times E_1(\mathbb{F}_{p^2})$ otherwise.

Assuming the decryption oracle $\mathcal{A}_{PK}$ returns a random pair when given a non-valid ciphertext, we can compute the probability of success for the distinguisher. $\mathcal{B}_{PK}$ only fails when $\mathcal{A}_{PK}$ fails for all r random pairs, or when $\mathcal{A}_{PK}$ returns a false positive (outputting the correct $(u_{1,i}, u_{2,i})$ by chance). Therefore $\mathcal{B}_{PK}$ fails with probability $(1 - \epsilon)^r + \frac{r}{2^{2m}}$. We have $2^{2m} \approx p$, and by the assumption $\epsilon > \frac{1}{(\log p)^e}$ we have $(1 - \epsilon)^r < (1 - \frac{1}{(\log p)^e})^r$, and so the success probability of $\mathcal{B}_{PK}$ is non-negligible by the choice of r. $\mathcal{B}$ runs in time $\tilde{O}(Tr)$ as all the elliptic curve point addition computations are negligible. □

We now aim to show that DSI_{PK} is indistinguishable from uniform, under the assumption that SILWE is KP.

Lemma 7.3.4. *Let $PK = (P_1, Q_1, (X_i, \phi(X_i) + Y_i)_{i=1}^t)$ be a public key for SILWE. There does not exist a polynomial time algorithm for distinguishing between DSI_{PK} and $\overline{DSI}_{PK}$.*

Proof. Let

$$\tau_X : E(\mathbb{F}_{p^2}) \to E(\mathbb{F}_{p^2}),\ Y \mapsto X + Y,$$

be the *translation-by-X map*. The result follows from the linearity of ϕ, the fact [87, III.3.6]:

$$\forall X \in E(\mathbb{F}_{p^2}),\ \tau_X \text{ is an isomorphism},$$

and that $\ker \psi$ is closed under addition. □

Proposition 7.3.5. *Let $PK = (P_1, Q_1, (X_i, \phi(X_i) + Y_i)_{i=1}^t)$ be a public key for SILWE. If inverting key generation is intractable (that is, finding $\ker(\phi)$ or $\ker(\psi)$ from PK is hard), then DSI_{PK} is indistinguishable from the uniform distribution on $E_0(\mathbb{F}_{p^2}) \times E_1(\mathbb{F}_{p^2})$.*

Proof. Suppose the algorithm $\mathcal{A}_{PK}$ can distinguish between samples from the uniform distribution on $E_0(\mathbb{F}_{p^2}) \times E_1(\mathbb{F}_{p^2})$ and DSI_{PK} with advantage ϵ in time T. By Lemma 7.3.4, $\mathcal{A}_{PK}$ is also a distinguisher for the uniform distribution on $E_0(\mathbb{F}_{p^2}) \times E_1(\mathbb{F}_{p^2})$ and $\overline{\mathrm{DSI}}_{PK}$. Let $(c_1,\ c_2) = (X,\ \phi(X) + Y) \in_R \overline{\mathrm{DSI}}_{PK}$.

We start by choosing a random point $P \in E_0[3^n]$ and finding an integer ℓ such that $[\ell]P \in \ker(\phi),\ 0 \leq \ell < 3^n$. Iterating this procedure correctly will reveal the kernel of ϕ in $\log \lambda$ steps as we will show momentarily. We first demonstrate that evaluating $\mathcal{A}_{PK}$ on the input $(c_1',\ c_2) = (c_1 + [r]P, c_2)$, for $r \in \mathbb{Z}$, leaks information about $\ker(\phi)$ and $\ker(\psi)$. Suppose $[r]P \in \ker(\phi)$. Then,

$$\begin{aligned}(c_1',\ c_2) &= (X + [r]P,\ \phi(X) + Y)\\ &= (X + [r]P,\ \phi(X) + \phi([r]P) + Y)\\ &= (X + [r]P,\ \phi(X + [r]P) + Y) \in \overline{\mathrm{DSI}}_{PK}.\end{aligned}$$

Suppose next that $[r]P \notin \ker(\phi)$. Then,

$$\begin{aligned}(c_1',\ c_2) &= (X + [r]P,\ \phi(X) + Y)\\ &= (X + [r]P,\ \phi(X) + \phi([r]P) - \phi([r]P) + Y)\\ &= (X + [r]P,\ \phi(X + [r]P) + (Y - \phi([r]P))).\end{aligned}$$

Observe that $c_2 - \phi(c_1') = Y - \phi([r]P) \in \ker(\psi)$ if and only if $\phi([r]P) \in \ker(\psi)$. Therefore, when $[r]P \notin \ker(\phi)$, the sample is from $\overline{\mathrm{DSI}}_{PK}$ if and only if $\phi([r]P) \in \ker(\psi)$.

