The Dark SIDH of Isogenies

Paul Bottinelli, Victoria de Quehen, Chris Leonardi, Anton Mosunov, Filip Pawlega, and Milap Sheth

ISARA Corporation, Waterloo, Canada
{paul.bottinelli,victoria.dequehen,chris.leonardi,filip.pawlega,milap.sheth}@isara.com
amosunov@uwaterloo.ca

Abstract. Many isogeny-based cryptosystems are believed to rely on the hardness of the Supersingular Decision Diffie-Hellman (SSDDH) problem. However, most cryptanalytic efforts have treated the hardness of this problem as being equivalent to the more generic supersingular ℓ^e-isogeny problem — an established hard problem in number theory.

In this work, we shine some light on the possibility that the combination of two additional pieces of information given in practical SSDDH instances — the image of the torsion subgroup, and the starting curve's endomorphism ring — can lead to better attacks cryptosystems relying on this assumption. We show that SIKE/SIDH are secure against our techniques. However, in certain settings, e.g., multi-party protocols, our results may suggest a larger gap between the security of these cryptosystems and the ℓ^e-isogeny problem.

Our analysis relies on the ability to find many endomorphisms on the base curve that have special properties. To the best of our knowledge, this class of endomorphisms has never been studied in the literature. We informally discuss the parameter sets where these endomorphisms should exist. We also present an algorithm which may provide information about additional torsion points under the party's private isogeny, which is of independent interest. Finally, we present a minor variation of the SIKE protocol that avoids exposing a known endomorphism ring.

Table of Contents

1 Introduction . . . 3
1.1 Our Contributions and Organization of the Paper . . . 4
2 Preliminaries . . . 5
2.1 Notation . . . 5
2.2 Hard Isogeny Problems . . . 6
3 Technical Preview . . . 8
4 Interpreting Private Keys as Eigenvectors . . . 10
4.1 The GPST Active Attack on SIDH . . . 11
4.2 GPST-Inspired Cryptanalysis . . . 13
5 Exploiting Endomorphisms . . . 16
5.1 Triangular Kernels . . . 16
5.2 Main Theorem . . . 21
6 Quadratic Forms and Endomorphism Rings . . . 23
6.1 Quadratic Forms from Degrees of Endomorphisms . . . 24
6.2 Quadratic Forms from Eigenspaces of Endomorphisms . . . 27
7 Instantiating the Oracle for $j = 1728$ . . . 31
7.1 The Quadratic Form for $j = 1728$ . . . 31
7.2 The Main Reduction for $j = 1728$ . . . 33
7.3 Characterizing Large Eigenspaces for $j = 1728$ . . . 35
8 Alternate Settings . . . 39
8.1 3-Party Setting . . . 39
8.2 4-Party Setting . . . 42
8.3 Unbalanced Setting . . . 43
8.4 Summary of our Results . . . 44
9 Using Endomorphisms with Almost-Eigenvectors . . . 45
9.1 Almost-Invariant Kernels . . . 45
9.2 Almost-Eigenvectors . . . 49
10 Learning Secret Torsion Information . . . 51
11 Recommendations . . . 56
11.1 Supersingular Isogeny Two-party Handshake (SITH) . . . 57
12 Conclusion and Future Work . . . 58
12.1 Future Work . . . 59
A Appendix . . . 61
A.1 SIKE . . . 61
A.2 Proofs . . . 62
A.3 Convenient Basis . . . 65
A.4 Solutions to Quadratic Equations . . . 66

1 Introduction

By the early 2000s, the elliptic curve discrete logarithm problem had become the primary choice for concrete instantiations of fundamental cryptographic protocols, and enabled many new advancements in the field. Looking closer at the structure of, and relationship between elliptic curve groups gave way to a natural generalization: rather than using scalar multiplication maps, cyclic isogenies could be used to achieve similar algebraic properties [9, 20, 22]. In 2006, Charles, Goren and Lauter [5] introduced the first cryptographic primitive (a hash function) relying on the hardness of finding isogenies between *supersingular* elliptic curves. Their work introduced the "supersingular ℓ^e-isogeny problem" in a cryptographic context, and provided heuristic analysis of its hardness by studying the structure of the set of all possible isogenies between supersingular elliptic curves.

In 2011, supersingular isogeny-based cryptography received renewed attention with the demonstration of a quantum sub-exponential algorithm for finding isogenies between ordinary elliptic curves by Childs, Jao, and Soukharev [7]. Motivated by avoiding this attack, De Feo and Jao [13] revisited the ℓ-isogeny graph of supersingular curves first considered in [5], and introduced the Supersingular Isogeny Diffie-Hellman protocol (SIDH).

In order to overcome a technical obstacle in constructing the protocol, the public keys in SIDH include the images of certain torsion points under the (private) isogenies. The Supersingular Isogeny Decisional Diffie-Hellman assumption (SSDDH) – and its computational versions Supersingular Isogeny Computational Diffie-Hellman assumption (SSCDH) and Computational Supersingular Isogeny assumption (CSSI) – were introduced to prove the security of this protocol given the additional information. The Supersingular Key Encapsulation Mechanism SIKE [12], currently being considered for standardization in the NIST Post-Quantum Standardization Process [6], also relies on the hardness of the relatively-new SSDDH problem *on a fixed starting curve*. This new computational assumption was presumed to be equivalent to the supersingular ℓ^e-isogeny problem.

The distinction between the SSDDH assumption and the ℓ^e-isogeny problem (namely, that torsion point images of private kernels are revealed) was first exploited in 2016, when the prominent work due to Galbraith, Petit, Silva, and Ti [11] (later referred to as "GPST") presented an active attack on the use of static-keys in SIDH.

As mentioned, SIKE actually relies on a potentially stronger assumption; SSDDH instantiated using a particular *starting* elliptic curve whose endomorphism ring is well-known. In 2017, the authors in [10] showed heuristically that solving the supersingular ℓ^e-isogeny problem is equivalent to the problem of constructing the endomorphism rings of *both* the starting and ending supersingular elliptic curves. Since the endomorphism rings of the target elliptic curves are unknown, and there are no known efficient algorithms which find the endomorphism rings of arbitrary supersingular elliptic curves, their approach does not reduce the

security of SIKE. However, their work demonstrated that the problem of finding endomorphism rings is intricately linked to the security of SIKE.

Since then, numerous works have shown constructions of additional primitives assuming the hardness of classical ℓ^e-isogeny problems, SSDDH, and strengthened variants of SSDDH. Most of the current best-known attacks [1,24] on SIDH/SIKE find direct solutions to the ℓ^e-isogeny problem. Recently, the authors in [14] have also suggested that the quantum attack of Biasse et al. [4] exploiting the algebraic structure of supersingular elliptic curves defined over $\mathbb{F}_p$ might be better than the often cited generic quantum claw-finding attack [24].

In 2017, Petit [17] described a passive polynomial time algorithm for solving the CSSI problem on two non-SIKE parameter sets, the first work to utilize both the image of the torsion points, and the knowledge of the endomorphism ring of the starting curve. In the first variant, one party reveals drastically more torsion information than in SIDH/SIKE (like in multi-party settings [3]), and in the other variant both parties work in torsion subgroups larger than p^2 (where p is the characteristic of the field) and reveal slightly more torsion information than in SIDH/SIKE. The novelty of the work we present in this paper is that we arrive at potentially stronger results in the same vein of that work, but using different methods. More specifically, we aim to answer the following question: is there an offline algorithm which solves CSSI by repeatedly restricting the search-space by a non-negligible factor, e.g., a *passive* counterpart to [11]?

1.1 Our Contributions and Organization of the Paper

This work constitutes an independent line of research into non-generic attacks on the instantiations of the CSSI problem. We describe a reduction between the security of SIDH-like protocols and the CSSI problem on certain starting elliptic curves. Given an oracle for this new problem, we present a *passive* algorithm which iteratively shrinks the search space for solutions to the CSSI problem, which corresponds to recovering a private key (isogeny) in protocols relying on the CSSI assumption. Our approach exploits the knowledge of the images of the private isogeny on a torsion subgroup, and the structure of the endomorphism ring of the starting curve.

In Section 2 we review the hardness assumptions used to establish the security of SIKE, and in Section 3 we give a technical overview of the work in this document. Similar to our reduction, the "GPST attack" [11] iteratively gives malformed public keys to halve the search space. Our methods can be interpreted as a generalization of the underlying ideas implicit in the GPST attack (see Table 1 for a comparison). While the best-known attacks deal with elliptic curves and other rich structures, our analysis interprets torsion points as *vectors* and endomorphisms as matrices, and applies techniques from linear algebra. We formulate the GPST attack using this terminology in Section 4.

Section 5 describes our main observation: a passive algorithm which, when given *desirable* endomorphisms, solves CSSI. We can increase the efficiency of our algorithm by assuming that these desirable endomorphisms are represented

in a novel way, called a *triangular kernel*. Thus, we show a reduction between the CSSI problem and the problem of finding desirable endomorphisms.