Combining these two cases, we see that if P is added to some c_1 from $(c_1,\ c_2) \in \overline{\mathrm{DSI}}_{PK}$, then the distinguisher $\overline{\mathcal{A}}_{PK}$ returns $\overline{\mathrm{DSI}}_{PK}$ if and only if $P \in \ker(\phi)$ or $\phi(P) \in \ker(\psi)$.

By the definition of ϕ and ψ their kernels are in the 3^n torsion subgroup of their respective elliptic curve domains. Recall that the kernel of the dual isogeny $\hat{\phi}$ is generated by $\phi(E_0[3^n])$, and so if $P \notin \ker(\phi)$, then $\phi(P) \in \ker(\hat{\phi})$. Due to the non-backtracking requirement of ψ in key generation, $\ker(\hat{\phi}) \cap \ker\psi = \{\infty_{E_1}\}$. Therefore, if $P \notin \ker(\phi)$, then $\phi(P) \notin \ker(\psi)$. Hence, on input $(c_1 + P,\ c_2)$, $\mathcal{A}_{PK}$ returns $\overline{\mathrm{DSI}}_{PK}$ if and only if $P \in \ker(\phi)$.

We now explain how to iterate this process to efficiently learn $\ker(\phi)$. The nine 3-torsion points of E_0 can then all be computed from the divison polynomial and checked with $\mathcal{A}_{PK}$ to learn which three (namely ∞ and $\pm[3^{n-1}]([k_1]R_0 + [k_2]S_0)$) are in the kernel of ϕ. This process can then be repeated iteratively to compute $[3^{n-i}]([k_1]R_0 + [k_2]S_0)$ for $i = 1, \ldots, n$ as follows: given the point $[3^{n-i}]([k_1]R_0 + [k_2]S_0)$, find the nine 3^{n-i-1}-torsion points U that satisfy $[3]U = \pm[3^{n-i}]([k_1]R_0 + [k_2]S_0)$, and then use $\mathcal{A}_{PK}$ as above to test which are in the kernel.

After $n \approx \frac{1}{2}\log p$ iterations this process returns a generator for the kernel of ϕ. Note that the samples used can be reused in subsequent steps, and so this procedure succeeds with high probability when $t \geq \frac{1}{\epsilon}$. With ϕ known, ψ can easily be found by constructing the points $c_2 - \phi(c_1) = Y \in \ker(\psi)$. □

Corollary 7.3.6. *If SILWE is KP, and subset sum is intractable on $E_0(\mathbb{F}_{p^2})$ and $E_1(\mathbb{F}_{p^2})$, then SILWE encryption is OW-CPA.*

We later show 7.4 is not KP, and hence not OW-CPA.

7.3.3 Key Sizes and Computation Cost

The work in this subsection follows closely the analysis of the original supersingular isogeny public-key encryption scheme [47]. Further optimizations have been made [33, 34], but as this scheme is shown to be insecure in Section 7.4, that effort will not be put in here.

Key sizes: Private keys can be stored as $2\lceil n \log_2 3\rceil + 2$ bits, since the points $[k_1]R_0 + [k_2]S_0$ can be expressed as $\left[\frac{k_1}{k_2}\right] R_0 + S_0$ or $R_0 + \left[\frac{k_2}{k_1}\right] S_0$. Points can be stored using only the x and z projective coordinates, and x-coordinates are in $\mathbb{F}_{p^2}$ while z-coordinates require only a single bit. Therefore public keys can be stored with $(2t + 2)(2\lceil\log_2 p\rceil + 2)$ bits.

Computation cost: Key generation requires two isogeny computations and two basis generations; see [34, 100] for efficient methods for these tasks. The remaining key generation operations are t point additions and sampling of random points. Encryption needs only approximately $t/2+2$ point additions, making it the most efficient step in this protocol. Decryption requires two isogeny computations and one discrete logarithm computation. Note that the isogenies ϕ and ψ have already been computed by the decrypter, and their images on bases can be precomputed. Therefore, the two isogeny computations in decryption can be substituted for more two discrete logarithms. When this improvement is made, only key generation contains isogeny computations.

7.4 Cryptanalysis

In this section we show that the encryption scheme defined in 7.3 is totally insecure. We divide the cryptanalysis into two parts, depending on the value t (the number of tuples given in the public keys).

First, let us assume that $t \leq 4$. We will show that encyption is not one-way under a key-only attack, i.e. ciphertexts can be decrypted by any eavesdropper with access to the public key. Let

$$c = (c_1, c_2) = \left(\sum_{i \in J} X_i, \left(\sum_{i \in J} \phi(X_i) + Y_i \right) + [M_1]P_1 + [M_2]Q_1 \right)$$

be a ciphertext. Since $t \leq 4$, it follows that there are fewer than 16 possibilities for the subset $J \subseteq \{1, \ldots, t\}$. A full list of possible values for c_1 can then easily be computed, and compared to the ciphertext to determine the subset J used in encryption. The value

$$[M_1]P_1 + [M_2]Q_1 = c_2 - \left(\sum_{i \in J} \phi(X_i) + Y_i \right)$$

can be computed, and the message can be found from this. Hence, when $t \leq 4$, a key-only message-recovery attack can be efficiently launched against this encryption scheme, showing it is not OW-CPA.

Second, and lastly, assume that $t > 4$. Again, we will totally break this encryption scheme by showing that it is not one-way under a key-only attack. In this case, we will recover part of the private key, $\ker \psi$, which enables decryption.

Recall,

$$E_0(\mathbb{F}_{p^2}) \cong \mathbb{Z}/2^m\mathbb{Z} \times \mathbb{Z}/2^m\mathbb{Z} \times \mathbb{Z}/3^n\mathbb{Z} \times \mathbb{Z}/3^n\mathbb{Z}.$$

It follows that for any points $A_1, \ldots, A_t \in E_0(\mathbb{F}_{p^2})$, there exists a non-trivial solution to

$$a_1, \ldots, a_t \in \mathbb{Z} : [a_1]A_1 + \cdots + [a_t]A_t = \infty,$$

because $t > 4$. Consider the function

$$f : \mathbb{Z}^t \to E_0(\mathbb{F}_{p^2}),$$

$$(a_1, \ldots, a_t) \mapsto \sum_{i=1}^{t} [a_i]X_i.$$

Shor's algorithm [83] can be used with this input to solve for a non-zero element of $\ker(f)$, say some $(b_1, \ldots, b_t)$. Similarly, since discrete logarithms are easy in the groups $\mathbb{Z}/2^m\mathbb{Z}$ and $\mathbb{Z}/3^n\mathbb{Z}$ using the methods of Pohlig-Hellman [79], a classical approach to finding such a solution $(b_1, \ldots, b_t)$ exists as well. Then,

$$\sum_{i=1}^{t} [b_i](\phi(X_i) + Y_i) = \sum_{i=1}^{t} [b_i]Y_i \in \ker \psi.$$

This random element of $\ker \psi$ can be computed directly from the public key, and therefore generates the entirety of $\ker \psi \subset E_1[3^n]$ with probability 2/3. Hence, this encryption scheme is not KP when $t > 4$. It is important to note here that, in the second case ($t > 4$), the attack works only because the elliptic curve groups are Abelian.

References

[1] Gora Adj, Daniel Cervantes-Vázquez, Jesús-Javier Chi-Domínguez, Alfred Menezes, and Francisco Rodríguez-Henríquez. On the cost of computing isogenies between supersingular elliptic curves. Cryptology ePrint Archive, Report 2018/313, 2018. https://eprint.iacr.org/2018/313.

[2] Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange—A New Hope. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 327–343, Austin, TX, 2016. USENIX Association.

[3] Sarah Arpin, Catalina Camacho-Navarro, Kristin E. Lauter, Joelle Lim, Kristina Nelson, Travis Scholl, and Jana Sotáková. Adventures in supersingularland. *arXiv preprint arXiv:1909.07779*, 2019.

[4] Reza Azarderakhsh, Amir Jalali, David Jao, and Vladimir Soukharev. Practical supersingular isogeny group key agreement. *IACR Cryptology ePrint Archive*, 2019:330, 2019.

[5] Reza Azarderakhsh, David Jao, Kassem Kalach, Brian Koziel, and Christopher Leonardi. Key compression for isogeny-based cryptosystems. In *Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography*, AsiaPKC '16, pages 1–10. ACM, 2016.

[6] Reza Azarderakhsh, David Jao, and Christopher Leonardi. Post-quantum static-static key agreement using multiple protocol instances. In *International Conference on Selected Areas in Cryptography*, pages 45–63. Springer, 2017.

[7] Efrat Bank, Catalina Camacho-Navarro, Kirsten Eisentraeger, Travis Morrison, and Jennifer Park. Cycles in the supersingular ℓ-isogeny graph and corresponding endomorphisms, 2018.