Once we fix a specific (starting) elliptic curve, constructing such endomorphisms essentially reduces to solving a particular quadratic form. In Section 6, we make explicit the reduction between the security of SIDH/SIKE variants and the problem of finding (desirable) solutions to this quadratic form. An important result of our investigations, which we show in Section 7, is that SIKE is secure against our cryptanalysis.

Section 8 explores whether these desirable endomorphisms exist in the multi-party setting (e.g., group key agreements [3]). We provide arguments that suggest the security of the standard 3-party and 4-party cases may be significantly less than currently believed. Specifically, we provide heuristic evidence about the existence of desirable endomorphisms for our attacks in the 4-party setting.

Section 9 examines the tradeoff between the runtime of our algorithm and the amount of information each desirable endomorphism provides. Section 10 focuses on improving the algorithm so that it requires fewer desirable endomorphisms. Informally, we achieve this by learning the images of the private isogeny on additional torsion points.

Finally, in Section 11, we introduce a modified version of SIKE (called SITH) which relies on a qualitatively weaker security hypothesis. We end with our conclusions and future work in Section 12. This work shares a similar conclusion to that of Petit [17], which presented a passive polynomial-time algorithm to solve two variants of the CSSI problem.

2 Preliminaries

As we will be working in a broader setting than SIKE, Section 2.1 introduces the necessary notation for general isogeny-based key exchanges, and Section 2.2 reviews the hard problems upon which the security of these key exchanges is based.

2.1 Notation

Throughout, we will assume a generalization of the SIKE setup. In particular, let $\mathbb{F}_q$ denote a finite field with q elements, where $q = p^2$ for some prime p of the form $p = N_1 N_2 - 1$, for coprime positive integers N_1 and N_2. The security parameter of isogeny-based cryptosystems is $\lambda = \log p$. In much of Sections 7 and 8 we will be assuming $p \equiv 3 \pmod 4$. In this case, as $\mathbb{F}_{p^2} \cong \mathbb{F}_p[x]/\langle x^2 + 1\rangle$, we can represent elements in $\mathbb{F}_{p^2}$ as $u + vi$, where $i^2 = -1$, for some $u, v \in \mathbb{F}_p$.

We will let $E(\mathbb{F})$ denote a supersingular elliptic curve E over a field $\mathbb{F}$. As well, let $E[N]$ denote the subgroup of N-torsion points over the algebraic closure $\overline{\mathbb{F}}_q$, and let $[N]$ denote the isogeny that acts as scalar multiplication by N.

For any $N \in \mathbb{Z}$, where $p \nmid N$, the subgroup $E[N] \cong \mathbb{Z}/N\mathbb{Z} \times \mathbb{Z}/N\mathbb{Z}$. We fix bases for the N_1 and N_2 torsion subgroups: $E[N_1] = \langle P_A, Q_A\rangle$ and $E[N_2] = \langle P_B, Q_B\rangle$.

Let $R_A = P_A + [r_A] \cdot Q_A$ for some number $r_A \in \{0, \ldots, N_A - 1\}$, and let $\phi_A : E \to E_A$ be an isogeny with $\ker(\phi_A) = \langle R_A \rangle$. Alice will be the static initiator in the key exchange, whose *private key* is r_A. We refer to R_A as her *private point*, which generates the kernel of her *private isogeny* ϕ_A. Her *public key* is $(\phi_A(E), \phi_A(P_B), \phi_A(Q_B))$.

Occasionally, we will also use an isogeny $\phi_B : E \to E_B$, where $\ker(\phi_B) = \langle P_B + [r_B] \cdot Q_B \rangle$ for some number $r_B \in \{0, \ldots, N_B - 1\}$. Bob will be the responder in the key exchange, whose *private key* is r_B. We refer to R_B as his *private point*, which generates the kernel of his *private isogeny* ϕ_B. His *public key* is $(\phi_B(E), \phi_B(P_A), \phi_B(Q_A))$.

Definition 2.1. Let $\phi : E \to E'$ be an isogeny. Then ϕ is said to be *cyclic* when there is no integer $m \neq \{\pm 1\}$ and isogeny ψ such that $\phi = [m] \cdot \psi$.

We denote the action of the isogeny ϕ restricted to the N-torsion points by $\phi|_{E[N]}$, typically with respect to a given basis $\{P, Q\}$ for $E[N]$. Finally, $\widehat{\phi}$ will denote the dual of an isogeny ϕ.

Let $\mathrm{End}(E)$ denote the endomorphism ring of E. Since E is supersingular, we can write $\mathrm{End}(E)$ as a $\mathbb{Z}$-module generated by some basis of endomorphisms $\{b_1, b_2, b_3, b_4\}$.

For a natural number N and linear transformation M on $E[N]$, let

$$\mathsf{Eig}_N M = \{R \in E[N] : |R| = N, \langle M(R) \rangle \subset \langle R \rangle\}.$$

We refer to this as the N-*eigenspace* of M, and call a torsion point R an *eigenvector* of M if it is in $\mathsf{Eig}_N(M)$. For an endomorphism ϕ_C and natural number N, let

$$\mathsf{Eig}_N(\phi_C) = \{R \in E[N] : |R| = N, \langle \phi_C(R) \rangle \subset \langle R \rangle\}.$$

We refer to this as the N-*eigenspace* of ϕ_C, and call a torsion point R an *eigenvector* of ϕ_C if it is in $\mathsf{Eig}_N(\phi_C)$ for some natural number N. Notice that if $\gcd(\deg \phi_C, N) = 1$ and $R \in \mathsf{Eig}_N(\phi_C)$, then $\langle R \rangle = \langle \phi_C(R) \rangle$.

Let $H(\rho)$ be the *information entropy* of a binary probability event ρ. Then the expected information content of ρ is $H(\rho)$ bits, and can be computed as follows

$$H(\rho) = \rho \log_2(1/\rho) + (1 - \rho) \log_2(1/(1 - \rho)).$$

2.2 Hard Isogeny Problems

As we discussed earlier, most of the current best-known attacks [1, 24] on SIDH/SIKE find direct solutions to the ℓ^e-isogeny problem:

Problem 2.2 (ℓ^e-Isogeny Problem). Suppose there exists an isogeny $\phi : E \to E'$ whose kernel is generated by $R \in E[\ell^e]$. If you are only given E and E', the problem is to find $\langle R \rangle$.

In contrast, the security of SIDH/SIKE is based on the following problem instantiated at a particular elliptic curve:

Problem 2.3 (Supersingular Decision Diffie-Hellman (SSDDH) Problem). 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 and
$$E_{AB} \cong E_0/\langle P_A + [r_A] \cdot Q_A, P_B + [r_B] \cdot Q_B \rangle,$$
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
$$E_C \cong E_0/\langle P_A + [r'_A] \cdot Q_A, P_B + [r'_B] \cdot Q_B \rangle,$$
where r'_A and r'_B are chosen at random from $\mathbb{Z}/N_1\mathbb{Z}$ and $\mathbb{Z}/N_2\mathbb{Z}$, respectively,

determine from which distribution the tuple is sampled.

A computational variant of this problem was introduced by Jao and De Feo in [13].

Problem 2.4 (Supersingular Computational Diffie-Hellman (SSCDH) Problem). Let $\phi_A : E_0 \to E_A$ be an isogeny whose kernel is equal to $\langle P_A + [r_A] \cdot Q_A \rangle$, and let $\phi_B : E_0 \to E_B$ be an isogeny whose kernel is equal to $\langle P_B + [r_B] \cdot Q_B \rangle$, where r_A (respectively r_B) is chosen at random from $\mathbb{Z}/N_1\mathbb{Z}$ (respectively $\mathbb{Z}/N_2\mathbb{Z}$). Given the curves E_A, E_B, and the points $\phi_A(P_B), \phi_A(Q_B), \phi_B(P_A), \phi_B(Q_A)$, find the j-invariant of $E_0/\langle P_A + [r_A] \cdot Q_A, P_B + [r_B] \cdot Q_B \rangle$.

SSDDH reduces to SSCDH in the obvious way, and the converse is true as well [26]. Both the decisional and computational Diffie-Hellman assumptions depend on the following computational problem in the obvious way.

Problem 2.5 (Computational Supersingular Isogeny (CSSI) Problem). Let E_0 be a supersingular elliptic curve, and let $\phi : E_0 \to E_A$ be an isogeny whose kernel is generated by $\langle P_A + [r_A] \cdot Q_A \rangle$ for a random $r_A \in \mathbb{Z}/N_1\mathbb{Z}$, over $\mathbb{F}_{p^2}$ where $p = N_1 N_2 - 1$ is a prime such that $\gcd(N_1, N_2) = 1$. Given E_A and the action of ϕ on $E_0[N_2]$, find r_A.

This work aims to study the security of SIDH/SIKE instances by solving the CSSI Problem, which in turn solves the SSCDH and SSDDH problems.