[8] Andrea Basso, Péter Kutas, Simon-Philipp Merz, Christophe Petit, and Charlotte Weitkämper. On adaptive attacks against Jao-Urbanik's isogeny-based protocol. Cryptology ePrint Archive, Report 2020/244, 2020. https://eprint.iacr.org/2020/244.

[9] Daniel J. Bernstein, Luca De Feo, Antonin Leroux, and Benjamin Smith. Faster computation of isogenies of large prime degree. Cryptology ePrint Archive, Report 2020/341, 2020. https://eprint.iacr.org/2020/341.

[10] Daniel J. Bernstein, Nadia Heninger, Paul Lou, and Luke Valenta. Post-quantum RSA. Cryptology ePrint Archive, Report 2017/351, 2017. http://eprint.iacr.org/2017/351.

[11] Jean-François Biasse, Annamaria Iezzi, and Michael J. Jacobson. A note on the security of CSIDH. In *International Conference on Cryptology in India*, pages 153–168. Springer, 2018.

[12] Jean-François Biasse, David Jao, and Anirudh Sankar. A quantum algorithm for computing isogenies between supersingular elliptic curves. In *Progress in Cryptology – INDOCRYPT 2014*, volume 8885 of *Lecture Notes in Computer Science*, pages 428–442. Springer International Publishing, 2014.

[13] David Biron, Ofer Biham, Eli Biham, Markus Grassl, and Daniel A. Lidar. Generalized Grover search algorithm for arbitrary initial amplitude distribution. In *1st NASA Conference on Quantum Computing and Quantum Communications*, 1998.

[14] Xavier Bonnetain and André Schrottenloher. Quantum security analysis of CSIDH. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 493–522. Springer, 2020.

[15] Joppe Bos, Craig Costello, Léo Ducas, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Ananth Raghunathan, and Douglas Stebila. Frodo: Take off the ring! practical, quantum-secure key exchange from LWE. In *Proc. 23rd ACM Conference on Computer and Communications Security (CCS) 2016*. ACM, October 2016.

[16] Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, and Damien Stehlé. CRYSTALS – Kyber: a CCA-secure module-lattice-based KEM. Cryptology ePrint Archive, Report 2017/634, 2017.

[17] A. Bostan, François Morain, Bruno Salvy, and Éric Schost. Fast algorithms for computing isogenies between elliptic curves. *Mathematics of Computation*, 77(263):1755–1778, 2008.

[18] Paul Bottinelli, Victoria de Quehen, Chris Leonardi, Anton Mosunov, Filip Pawlega, and Milap Sheth. The Dark SIDH of Isogenies. Cryptology ePrint Archive, Report 2019/1333, 2019. https://eprint.iacr.org/2019/1333.

[19] Reinier Bröker. Constructing supersingular elliptic curves. *J. Comb. Number Theory*, 1:269–273, 2009.

[20] Reinier Bröker, Juliana Belding, Andreas Enge, and Kristin E. Lauter. Computing Hilbert class polynomials. *Algorithmic Number Theory Symposium*, VIII:282–295, 2008.

[21] Reinier Bröker, Dennis Charles, and Kristin E. Lauter. Evaluating large degree isogenies and applications to pairing based cryptography. In *Pairing-Based Cryptography - Pairing 2008*, volume 5209 of *LNCS*, pages 100–112, 2008.

[22] Ran Canetti and Hugo Krawczyk. Analysis of key-exchange protocols and their use for building secure channels. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 453–474. Springer, 2001.

[23] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes. CSIDH: an efficient post-quantum commutative group action. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 395–427. Springer, 2018.

[24] Wouter Castryck, Lorenz Panny, and Frederik Vercauteren. Rational isogenies from irrational endomorphisms. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 523–548. Springer, 2020.

[25] Juan Marcos Cerviño. On the correspondence between supersingular elliptic curves and maximal quaternionic orders. *Mathematisches Institut Georg-August-Universität Göttingen Seminars Summer Term 2004 (Universitätsverlag Göttingen)*, pages 53–60, 2004.

[26] Denis X. Charles, Kristin E. Lauter, and Eyal Z. Goren. Cryptographic hash functions from expander graphs. *Journal of CRYPTOLOGY*, 22(1):93–113, 2009.

[27] Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, and Daniel Smith-Tone. Report on post-quantum cryptography. *NIST IR 8105*, February 2016.

[28] Andrew M. Childs, David Jao, and Vladimir Soukharev. Constructing elliptic curve isogenies in quantum subexponential time, 2010. *Preprint available at http://arxiv.org/abs/1012.4019.*

[29] Andrew M. Childs, David Jao, and Vladimir Soukharev. Constructing elliptic curve isogenies in quantum subexponential time. *J. Math. Cryptol*, 8:1–29, 2014.