The security of the NIST Round 1 version of SIKE [6] is based on an instantiation of the SSDDH problem at the elliptic curve $E_0 : y^2 = x^3 + x$ with j-invariant 1728. The endomorphism ring of this elliptic curve is known, and it has many endomorphisms with small norm. It is the knowledge of the core structure of the endomorphism ring of the starting elliptic curve E_0, along with torsion points $\phi_A(P_B)$ and $\phi_A(Q_B)$, that we exploit in the security reductions given in this paper. The elliptic curve in the Round 2 Version of SIKE has a related endomorphism ring because it is adjacent to the SIKE Round 1 curve on the 2-isogeny graph, and can therefore be analyzed in a similar manner.

3 Technical Preview

Constructing public-key cryptosystems like SIDH relying on the hardness of SSDDH can be done by viewing an isogeny as a private key, and an image curve (and torsion points) under that isogeny as the public key. Appendix A.1 presents numerous approaches for passive attacks on SIDH that have been presented in the literature, such as the generic claw-finding algorithms from Tani [24], a quantum algorithm performing unstructured searches through an algebraically constrained space of candidate solutions by Biasse, Jao and Sankar [4], and the work by Petit [17] which showed a relationship between the hardness of multi-party SIDH-like protocols and solving certain quadratic forms.

While we study similar concepts as this last attack, our approach is structurally different, as we draw inspiration from the *active* GPST attack [11], which showed why SIDH is not IND-CCA secure. See Table 1 for a comparison between this work and the GPST attack. For this discussion let us assume that Bob is a dishonest party wishing to discover the other party's, Alice's, private key.

In the standard SIDH protocol, Bob is required to send particular points derived from his private isogeny during the run of the protocol. In the GPST attack (which we describe in more detail in Section 4.1), Bob, acting as an attacker, repeatedly sends a specially crafted linear combination of his points instead.

Bob's correct points form a basis of $E_B[N_1]$, as do his malicious points. We will consider the *change of basis matrix* between these bases. The main idea behind the GPST attack is that, with probability $1/2$, Alice is able to reconstruct the correct isogeny given Bob's malicious points. Whether or not this second isogeny of hers is constructed correctly gives Bob information about Alice's private key, and this information is completely determined by the change of basis matrix that Bob used.

In our methods, the attacker uses an endomorphism on the starting curve instead of a change of basis matrix. More specifically, we show that given a particular endomorphism, the attacker can discover information about Alice's private point.

Contrary to the GPST attack, we provide an reduction which requires access to *desirable* endomorphisms. Hence, for our reduction to result in a useful attack we need i) to obtain a non-negligible amount of information about Alice's private point during each iteration, and ii) that the cost of each iteration only requires reasonably bounded effort. These two goals add different restrictions on the type of endomorphisms that we consider.

We first study the amount of information determined at each iteration. In the GPST attack, the change of basis matrix between Bob's correct points and his malformed points can be thought of as acting on the subgroup $E[N_1]$ containing Alice's private point (as described in Section 4). In this interpretation, if Alice's private point is an eigenvector of this change of basis matrix, then the protocol succeeds, otherwise (with overwhelming probability) the protocol fails.

The optimal situation for Bob is if the protocol, at each iteration of the GPST attack, succeeds exactly half the time. This way, he discovers a bit of

information of Alice's private point during each iteration. Our observation is that the protocol succeeds exactly half the time because Bob chooses a change of basis matrix such that Alice's private point is an eigenvector exactly half of the time. More generally, if Alice's private point is an eigenvector of the change of basis matrix with a probability $\rho < 1/2$, then Bob is expected to discover $H(\rho)$ bits of information about Alice's private key (the information entropy of ρ). Based on the success or failure of the protocol with the malformed image points, Bob can adapt the change of basis matrix, and repeat a similar procedure, learning $H(\rho)$ bits of Alice's private key with each iteration.

	GPST attack	Our work
Mode	Active	Passive
Algorithm type	Attack	Reduction (requires Oracle queries)
Target	Static SIDH	Instantiated SSDDH
Objects	Change of basis matrices	Endomorphisms
Bits recovered	1 bit	$H(\rho)$ bits
Runtime	Poly-time	Depends on degree of endomorphism
Memory	Poly-space	Poly-space

Table 1: Comparison of the GPST attack with our work.

Instead of a change of basis matrix, our methods assume that the reduction has access to desirable endomorphisms on the starting curve. However, such endomorphisms also act as a matrix on the subspace that contains all of Alice's possible private points. As with the GPST attack, if Alice's private point is an eigenvector of the endomorphism with a high probability $\rho \leq 1/2$, then, using our methods, the attacker learns close to one bit of information of Alice's private key. By repeating this attack with different endomorphisms, the attacker can discover Alice's private key. Therefore, in order to gain a significant amount of information about Alice's private point, the attacker uses endomorphisms where Alice's private point is an eigenvector of the endomorphisms with a reasonably high probability.

Given an endomorphism (with many eigenvectors), the algorithm in our reduction requires the adversary to construct an isogeny on Alice's curve of the same degree. The runtime of our algorithm depends on the time it takes the attacker to construct this isogeny. As the endomorphism and the related isogeny have the same degree, the difficulty of constructing this related isogeny roughly grows with the degree of the endomorphism. Thus, we are interested in using endomorphisms that have particular degrees.

Alice's public key includes the image under her private isogeny of the subspace of points whose order divides $N_2 = 3^b$. As we will see in Section 5, this additional information makes it easy to construct an isogeny on Alice's curve E_A that is the composition of three isogenies, where the first and last isogenies have degrees dividing N_2 and the middle isogeny is ideally of small degree. Thus, we are interested in endomorphisms (on the starting curve) whose degrees have the form kL where $L \mid N_2^2$ and k is small and coprime to N_1. One of our contributions in

that section is the notion of a triangular kernel (see Section 5), which is a triple of points that generates the isogenies in this decomposition.

In summary, for our reduction to be practical, it requires endomorphisms on the starting curve with large eigenspaces and certain conditions on the degrees. As above, by repeating this procedure with many endomorphisms, Alice's private key can be recovered.

In order to investigate the existence of desirable endomorphism, we can exploit the knowledge of the endomorphism ring of the starting curve of SIDH/SIKE. In this case, the problem of finding desirable endomorphisms is reduced to solving a particular quadratic form. By studying the quadratic forms arising in the SIKE setting, we are able to show the following non-existence result: no desirable endomorphisms exist that make the above attack strategy on SIKE more efficient than known attacks. A similar result holds for the standard parameterizations of SIDH.

However, investigating the quadratic form arising in multi-party isogeny-based protocols (where the structure of the prime is modified to $p = N_1 \cdot \ldots \cdot N_n - 1$, for coprime natural numbers $N_1, \ldots, N_n$) leads us to heuristic arguments about the existence of desirable endomorphisms for these cases. This suggests that there exist endomorphisms that could be used to give an improved attack in the 4-party case. That being said, it is unclear how difficult it is to construct such endomorphisms. In the 3-party case we do not have heuristics for why such endomorphisms should exist, but if they do, then our cryptanalysis improves upon the best-known attacks in the literature.

We also show a trade-off between the amount of information about Alice's private point that an endomorphism reveals, and how efficiently this information can be determined. This trade-off is between the size of the eigenspace of the desirable endomorphism and the degree of the endomorphism on the image curve (specifically, the part of the endomorphism that needs to be searched for exhaustively).

Additionally, we show that if Alice's key is found to be in the eigenspace of two (independent) desirable endomorphisms, then we can improve our reduction. In particular, these endomorphisms can be used to find information about the image of torsion points on the starting curve under Alice's private isogeny. Thus, instead of just using the torsion information in Alice's public key, we have additional torsion information to exploit.

We conclude this paper with a recommendation consisting of a SIKE-like protocol which randomizes the starting curve, and thus avoids any potential future attack that utilizes the known endomorphism ring of the starting curve.

4 Interpreting Private Keys as Eigenvectors

In this section, we reformulate the work of GPST [11] to motivate our approach to solving isogeny problems. Section 4.1 begins our investigation into cryptanalysis by recalling the GPST attack, the most devastating attack on the CPA variant of SIDH to date, and reformulating it in terms of whether or not a party's private

point is an eigenvector of a change of basis matrix. In Section 4.2 we replace the matrices in the GPST attack by endomorphisms, and this gives us our first results concerning whether or not a party's private point is an eigenvector of an endomorphism.

4.1 The GPST Active Attack on SIDH

In 2015, Galbraith, Petit, Shani and Ti [11] introduced an active attack on users with static SIDH keys who do not use a Fujisaki-Okamoto type transformation [16] to verify the other communicating party's public key. This attack was originally formulated in terms of Bob's public key containing a malicious linear combination of the points $\phi_B(P_A)$ and $\phi_B(Q_A)$. It has since been observed that Bob is, in essence, altering the points of $\phi_B(P_A)$ and $\phi_B(Q_A)$ using a change of basis matrix [25], where the matrix has entries in $\mathbb{Z}/N_1\mathbb{Z}$ and is chosen to have determinant 1 to avoid detection (using the Weil-pairing test [21, §III.8]). In this subsection, we will rephrase the GPST attack by noticing that these change of basis matrices are designed to have large eigenspaces.