[30] Henri Cohen. *A Course in Computational Algebraic Number Theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, 1993.

[31] Henri Cohen and Gerhard Frey, editors. *Handbook of Elliptic and Hyperelliptic Curve Cryptography*. Chapman and Hall/CRC, 2005.

[32] Ian Connell. Elliptic curve handbook. http://www.math.mcgill.ca/connell/public/, 1999.

[33] Craig Costello, David Jao, Patrick Longa, Michael Naehrig, Joost Renes, and David Urbanik. Efficient compression of SIDH public keys. Cryptology ePrint Archive, Report 2016/963, 2016.

[34] Craig Costello, Patrick Longa, and Michael Naehrig. *Efficient Algorithms for Supersingular Isogeny Diffie-Hellman*, pages 572–601. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[35] Craig Costello, Patrick Longa, Michael Naehrig, Joost Renes, and Fernando Virdia. Improved classical cryptanalysis of SIKE in practice. Cryptology ePrint Archive, Report 2019/298, 2019. https://eprint.iacr.org/2019/298.

[36] Jean-Marc Couveignes. Hard homogenous spaces. http://eprint.iacr.org/2006/291/, 2006.

[37] David A. Cox. Primes of the form $x^2 + ny^2$. a wiley-interscience publication, 1989.

[38] Hassan Daghigh, Ruholla Khodakaramian Gilan, and Fatemeh Seifi Shahpar. Diffie-Hellman type key exchange protocols based on isogenies. *Bulletin of the Iranian Mathematical Society*, 43(4):77–88, August 2017.

[39] Luca De Feo, Cyril Hugounenq, Jérôme Plût, and Éric Schost. Explicit isogenies in quadratic time in any characteristic. *LMS Journal of Computation and Mathematics*, 19(A):267–282, 2016.

[40] Luca De Feo, Simon Masson, Christophe Petit, and Antonio Sanso. Verifiable delay functions from supersingular isogenies and pairings. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 248–277. Springer, 2019.

[41] Christina Delfs and Steven D. Galbraith. Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. In *Designs, Codes and Cryptography*, volume 78, pages 425–440, 2016.

[42] Max Deuring. Die typen der multiplikatorenringe elliptischer funktionenkörper. *Abh. Math. Sem.*, 14:197–272, 1941.

[43] Jintai Ding, Xiang Xie, and Xiaodong Lin. A simple provably secure key exchange scheme based on the learning with errors problem. Cryptology ePrint Archive, Report 2012/688, 2012. http://eprint.iacr.org/2012/688.

[44] Samuel Dobson, Steven D. Galbraith, Jason LeGrow, Yan Bo Ti, and Lukas Zobernig. An adaptive attack on 2-SIDH. Cryptology ePrint Archive, Report 2019/890, 2019. https://eprint.iacr.org/2019/890.

[45] Kirsten Eisenträger, Sean Hallgren, Kristin E. Lauter, Travis Morrison, and Christophe Petit. Supersingular isogeny graphs and endomorphism rings: reductions and solutions. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 329–368. Springer, 2018.

[46] Kirsten Eisenträger, Sean Hallgren, Chris Leonardi, Travis Morrison, and Jennifer Park. Computing endomorphism rings of supersingular elliptic curves and connections to pathfinding in isogeny graphs. *arXiv preprint arXiv:2004.11495. To Appear in ANTS 2020*, 2020.

[47] 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.

[48] Scott Fluhrer. Cryptanalysis of ring-LWE based key exchange with key share reuse. Cryptology ePrint Archive, Report 2016/085, 2016. http://eprint.iacr.org/2016/085.

[49] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In Michael J. Wiener, editor, *Advances in Cryptology – CRYPTO '99: 19th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 1999, Proceedings*, pages 537–554, 1999.

[50] Satoshi Furukawa, Noboru Kunihiro, and Katsuyuki Takashima. Multi-party key exchange protocols from supersingular isogenies. In *2018 International Symposium on Information Theory and Its Applications (ISITA)*, pages 208–212, 2018.

[51] Steven D. Galbraith. *Mathematics of Public Key Cryptography.* Cambridge University Press, 2012.

[52] Steven D. Galbraith, Florian Hess, and Nigel P. Smart. Extending the GHS Weil descent attack. In *Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 29–44, 2002.

[53] Steven D. Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. Cryptology ePrint Archive, Report 2016/859, 2016. http://eprint.iacr.org/2016/859.

[54] Steven D. Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. In *ASIACRYPT (1)*, pages 3–33. Springer, 2017.

[55] Steven D. Galbraith and Anton Stolbunov. Improved algorithm for the isogeny problem for ordinary elliptic curves. *Applicable Algebra in Engineering, Communication and Computing*, 24(2):107–131, 2013.

[56] Lov K. Grover. A fast quantum mechanical algorithm for database search. *In Proc. ACM STOC*, 1996.

[57] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In *Theory of Cryptography Conference*, pages 341–371. Springer, 2017.

[58] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *International Workshop on Post-Quantum Cryptography*, pages 19–34. Springer, 2011.

[59] David Jao, Jason LeGrow, Christopher Leonardi, and Luis Ruiz-Lopez. A subexponential-time, polynomial quantum space algorithm for inverting the CM group action. *Journal of Mathematical Cryptology*, 2018.

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

[61] 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.

[62] Kiran S. Kedlaya and Christopher Umans. Fast polynomial factorization and modular composition. *SIAM J. Comput.*, 40:1767–1802, 2008.

[63] Daniel Kirkwood, Bradley C. Lackey, John McVey, Mark Motley, Jerome A. Solinas, and David Tuller. Failure is not an option: Standardization issues for post-quantum key agreement. *Workshop on Cybersecurity in a Post-Quantum World*, 2015.

[64] David Kohel. *Endomorphism rings of elliptic curves over finite fields*. PhD thesis, University of California at Berkeley, 1996.

[65] David Kohel, Kristin E. Lauter, Christophe Petit, and Jean-Pierre Tignol. On the quaternion ℓ-isogeny path problem. *LMS Journal of Computation and Mathematics*, 17(A):418–432, 2014.

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

[67] Serge Lang. *Elliptic Functions*, volume 112 of *Graduate Texts in Mathematics*. Springer, 1987.

[68] Christopher Leonardi and Luis Ruiz-Lopez. Homomorphism learning problems and its applications to public-key cryptography. *CFAIL 2019*, 2019.

[69] Reynald Lercier and François Morain. Counting the number of points on elliptic curves over finite fields: strategies and performances. In *Advances in Cryptology - EUROCRYPT '95*, volume 921 of *LNCS*, pages 79–94. Springer, 1995.

[70] John E. Littlewood. On the class-number of the corpus $\mathrm{P}(\sqrt{-k})$. *Proceedings of the London Mathematical Society*, s2-27(1):358–372, 1928.

[71] Ken McMurdy. Explicit representation of the endomorphism rings of supersingular elliptic curves, 2014.

[72] James S. Milne. Modular functions and modular forms. *University of Michigan lecture notes*, 1997.

[73] Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. MDPC—McEliece: New McEliece Variants from Moderate Density Parity-Check Codes. In *IEEE International Symposium on Information Theory - ISIT 2013*, pages 2069–2073, 2013.

[74] Chris Peikert. *Lattice Cryptography for the Internet*, pages 197–219. Springer International Publishing, 2014.

[75] Chris Peikert. He gives C-sieves on the CSIDH. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 463–492. Springer, 2020.

[76] Christophe Petit. Faster algorithms for isogeny problems using torsion point images. In *ASIACRYPT*, 2017.

[77] Christophe Petit and Kristin E. Lauter. Hard and easy problems for supersingular isogeny graphs. *IACR Cryptology ePrint Archive*, 2017:962, 2017.

[78] Arnold Pizer. An algorithm for computing modular forms on $\gamma_0(n)$. 1980.

[79] Stephen Pohlig and Martin Hellman. An improved algorithm for computing logarithms over GF(p) and its cryptographic significance. In *IEEE Transactions on Information Theory*, volume 24, pages 106–110, 1978.

[80] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009.

[81] J. Barkley Rosser and Lowell Schoenfeld. Approximate formulas for some functions of prime numbers. *Illinois J. Math.*, 6(1):64–94, 03 1962.

[82] Alexander Rostovtsev and Anton Stolbunov. Public-key cryptosystem based on isogenies. *IACR Cryptol. ePrint Arch.*, 2006:145, 2006.

[83] Peter Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26:1484–1509, 1997.

[84] Igor E. Shparlinski and Andrew V. Sutherland. On the distribution of Atkin and Elkies primes for reductions of elliptic curves on average. *LMS Journal of Computation and Mathematics*, 18(1):308–322, 2015.

[85] Joseph H. Silverman. *The Arithmetic of Elliptic Curves*. Springer, 1986.

[86] Joseph H. Silverman. *Advanced topics in the arithmetic of elliptic curves*. Number 151 in Lecture Notes in Mathematics. Springer, 1994.

[87] Joseph H. Silverman. *The Arithmetic of Elliptic Curves*, volume 106 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2 edition, 2009.