Notation 4.1. Throughout this section, $p = 2^a 3^b - 1$, where $N_1 = 2^a \approx 3^b = N_2$.

In this subsection we assume that the adversary, Bob, is trying to find Alice's private key. A similar analysis could be done to attack Bob's private key (see [11] for the details).

Specifically, in this attack, Bob maliciously alters the image points $\phi_B(P_A)$ and $\phi_B(Q_A)$ to another linear combination of those points before sending them to Alice. Depending on whether or not Alice and Bob compute the same shared secret key, by observing if Alice terminates the session, Bob can deduce one bit of information of Alice's private key r_A. By repeating this attack with n different linear combinations of the image points $\phi_B(P_A)$ and $\phi_B(Q_A)$, Bob can discover all n bits of Alice's private key. This attack is devastating against static keys.

We provide the first iteration in the GPST attack as an illustration.

Attack (GPST Attack Iteration 1). Suppose that instead of sending Alice his public key $E_B, \phi_B(P_A), \phi_B(Q_A)$, Bob maliciously sends $E_B, \phi_B(P_A), \phi_B(Q_A) + [2^{a-1}] \cdot \phi_B(P_A)$. Then Alice follows the protocol and calculates ψ'_A with kernel

$$\langle \phi_B(P_A) + [r_A] \cdot (\phi_B(Q_A) + [2^{a-1}] \cdot \phi_B(P_A)) \rangle,$$

although Alice believes that she is calculating ψ_A with kernel

$$\langle \phi_B(P_A) + [r_A] \cdot \phi_B(Q_A) \rangle.$$

Meanwhile, Bob calculates ψ_B with kernel

$$\langle \phi_A(P_B) + [r_B] \cdot \phi_A(Q_B) \rangle.$$

We know from the theory of isogenies, that $j(\psi_A(E_B)) = j(\psi_B(E_A))$. The order of $\phi_B(P_A)$ is 2^a, which implies $[2^{a-1} r_A] \cdot P_A = 0$ if and only if r_A is even. Thus, if r_A is even then $j(\psi'_A(E_B)) = j(\psi_A(E_B)) = j(\psi_B(E_A))$. Conversely, if r_A is odd, then almost always $j(\psi'_A(E_B)) \neq j(\psi_B(E_A))$. Therefore, if the protocol runs correctly, then r_A is (almost always) even, otherwise r_A is odd.

The first iteration reveals a single bit of information about Alice's private key. In each of the following n iterations of the GPST attack, Bob adaptively adjusts the linear combination of points in his public key to reveal additional bits of Alice's private key.

It has been observed that adaptively adjusting the linear combination of points is the same as altering the points of $\phi_B(P_A)$ and $\phi_B(Q_A)$ using a change of basis matrix (i.e., an invertible linear transformation) [25]. However, what has not been previously observed is that the GPST attack is possible because Bob uses change of basis matrices with large eigenspaces.

Along these lines, a better way to think of the GPST attack is that it exploits whether or not Alice's secret kernel, $\ker \phi_A$, is invariant under particular change of basis matrices to determine r_A. We state this reinterpretation of the GPST attack in the language of linear algebra in the following discussion. This perspective will be useful in Section 5, where we will use similar language to describe potential offline attacks where Alice's kernel may be invariant under some particular endomorphisms.

Remark 4.2. Given a linear map M on $E[N]$, a subgroup G of $E[N]$, and an isogeny ϕ_B with $\gcd(\deg \phi_B, N) = 1$, if $M(G) = G$, then the isogenies with kernels $\langle \phi_B(G) \rangle$ and $\langle \phi_B(M(G)) \rangle$ are equal.

The following notation will transform Remark 4.2 into the framework of the GPST attack (see Proposition 4.5).

Notation 4.3. Suppose Bob chooses a linear transformation M on $E[2^a]$. We will represent M as a matrix in $\mathrm{SL}_2(\mathbb{Z}/2^a\mathbb{Z})$ with respect to the basis $\{P_A, Q_A\}$. We also represent $P_A + [r_A] \cdot Q_A$ as the vector $\left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]$, and $\phi_B(P_A) + [r_A] \cdot \phi_B(Q_A)$ as $\phi_B\left(\left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]\right)$.

When Bob sends the malicious points to Alice, he is actually using a change of basis matrix to send a different basis of $E_B[2^a]$.

Lemma 4.4. *If M is an invertible matrix acting on $E[2^a]$, and R is an eigenvector of M, then $M(\langle R \rangle) = \langle R \rangle$.*

Proof. Since M is invertible, $|M(R)| = |R|$. Thus $M(\langle R \rangle) = \langle R \rangle$. □

Substituting Lemma 4.4 into Remark 4.2 gives us the following proposition, which is the essence of the GPST attack. To make the GPST attack practical the change of basis matrices are chosen to have a high percentage of eigenvectors.

Proposition 4.5. *Suppose M is an invertible matrix acting on $E[2^a]$. If $\left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]$ is an eigenvector of M, then the isogeny ψ_A whose kernel is generated by $\phi_B\left(\left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]\right)$ is equal to the isogeny ψ'_A whose kernel is generated by $\phi_B\left(M\left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]\right)$.*

Proof. Letting $R = \left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]$ in Lemma 4.4, we find that $M(\langle R \rangle) = \langle R \rangle$. The result follows from letting $G = \langle R \rangle$ in Remark 4.2. □

To see how Proposition 4.5 provides a new interpretation of the GPST attack, notice that in Iteration 1 of the attack, if $\left[\begin{smallmatrix}1\\ r_A\end{smallmatrix}\right]$ is an eigenvector of $M = \left[\begin{smallmatrix}1 & 0\\ 2^{a-1} & 1\end{smallmatrix}\right]$, then $\psi_A = \psi'_A$ (the converse is almost always true). Iteration 1 is efficient because Alice's private key is an eigenvector of M with probability $1/2$. Thus, this iteration allows the attacker to gain a bit of information about Alice's private key. The change of basis matrix M in each successive iteration in the GPST attack is adaptively chosen so that Alice's private key is an eigenvector of M with probability $1/2$. This allows the attacker to gain an additional bit of information about Alice's key per iteration.

The central idea of this paper is to replace the change of basis matrix in Proposition 4.5 by an endomorphism with many eigenvalues, and thus design an offline algorithm which repeatedly restricts the search-space for recovering a private isogeny in SIDH-like protocols.

4.2 GPST-Inspired Cryptanalysis

We now describe a first attempt at generalizing the GPST attack to the language of endomorphisms. Although our ideas were inspired by the active GPST attack, our aim differs in that our work is towards an offline attack on private keys; that is, our methods do not require any participation from Alice after she provides her public key, nor multiple interactions with her. Unless otherwise specified, we will aim to attack Alice's private key, although our methods can be applied to any party.

We note that the approach given in this subsection is not meant to be practical and certainly is not useful in the SIKE setting. However, it presents the basic ideas upon which the rest of this work is built. Specifically, although this subsection does not use the (image) points provided in Alice's public key, a more practical approach of attacking isogeny-based algorithms is presented in Section 5 that does utilize these points.

The attacker chooses a particular endomorphism ϕ_C on E that will mimic the role the matrix M played in the GPST attack, as presented in Section 4.1. We introduce the following notation related to ϕ_C that distinguishes the commutative diagram in Figure 1 as being different from the usual SIDH setup (see Figure 20). This endomorphism ϕ_C should be thought of as different from ϕ_B for two reasons: it is an endomorphism (an isogeny from E to itself), and it will not be used as part of an SIDH key exchange.

Notation 4.6. In addition to the notation of Section 2.1, let ϕ_C be a cyclic endomorphism on E of degree k that is generated by a point R_C in $E(\overline{\mathbb{F}}_{p^2})$, where $\gcd(k, N_1) = 1$.

Let $\{P_C, Q_C\}$ denote a basis for $E[k]$. Without loss of generality we can assume $R_C = P_C + [r_C]\cdot Q_C$ for some r_C. Let ψ_C denote the isogeny on E_A with the kernel $\langle \phi_A(R_C)\rangle$, and ψ_A^C denote the isogeny on E with kernel $\langle \phi_C(P_A + [r_A]\cdot Q_A)\rangle$, as shown in Figure 1. (Here the superscript C in ψ_A^C refers to the fact that its kernel is the image of $\ker \phi_A$ under the map ϕ_C, as opposed to ψ_A in SIKE whose kernel is the image of $\ker \phi_A$ under ϕ_B in Figure 20.)

As ϕ_C acts as a linear transformation on $E[N_1]$, we can let $M \in \mathrm{SL}_2(\mathbb{Z}/N_1\mathbb{Z})$ model the action of ϕ_C on $E[N_1]$ with respect to the basis $\{P_A, Q_A\}$. Let E_{CA} be the image of ψ_C, and let E_{AC} be the image of ψ_A^C.

As $\gcd(k, \deg \phi_A) = 1$, the matrix M acts on $\{P_A, Q_A\}$ as an (invertible) change of basis matrix. Also, as $\gcd(k, N_1) = 1$, the square in Figure 1 commutes; that is, $\psi_C \circ \phi_A = \psi_A^C \circ \phi_C$.

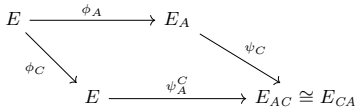

Fig. 1: Commutative diagram with endomorphism ϕ_C

With this new notation, we observe the following proposition (similar to Proposition 4.5). Proposition 4.7 will allow us to replace the active part of the GPST attack with an offline computation.

Proposition 4.7. *Suppose $P_A + [r_A] \cdot Q_A$ is a point of order N_1 on E and $\phi_A : E \to E_A$ is an isogeny with kernel $\langle P_A + [r_A] \cdot Q_A \rangle$. If $P_A + [r_A] \cdot Q_A = \left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]$ is an eigenvector with respect to some endomorphism on E of degree k, with $\gcd(k, N_1) = 1$, then there exists an endomorphism on E_A of degree k.*

Proof. Suppose ϕ_C is an endomorphism on E of degree k, such that Alice's private point $P_A + [r_A] \cdot Q_A$ is an eigenvector with respect to ϕ_C. As $\gcd(k, N_1) = 1$, we see that ϕ_C acts as an invertible linear transformation on $E[N_1]$. Thus by Lemma 4.4

$$\langle P_A + [r_A] \cdot Q_A \rangle = \phi_C(\langle P_A + [r_A] \cdot Q_A \rangle).$$

As ϕ_A has kernel $\langle P_A + [r_A] \cdot Q_A \rangle$ and ψ_A^C has kernel $\phi_C(\langle P_A + [r_A] \cdot Q_A \rangle)$ (see Figure 1), we see that $\phi_A \cong \psi_A^C$. Thus

$$E_{CA} \cong E_{AC} = \psi_A^C(E) \cong \phi_A(E) = E_A.$$

Therefore, ψ_C is an endomorphism on E_A of degree k. □

We now explicitly describe the kernel of this endomorphism for later use.

Corollary 4.8. *If R_A is an eigenvector of ϕ_C, then the endomorphism of degree k from Proposition 4.7 has kernel $\phi_A(\ker \phi_C)$.*

If $k = \deg \phi_C$ is small enough to allow us to brute-force the computation of all codomain j-invariants of all k-isogenies from E_A, then it is easy to use Proposition 4.7 to test if $\left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]$ is an eigenvector of ϕ_C or not. This is made concrete in the following theorem.

Theorem 4.9. *Suppose we are given*

1. *a supersingular elliptic curve $E(\mathbb{F}_{p^2})$ such that $p = N_1 N_2 - 1$ for coprime N_1 and N_2,*
2. *the image of an N_1-degree isogeny $E_A = \phi_A(E)$ with kernel $\langle R_A \rangle$, and*

3. *k such that there exists a k-endomorphism ϕ_C of E, where* $\gcd(k, N_1) = 1$ *and* $k < N_1$.

Then there exists a (classical) algorithm with worst case runtime $\tilde{O}(k^3)$ which decides whether $R_A \in \mathsf{Eig}_{N_1}(\phi_C)$ or $R_A \notin \mathsf{Eig}_{N_1}(\phi_C)$ with overwhelmingly high probability. Further, if k is $\log p$-smooth, then the runtime is $\tilde{O}(\sqrt{k})$.

Proof. By Proposition 4.7 and since $k < N_1$, it follows that we need to examine the difficulty of testing if E_A is k-isogenous to E_A. We will examine to the main two computations involved: constructing a field extension for which $E_A[k]$ is defined, and then the computation of degree k isogenies.

We begin by discussing the difficulty of finding an appropriate extension. Factoring the k-th division polynomial over $\mathbb{F}_{p^2}$ will give an irreducible polynomial of degree k (many exist, but any will suffice) which will give an appropriate field extension to contain all x-coordinates of $E_A[k]$. The degree of this division polynomial is $\frac{k^2-1}{2}$ and the polynomial requires this much time and space to compute. Therefore, by [15], finding a root of this polynomial takes $\tilde{O}(k^3)$ time. A quadratic extension on this field will then be guaranteed to contain the y-coordinates as well, and thus all of $E_A[k]$. However, when k is, say D-smooth, this field can be constructed as a tower of extensions, and thus only takes $O(D \log k) = O(\log^2 p)$ time.

Next we assume the field extension has been constructed. When k is prime, constructing all k-isogenies with domain E_A using Vélu's formulas involves computing the $k+1$ isogenies of prime degree k and domain E_A. Prime degree isogenies currently require $O(k)$ operations to compute [27]. 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 $k = k_1 k_2$ for some $\log p$-smooth positive integers k_1 and k_2 each approximately of size $\sqrt{k}$, then 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 [13, 5.1], and $O(\sqrt{k})$ space. When k is $\log p$-smooth, then the isogeny computations themselves are $O(\log k)$ which is negligible.

Thus, the worst case for this iteration 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 $100,000$ [23]) this computation can be performed by checking if the tuple $(j(E_A), j(E_A))$ is a root of the k^{th} modular polynomial. Therefore, the runtime of this step is dominated by the cost of creating a field extension, namely $\tilde{O}(k^3)$.

By Proposition 4.7, if R_A is an eigenvector of ϕ_C then the above process will succeed as E_A must have an endomorphism of degree k. If R_A is not an eigenvector, then a false-positive endomorphism may exist, but is highly improbable when $k < N_1$. □

Remark 4.10. As we see from the proof, the best case for this algorithm is if k is a $\log p$-smooth integer, where the algorithm takes approximately $\sqrt{k}$ time and space (logarithmic factors omitted).

Suppose that for some $\rho = \omega(1/\operatorname{poly}(\lambda))$, there is an endomorphism ϕ_C with small degree k, such that the conditions

$$|E[N_1] \cap \mathsf{Eig}_{N_1}(\phi_C)| \leq (1-\rho) \cdot |S|, \text{ and} \tag{1}$$

$$|E[N_1] \setminus \mathsf{Eig}_{N_1}(\phi_C)| \leq (1-\rho) \cdot |S| \tag{2}$$

hold. By applying the result of Theorem 4.9, we can discover

$$H(\rho) = \rho \log_2(1/\rho) + (1-\rho)\log_2(1/(1-\rho))$$

bits of information about Alice's key.

Remark 4.11. Although there are matrices on $E[N_1]$ (for instance, those used in the GPST attack, see [11]) that satisfy Conditions 1 and 2, if p has the standard form $p = 2^a 3^b - 1$, then there is no endomorphism that satisfies these conditions with degree $k \in O(p)$ (see Corollary A.3). Thus the algorithm referred to in Theorem 4.9 does not give a viable attack on SIDH.

This theorem does, however, provide the basic premise on which the rest of this work is built.

5 Exploiting Endomorphisms

The goal of this section is to prove our first main result, Theorem 5.11, which reduces the security of SIDH/SIKE variants to the problem of finding certain types of endomorphisms of the starting elliptic curve, which we will refer to as *desirable endomorphisms*. We will prove the main result of this section by giving a stronger version of the algorithm from Theorem 4.9 which utilizes the torsion group information given in Alice's public key.

5.1 Triangular Kernels

The input of the algorithm from Theorem 4.9 is E, E_A, and $\ker \phi_C$, where E is a public parameter of the system, E_A is part of Alice's public key, and $\ker \phi_C$ is a pre-computed endomorphism with a low degree and many eigenvectors.

However, in isogeny-based key establishments, Alice's public key contains more information than simply E_A. It also includes the image under ϕ_A of a large torsion subgroup, namely $\phi_A|_{E[N_2]}$. Thus, from Alice's public key, it is efficient to calculate an isogeny whose kernel is contained in $E_A[N_2]$. Moreover, it is faster (than simply using brute-force) to calculate an isogeny whose kernel has a large intersection with $E_A[N_2]$. In this section, we will formalize this idea.

Suppose we have an endomorphism ϕ_C of degree Lk, where $L \mid N_2$, $\gcd(k, N_1) = 1$ and $k < N_1$. Then, there exist isogenies $\phi_{C,1}$ and $\phi_{C,0}$ on E such that $\phi_C = \widehat{\phi}_{C,0} \circ \phi_{C,1}$, and $\ker \phi_{C,0} = \ker \phi_C \cap E[N_2]$, see Figure 2. Knowledge of Alice's public key implies that it is efficient to calculate $\phi_A(\ker \phi_{C,0})$, and so this fact can be used to make Theorem 4.9 faster in the case where $\gcd(\deg \phi_C, N_2) > 1$.