[88] Benjamin Smith. Mappings of elliptic curves. Webpage, 2008.

[89] Anton Stolbunov. Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves. *NISK*, pages 97–109, 2009.

[90] Andrew V. Sutherland. On the evaluation of modular polynomials. *The Open Book Series*, 1(1):531–555, Nov 2013.

[91] Seiichiro Tani. Claw finding algorithms using quantum walk. *Theoretical Computer Science*, 410:5285–5297, 2009.

[92] John Tate. Endomorphisms of Abelian varieties over finite fields. *Inventiones mathemticae*, 2(2):134–144, 1966.

[93] David Urbanik and David Jao. SoK: The problem landscape of SIDH. In *Proceedings of the 5th ACM on ASIA Public-Key Cryptography Workshop*, APKC '18, pages 53–60, New York, NY, USA, 2018. ACM.

[94] David Urbanik and David Jao. New techniques for SIDH-based NIKE. *Journal of Mathematical Cryptology*, 14(1):120 – 128, 2020.

[95] Jacques Vélu. Isogénies entre courbes elliptiques. *CR Acad. Sc. Paris.*, 273:238–241, 1971.

[96] John Voight. Quaternion algebras. *Version v0*, 9, 2018.

[97] Ingo von Maurich, Lukas Heberle, and Tim Güneysu. *IND-CCA Secure Hybrid Encryption from QC-MDPC Niederreiter*, pages 1–17. Springer International Publishing, 2016.

[98] Lawrence C. Washington. *Elliptic Curves: Number Theory and Cryptography*. Chapman and Hall/CRC, 2008.

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

[100] Gustavo Zanon, Marcos A. Simplício Jr., Geovandro C. C. F. Pereira, Javad Doliskani, and Paulo S. L. M. Barreto. Faster isogeny-based compressed key agreement. In Tanja Lange and Rainer Steinwandt, editors, *Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018, Fort Lauderdale, FL, USA, April 9-11, 2018, Proceedings*, volume 10786 of *Lecture Notes in Computer Science*, pages 248–268. Springer, 2018.

Appendices

Appendix A

Relating Fixed Kernels to the Endomorphism Algebra

We present here a remark on the relationship between Proposition 3.2.1 and Lemma 42.2.9 of [96]. The lemma of Voight is as follows. Let $\mathbb{B}_{p,\infty}$ be the quaternion algebra ramified at exactly p and ∞. Let $E(\overline{\mathbb{F}}_p)$ be a supersingular elliptic curve with $\operatorname{End}(E) \cong \mathcal{O} \subset \mathbb{B}_{p,\infty}$. Let $I \subset \mathcal{O}$ be a non-zero integral left $\mathcal{O}$-ideal. Define

$$E[I] := \bigcap_{\alpha \in I} \ker \alpha,$$

and denote by ϕ_I the isogeny $\phi_I : E \to E_I$, where $E_I = E/E[I]$.

Lemma A.0.1. *The ring homomorphism*

$$\iota : \operatorname{End}(E_I) \hookrightarrow \mathbb{B}_{p,\infty},$$
$$\iota(\beta) = \phi_I^{-1} \beta \phi_I = \frac{1}{\deg \phi_I} \hat{\phi}_I \beta \phi_I$$

is injective and $\iota(\operatorname{End}(E_I)) = O_R(I)$, the right order of I.

The proof of Lemma A.0.1 can be found in [96] on page 772, as the proof of Lemma 42.2.9. To relate Lemma A.0.1 to Proposition 3.2.1 we first need the following lemma.

Lemma A.0.2. *Let the setup be as in Proposition 3.2.1. The endomorphism*

$$\iota := \phi \circ \psi \circ \hat{\phi} \in \operatorname{End}(E')$$

annihilates the subgroup $E'[N]$.

Proof. Let $P \in E'[N]$ generate the $\ker \hat{\phi}$, and let $Q \in E'[N]$ be linearly independent of P and of full order. It is clear that $\iota(P) = \infty_{E'}$, so we must show that ι also annihilates Q.

Since $\iota(E'[N]) = \ker \phi$, and $Q \notin \ker \hat{\phi}$, we have that $\hat{\phi}(Q) \in \ker \phi$ and is of full order. But ψ fixes $\ker \phi$, so $\psi \circ \hat{\phi}(Q) \in \ker \phi$. □

Lemma 42.2.9 of [96] states that, given an element $\beta \in \mathrm{End}(E_I)$ and an isogeny $\phi_I : E_0 \to E_I$, the quantity

$$\phi_I^{-1} \beta \phi_I = \frac{1}{\deg \phi_I} \left(\hat{\phi}_I \beta \phi_I \right)$$

corresponds to an element of the quaternion algebra, isomorphic to $\mathrm{End}(E_0) \otimes \mathbb{Q}$, but not necessarily to an element of $\mathrm{End}(E_0)$. Proposition 3.2.1 and Lemma A.0.2 state that if β fixes the kernel subgroup of ϕ_I and these two isogenies have coprime degrees, then the above composition is in fact an element of $\mathrm{End}(E_0)$.

Appendix B

Convenient Bases

We examine the construction of a basis P, Q of a supersingular $E(\mathbb{F}_{p^2})$ with $j(E) = 1728$ such that $Q = \iota(P) = \pi(P)$, where ι is the distortion map

$$\iota : E \to E,\ (x, y) \mapsto (-x, i \cdot y),$$

for some $i \in \overline{\mathbb{F}}_p \backslash \mathbb{F}_p$ with $i^2 = -1$, and π is the Frobenius endomorphism

$$\pi : E \to E,\ (x, y) \mapsto (x^p, y^p).$$

Lemma B.0.1. *Let P be a point on $E(\mathbb{F}_{p^2})$. The following are equivalent*

1. *$\pi(P) = \iota(P)$,*
2. *$P = (ui, v(1 - i))$ for some $u, v \in \mathbb{F}_p$, and*
3. *$u - u^3 = -2v^2$ for $u, v \in \mathbb{F}_p$.*

Proof. We start by proving the first and second items are equivalent. Let $P = (r + ui, v + si) \in E(\mathbb{F}_{p^2})$, for $r, u, v, s \in \mathbb{F}_p$, and assume $\pi(P) = \iota(P)$. Then $\pi(P) = (r - ui, v - si)$ and $\iota(P) = (-r - ui, -s + vi)$. Hence, $r = 0$ and $s = -v$, and we can write $P = (ui, v(1 - i))$.

Next, suppose $P = (ui, v(1 - i))$ for some $u, v \in \mathbb{F}_p$. Then

$$\pi(P) = (-ui, v(1 + i)) = (-(ui), (i)(v(1 - i))) = \iota(P).$$

Now we prove the second item is equivalent to the third. Suppose $P = (ui, v(1 - i))$ for some $u, v \in \mathbb{F}_p$. Then

$$(ui)^3 + (ui) = (v(1 - i))^2$$

implying that

$$-2v^2 = u - u^3.$$

Lastly, let $u, v \in \mathbb{F}_p$ satisfy $u - u^3 = -2v^2$. Then

$$(ui)^3 + (ui) = -i(u^3 - u) = -i(-2v^2) = 2iv^2 = (v(1-i))^2,$$

so there is a point $(ui, v(1-i))$ on $E(\mathbb{F}_{p^2})$. □

Example B.0.2. *We give an example of a basis of the form $\{P, \iota P\} = \{P, \pi(P)\}$ for the Round 1 SIKE subission's power of 3 torsion subgroup. Consider the prime $p = 2^{372}3^{239} - 1$ and elliptic curve $E : y^2 = x^3 + x$. Let*

$$\begin{aligned}P = (&570556479520931242046188854123607496106560164061907466550384438849978105\\ &317427915430702852898083512987673860531431396211577428396652625840972800081\\ &121950020629423752030609821533127797086914328692076275754730669931633440054\\ &111 \cdot i, 170529679711570871948934177887081253029965384490684199363468409357092\\ &795616015244936108185015599117514645468695608544218109856248721923973057187\\ &123901749477184914154433309844493495718071811551302575552223407602899906055\\ &523667 + 101841880620577343810288340599797240683974242610583869707527212696 9\\ &758614506646360156677471154546759610990774292323165816709405572780063041998\\ &697334486940233979179268564560348986490401288096337567765953462758389897275\\ &9596053164 \cdot i).\end{aligned}$$

Then P has order 3^{239}, satisfies the condition of Lemma B.0.1 (so $\iota(P) = \pi(P)$), and $\{P, \pi(P)\}$ is a basis for $E[3^{239}]$. Thus, by the discussion in Section 3.5, with respect to this basis for $E[3^{239}]$, endomorphisms of the form

$$\psi = [w] \cdot [1] + [x] \cdot \iota + [y] \cdot \tfrac{[1]+\pi}{2} + [z] \cdot \tfrac{\iota + \iota\pi}{2}.$$

act as the matrix $\begin{bmatrix} w + \frac{y-z}{2} & -x + \frac{y-z}{2} \\ x + \frac{y+z}{2} & w + \frac{y+z}{2} \end{bmatrix}$ on $E[3^{239}]$.