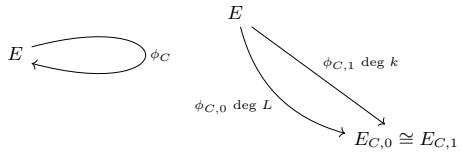

Fig. 2: Decomposing endomorphisms (simple)

In fact, not only are we able to improve Theorem 4.9 if $\gcd(\deg \phi_C, N_2)$ is large, but we can improve Theorem 4.9 even more if $\gcd(\deg \phi_C, N_2^2)$ is larger than $\gcd(\deg \phi_C, N_2)$. When this holds, there is another natural way to decompose ϕ_C. This leads us to introduce a new definition that captures the concept behind this type of decomposition.

Definition 5.1. Suppose ϕ_C is a cyclic endomorphism of E. A *triangular decomposition* of ϕ_C with respect to N_2 is a triple of cyclic isogenies $\phi_{C,0}, \phi_{C,1}, \phi_{C,2}$, where $\phi_{C,0}$ and $\phi_{C,1}$ have degrees dividing N_2,

$$\phi_C = \widehat{\phi}_{C,0} \circ \phi_{C,2} \circ \phi_{C,1},$$

and if $\gcd(N_2, \deg \phi_{C,2}) \neq 1$, then $\deg \phi_{C,0} = \deg \phi_{C,1} = N_2$.

A *triangular kernel* of ϕ_C with respect to N_2 is a triple of torsion points denoted by $\ker_\triangle \phi_C = (K_0, K_1, K_2)$, which generate the kernels of the corresponding isogenies of a triangular decomposition, that is, $\ker \phi_{C,i} = \langle K_i \rangle$. Furthermore, let $k = |K_2|$.

Remark 5.2. This representation has the advantage that only the extension field containing the k-torsion points is needed to write the kernel, instead of the $(N_2)^2 k$-torsion points. This is because $K_0, K_1 \in E[N_2]$.

Notice that K_0, K_1 and K_2 could theoretically all be trivial.

Notation 5.3. Let $\phi_{C,0}, \phi_{C,1}, \phi_{C,2}$ denote a triangular decomposition of ϕ_C with respect to N_2. Let $E_{C,0}, E_{C,1}$ and $E_{C,1,2}$ denote the images of $\phi_{C,0}, \phi_{C,1}, \phi_{C,2}$, respectively, as illustrated in Figure 3. Then, up to isomorphism, $E_{C,0} \cong E_{C,1,2}$.

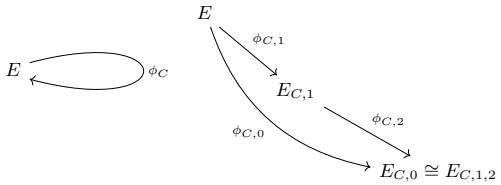

Fig. 3: Decomposing endomorphisms

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

Notation 5.4. Let $\psi_{C,0}$ be the isogeny with domain E_A and kernel $\phi_A(\ker \phi_{C,0})$, and $\psi_{A,0}^C$ be the isogeny with domain $E_{C,0}$ and kernel $\phi_{C,0}(\ker \phi_A)$. Let the images be $E_{CA,0} = \psi_{C,0}(E_A)$ and $E_{AC,0} = \psi_{A,0}^C(E_{C,0})$, as illustrated in Figure 4.

Since $\deg \phi_{C,0}$ and $\deg \phi_A$ are relatively prime, $E_{CA,0} \cong E_{AC,0}$.

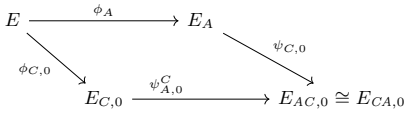

Fig. 4: Maps with known kernels

Notation 5.5. We decompose the isogeny with domain E_A and kernel equal to $\phi_A(\ker \phi_{C,2} \circ \phi_{C,1})$ as $\psi_{C,2} \circ \psi_{C,1}$, where $\ker \psi_{C,1} = \phi_A(\ker \phi_{C,1})$ and $\ker \psi_{C,2} = \psi_{C,1} \circ \phi_A(\ker \phi_{C,2})$. Let $\psi^C_{A,1,2}$ be the isogeny with domain $E_{C,1,2}$ whose kernel is $\phi_{C,2} \circ \phi_{C,1}(\ker \phi_A)$. We will let the images be $E_{CA,1} = \psi_{C,1}(E_A), E_{CA,1,2} = \psi_{C,2}(E_{CA,1})$ and $E_{AC,1,2} = \psi^C_{A,1,2}(E_{C,1,2})$.

Note that $\deg \psi_{C,1} = \deg \phi_{C,1}$ (which implies, $\deg \psi_{C,1} \mid N_2$), and $\deg \psi_{C,2} = \deg \phi_{C,2}$. Since $\deg \phi_{C,2} \circ \phi_{C,1}$ and $\deg \phi_A$ are relatively prime, $E_{CA,1,2} \cong E_{AC,1,2}$ as shown in Figure 5.

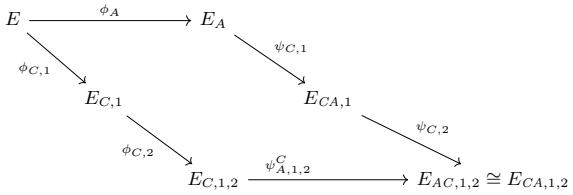

Fig. 5: Maps with known and unknown kernels

Now Bob can calculate $\psi_{C,0}$ and $\psi_{C,1}$, since Alice sent him $\phi_A|_{E[N_2]}$. Putting the previous two diagrams together gives us Figure 6.

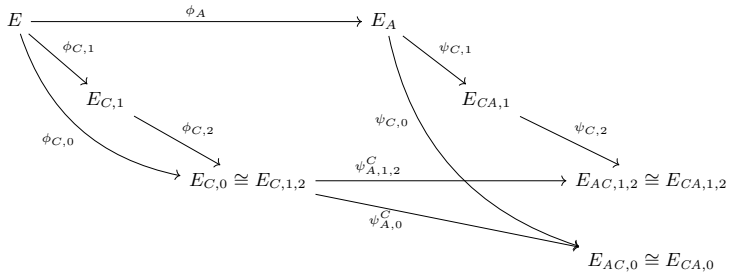

Fig. 6: Combining Figures 4 and 5

As our goal is to adapt the results of Section 4.2 to incorporate the torsion information revealed by Alice, we next present the analogous Proposition 4.7.

Lemma 5.6. *Suppose* $\gcd(k, N_1) = 1$. *If R_A is an eigenvector with respect to ϕ_C, then $E_{CA,1}$ is k-isogenous to $E_{CA,0}$.*

Proof. Choose a triangular decomposition of ϕ_C as follows:

$$\phi_C = \widehat{\phi}_{C,0} \circ \phi_{C,2} \circ \phi_{C,1}.$$

Let $\phi'_C = \phi_{C,2} \circ \phi_{C,1} \circ \widehat{\phi}_{C,0}$. Then ϕ'_C is an endomorphism of $E_{C,0}$. Moreover,

$$\begin{aligned}\phi'_C(\phi_{C,0}(R_A)) &= \phi_{C,2} \circ \phi_{C,1} \circ \widehat{\phi}_{C,0}(\phi_{C,0}(R_A)) \\ &= \phi_{C,2} \circ \phi_{C,1}([\deg \phi_{C,0}] \cdot R_A) \\ &= [\deg \phi_{C,0}] \cdot \phi_{C,2} \circ \phi_{C,1}(R_A) \\ &= \phi_{C,0}(\widehat{\phi}_{C,0} \cdot \phi_{C,2} \circ \phi_{C,1})(R_A).\end{aligned}$$

However, as R_A is an eigenvector of ϕ_C (with eigenvalue λ where $\gcd(\lambda, kN_2) = 1$), this implies

$$\phi'_C(\phi_{C,0}(R_A)) = [\lambda] \cdot \phi_{C,0}(R_A).$$

Thus $\phi_{C,0}(R_A)$ is an eigenvector of ϕ'_C.

Recall that $\langle \phi_{C,0}(R_A) \rangle$ is the kernel of the isogeny $\psi^C_{A,0}$ on $E_{C,0}$, see Figure 7. Therefore, we can apply Corollary 4.8, and so ψ'_C is an endomorphism on $E_{AC,0}$, where $\ker \psi'_C = \psi^C_{A,0}(\ker \phi'_C)$.

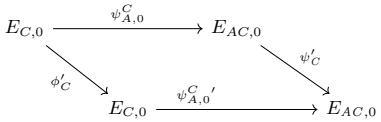

Fig. 7: ϕ'_C fixing the kernel of $\psi^C_{A,0}$

Let ${\psi^C_{A,0}}'$ be the isogeny on $E_{C,0}$ with kernel $\phi'_C(\ker \psi^C_{A,0})$. Since $\gcd(k, N_1) = 1$, the following equation holds: ${\psi^C_{A,0}}' \circ \phi'_C \cong \psi'_C \circ \psi^C_{A,0}$.

It remains to be shown that $E_{AC,0} \cong E_{AC,1,2}$, that is, the codomain of ${\psi^C_{A,0}}'$ is isomorphic to $E_{AC,0}$. Similar to above we find

$$\begin{aligned}\ker {\psi^C_{A,0}}' &= \phi'_C(\ker \psi^C_{A,0}) \\ &= \phi_{C,2} \circ \phi_{C,1} \circ \widehat{\phi}_{C,0}(\langle \phi_{C,0}(R_A) \rangle) \\ &= \phi_{C,2} \circ \phi_{C,1}(\langle [\deg \phi_{C,0}] \cdot R_A \rangle) \\ &= [\deg \phi_{C,0}] \cdot \ker(\psi^C_{A,1,2}) \\ &= \ker(\psi^C_{A,1,2}).\end{aligned}$$

Then,

$$\begin{aligned}E_{AC,1,2} &\cong \psi^C_{A,1,2}(E_{C,1,2}) \cong {\psi^C_{A,0}}'(E_{C,0}) \\ &\cong \psi^C_{A,1,2} \circ \phi'_C(E_{C,0}) \cong \psi'_C \circ \psi^C_{A,0}(E_{C,0}) \cong E_{AC,0}.\end{aligned}$$

Thus $E_{AC,0}$ and $E_{AC,1,2}$ are isomorphic. Hence

$$E_{CA,0} \cong E_{AC,0} \cong E_{AC,1,2} \cong E_{CA,1,2}.$$

However, $\psi_{C,2}$ is a k-isogeny between $E_{CA,1}$ and $E_{CA,1,2}$. This implies $E_{CA,1}$ and $E_{CA,0}$ are k-isogenous. □

If k is small enough, then Lemma 5.6 will prove useful in the coming reduction. We note the similarity of Theorem 5.7 to Theorem 4.9, except that it allows us to extract a factor of up to N_2^2 out of the runtime of Theorem 4.9.

Theorem 5.7. *Suppose we are 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 curve of an* N_1*-degree isogeny* $E_A = \phi_A(E)$ *with kernel* $\langle R_A \rangle$*,*
3. *the action* $\phi_A|_{E[N_2]}$*, and*
4. *a triangular kernel* $\ker_\triangle \phi_C$ *of a cyclic* Lk*-degree endomorphism* ϕ_C *in* $\mathbb{F}_{p^2}$ *such that:*
   (a) $\gcd(k, N_1) = 1$*,*
   (b) $L \mid (N_2)^2$*, and*
   (c) $k < N_1$*.*

Then there exists a (classical) algorithm with worst case runtime $\tilde{O}(k^3)$ *which decides whether* $R_A \in \mathsf{Eig}_{N_1}(\phi_C)$ *or* $R_A \notin \mathsf{Eig}_{N_1}(\phi_C)$ *with overwhelmingly high probability. Further, if* k *is* $\log p$*-smooth, then the runtime is* $\tilde{O}(\sqrt{k})$*.*

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 (our algorithm saying $R_A \in \mathsf{Eig}_{N_1}(\phi_C)$ when $R_A \notin \mathsf{Eig}_{N_1}(\phi_C)$), can be approximated using the mixing properties of the isogeny graph.

Algorithm 5.8.
Input: $E, p, N_1, N_2, \{P_A, Q_A\}, E_A, \phi_A|_{E[N_2]}, \ker_\triangle \phi_C = (K_0, K_1, K_2)$, and a natural number $k = |K_2|$. Note: K_2 is not actually needed, only $k = |K_2|$.
Output: True if $R_A \in \mathsf{Eig}_{N_1}(\phi_C)$, and False if $R_A \notin \mathsf{Eig}_{N_1}(\phi_C)$

1. Use $\phi_A|_{E[N_2]}$ to compute $\phi_A(K_0)$ and $\phi_A(K_1)$.
2. Compute the isogenies $\psi_{C,0}$, and $\psi_{C,1}$ with respective kernels $\langle \phi_A(K_0) \rangle$ and $\langle \phi_A(K_1) \rangle$ (see Figure 6).
3. For all k-isogenies from $E_{CA,0}$, check if their codomain has j-invariant $j(E_{CA,1})$.
4. If one does, then return True, otherwise return False.

Proof. First, we discuss the success probability. If Algorithm 5.8 returns False, then $\left[\begin{smallmatrix} 1 \\ r_A \end{smallmatrix}\right]$ is not an eigenvector with respect to ϕ_C by the contrapositive of Lemma 5.6. Suppose Algorithm 5.8 returns True. Notice that the total number of non-backtracking isogenies from $E_{CA,0}$ 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 families of 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 $E_{CA,0}$ and $E_{CA,1}$ 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 $E_{CA,1}$ to E_A, then the kernel subgroup is fixed by ϕ_C.

Next, we discuss the runtime. Step 1 and Step 2 are efficient in p since N_2 is $\log p$-smooth. The analysis of verifying when $E_{CA,1}$ and E_A are k-isogenous is identical to the proof of Theorem 4.9. Thus, the worst case is when k is prime, with runtime $O(k^3)$. □

It follows from this theorem that if k is small enough, then it will be feasible to test if an unknown $\ker \phi_A$ is fixed by an endomorphism ϕ_C or not. Algorithm 5.8 will be a subroutine in our main reduction (Theorem 5.11). That reduction will assume an oracle which outputs triangular kernels of endomorphisms, and then use Algorithm 5.8 with each of those endomorphisms. In Section 7.3 we demonstrate that the SIKE/SIDH starting curve likely does not have an endomorphism which satisfy the conditions of Theorem 5.7.

5.2 Main Theorem

In this section, we present the first main result of the paper, Theorem 5.11. We prove this result by describing Algorithm 5.12 and analyzing its runtime. We start by presenting Oracle 5.9, 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, see Remark 5.2. Since we are no longer discussing a single Lk-isogeny, with $k \leq N_1$, but potentially multiple from repeated calls to an oracle, we instead use $K \leq N_1$ to denote the upper bound on all such k. We also introduce the variable ρ to quantify the amount of information each endomorphism provides. The closer ρ is to $1/2$, the closer the endomorphism is to providing a full bit of information on Alice's private key (by the definition of $H(\rho)$).

Oracle 5.9.
Input: E, p, N_2, a set $S \subseteq E[N_1]$, an integer $K \leq N_1$, and $\rho \in (0, 1/2]$ satisfying $\rho = \omega(1/\operatorname{poly}(\lambda))$.
Output: The $\ker_{\triangle} \phi_C = (K_0, K_1, K_2)$ of a cyclic endomorphism ϕ_C such that the following constraints hold:

1. $|K_2| \leq K$,
2. $\gcd(|K_2|, N_1) = 1$,
3. $|S \cap \mathsf{Eig}_{N_1}(\phi_C)| \leq (1-\rho) \cdot |S|$, and
4. $|S \setminus \mathsf{Eig}_{N_1}(\phi_C)| \leq (1-\rho) \cdot |S|$,

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

With the following definition, we give a name to the endomorphisms output by Oracle 5.9.

Definition 5.10. We call an endomorphism that satisfies the conditions of Oracle 5.9 a *desirable endomorphism*.

The next theorem gives our main reduction. In essence, it states that each endomorphism ϕ_C returned by Oracle 5.9 can be used to gain information about Alice's private key r_A. More specifically, it is possible to use Algorithm 5.8 to decide whether or not R_A is in the eigenspace of each endomorphism ϕ_C.

Theorem 5.11. *Suppose we are 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 $E_A = \phi_A(E)$,*
3. *the action $\phi_A|_{E[N_2]}$, and*
4. *access to Oracle 5.9, $\mathcal{O}$, such that for an overwhelming fraction of sets S, $\mathcal{O}$ will succeed for a non-negligible fraction of $K \in \{0, \ldots, N_1\}$ and $\rho \in \left[\frac{1}{f(\lambda)}, \frac{1}{2}\right]$, where f is some fixed polynomial.*

Then there exists a (classical) algorithm which outputs r_A, where $\ker \phi_A = \langle R_A \rangle$, 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 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 Theorem 5.11. At a high level, Algorithm 5.12 iteratively reduces the size of the search space, which is denoted S_i at the i^{th} iteration, for Alice's private point. Step 5 is not required to prove the runtimes as stated in Theorem 5.11, however, we include it to highlight operational improvements that can be made.

Algorithm 5.12.
Input: $E, p, N_1, N_2, \{P_A, Q_A\}, E_A, \phi_A|_{E[N_2]}$, the polynomial f, and access to Oracle 5.9 denoted $\mathcal{O}$.
Output: r_A or $\perp$.

1. Let $S_0 = \{P_A + [r] \cdot Q_A \mid 0 \leq r < N_1\}$, and $i = 0$.
2. Set $\rho = 1/f(\lambda)$.
3. Set $K = N_1$.
4. Call $\mathcal{O}$ with $(E, p, S_i, N_2, K, \rho)$:
   If $\mathcal{O}$ outputs $\perp$, then return $\perp$.
   Else, obtain ϕ_C from $\mathcal{O}$ (satisfying the conditions 1 to 4 from Oracle 5.9).
5. 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 ρ and call $\mathcal{O}$.
   Let ρ be the last value for which $\mathcal{O}$ did not output $\perp$.
   Let (K_0, K_1, K_2) be the output of $\mathcal{O}$ called with $(E, p, S_i, N_2, K, \rho)$.
6. Let $X = (S_i \cap \mathsf{Eig}_{N_1}(\phi_C))$ and $Y = (S_i \setminus \mathsf{Eig}_{N_1}(\phi_C))$.

7. Use the Algorithm 5.8 with input E, p, N_1, N_2, $\{P_A, Q_A\}$, $E_A, \phi_A|_{E[N_2]}$, (K_0, K_1, K_2), and k to determine whether $R_A \in X$ or $R_A \in Y$.
8. If $R_A \in X$, then let $S_{i+1} = X$, otherwise if $R_A \in Y$, then let $S_{i+1} = Y$.
9. Increment i and repeat Steps 2 to 8 until $|S_i| \leq f(\lambda)$.
10. For each point $R \in S_i$, compute the isogeny with kernel $\langle R \rangle$, and return R if the image curve is isomorphic to E_A.

We now analyze Algorithm 5.12, thereby proving Theorem 5.11.

Proof. (Theorem 5.11) The proof will consist in analyzing the success probability and runtime of Algorithm 5.12. In particular, we will now show that in the setting of Theorem 5.11, Algorithm 5.12 runs in time $K^3 \operatorname{poly}(\lambda)$.

Let σ_1 be the fraction of sets $S \subseteq E[N_1]$ for which there exists a non-negligible amount of $K \leq N_1$ and $\rho \leq 1/2$ 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 8 instead of outputting $\perp$.

Note that at the end of Step 7, $|S_{i+1}| \leq (1-\rho)|S_i|$ for all i. Let $C = \operatorname{poly}(\lambda)$ and $m = \left\lceil \frac{\log N_1 - \log C}{-\log(1-\rho)} \right\rceil$. Then

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

This implies that to ensure $|S_m| \leq (1-\rho)^m N_1$ has polynomial size, $O(m)$ calls to $\mathcal{O}$ are required. Therefore, we expect there to be at least $O(\log(\lambda))$ many iterations of Steps 2 to 8.

Step 5 performs two binary searches using $\mathcal{O}$. The search for the minimum K takes $\log N_1$ calls, and the search for the maximum ρ takes $\log(\operatorname{poly}(\lambda))$ calls, for a total of $\operatorname{poly}(\lambda)$ calls. By the statement of Theorem 5.7, Step 7 will terminate with high probability, say σ_2, in worst case time $\tilde{O}(K^3)$. Therefore, since Steps 2 to 8 happens $O(\log \lambda)$ many times, Algorithm 5.12 terminates in worst-case time $K^3 \operatorname{poly}(\lambda) \log(\lambda)$, and succeeds with probability $(\sigma_1 \sigma_2)^{O(\log(\lambda))} = 1 - \operatorname{negl}(\lambda)$. □

We will further discuss the relationship between ρ and K in Section 9.2.

6 Quadratic Forms and Endomorphism Rings

If we know the structure of $\operatorname{End}(E)$, then we can efficiently reduce Oracle 5.9 to an oracle which finds solutions to a particular multivariate quadratic equation that satisfy certain algebraic conditions. In particular, once the endomorphism is described in terms of a basis, then finding an endomorphism of a particular degree amounts to finding a solution to a particular multivariate quadratic equation (see Section 6.1). Similarly, if we choose a basis for both $\operatorname{End}(E)$ and $E[N_1]$, then the eigenvector conditions of the endomorphism amount to finding the eigenspace of a particular matrix (see Section 6.2).

6.1 Quadratic Forms from Degrees of Endomorphisms

So far we have been describing endomorphisms by giving their kernels. Since an endomorphism is well-defined up to isomorphism by its kernel, knowing its kernel, or even better a triangular kernel, makes it is easy to calculate the endomorphism using Vélu's formulas.

It is well known that the endomorphism ring of an elliptic curve E has the structure of a 4-dimensional $\mathbb{Z}$-module. In other words, there exist endomorphisms b_1, b_2, b_3, b_4 of E such that

$$\{[w] \cdot b_1 + [x] \cdot b_2 + [y] \cdot b_3 + [z] \cdot b_4 \mid w, x, y, z \in \mathbb{Z}\}$$

describes the set of endomorphisms in E. We use the phrase *knowing the endomorphism ring of an elliptic curve*, to mean that we know an explicit basis $\{b_1, b_2, b_3, b_4\}$ of $\mathrm{End}(E)$.

One advantage of a basis representation is that there is a simple formula for computing the degree of a general endomorphism in terms of the respective traces and degrees of the endomorphisms in the basis. Another advantage, which we will see in Section 6.2, is that the description of an endomorphism in terms of a well-known basis makes it easy to explicitly find the eigenspace of that endomorphism.

In Proposition 6.3 we will show how to turn a description of an endomorphism in terms of a basis of $\mathrm{End}(E)$ into the triangular kernel description, so that the results of Section 5 can be utilized. We will do this using the following lemma, which shows how to find the action of an endomorphism on a torsion subgroup from its basis coefficients.

Lemma 6.1. *Suppose* $\mathrm{End}(E) = \langle b_1, b_2, b_3, b_4 \rangle$ *and* N *is a natural number. For integer variables* (w, x, y, z)*, the action of any endomorphism*

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

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 the best case* $\Omega(\log^2 p)$ *time (when* N *is* $\log p$*-smooth).*

Proof. We prove Lemma 6.1 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 6.2.
Input: E, a basis $\{b_1, b_2, b_3, b_4\}$ of $\mathrm{End}(E)$, integer variables (w, x, y, z), N, 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 $i = 1, \ldots, 4$. Solving the discrete logarithm for these values, in terms of P and Q, gives $b_i|_{E[N]}$ 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. Output P, Q, M.

The most difficult part of Algorithm 6.2 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 Theorem 4.9. 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 Theorem 4.9).

Step 2 has runtime $O\Big(\sum_{1\leq i\leq r} e_i(\log N + \sqrt{q_i})\Big)$ [19], which is always less than the runtime in Step 1. □

Now, we will use Lemma 6.1 in Proposition 6.3 to transform endomorphisms (in terms of their basis) into a triangular kernel. This will allow us to apply Theorem 5.11 with the basis representation.

More specifically, in Proposition 6.3 we will find a triangular kernel (K_0, K_1, K_2), where $\phi_C = \widehat{\phi}_{C,0} \circ \phi_{C,2} \circ \phi_{C,1}$. To do this we first fix the orders of K_0, K_1, K_2 to be numbers that satisfy the conditions of a triangular kernel. Additionally, $\gcd(|K_2|, N_1) = 1$. We can use Lemma 6.1 and the following facts to find (K_0, K_1, K_2):

- $\ker \phi_{C,0} = \ker \widehat{M}$, where $\widehat{M}$ describes the action of $\widehat{\phi}_C$ on $E[|K_0|]$,
- $\ker \phi_{C,1} = \ker M$, where M describes the action of ϕ_C on $E[|K_1|]$, and
- $\ker \phi_{C,2} = \phi_{C,1}(\ker M_k)$, where M_k describes the action of ϕ_C on $E[|K_1| \cdot |K_2|]$.

Proposition 6.3. *Suppose we are given*

1. *a starting supersingular elliptic curve $E(\mathbb{F}_{p^2})$ such that $p = N_1 N_2 - 1$ for coprime, positive integers N_1 and N_2, such that N_2 is $\log p$-smooth,*
2. *a basis $\{b_1, b_2, b_3, b_4\}$ of $\mathrm{End}(E)$, and*
3. *a cyclic Lk-degree endomorphism*
$$\phi_C = [w_0] \cdot b_1 + [x_0] \cdot b_2 + [y_0] \cdot b_3 + [z_0] \cdot b_4$$
of E, where $L \mid (N_2)^2$.

Then there exists an algorithm to find points generating the triangular kernel of ϕ_C with respect to N_2 whose worst case runtime is $\tilde{O}(k^3)$. Further if k is $\log p$-smooth, then the runtime is $\tilde{O}(\sqrt{k})$.

Proof. We prove Proposition 6.3 by describing the necessary steps in the algorithm and analyzing their runtime.

1. Write $\deg \phi_C = Lk$ with L maximal such that $L \mid (N_2)^2$.
2. Let $L_0 = \min(L, N_2)$. We will find a triangular kernel (K_0, K_1, K_2), where $|K_1| = L_0, |K_0| = L/L_0$, and $|K_2| = k$.
3. First we solve for K_0. Fix a basis $\{P_0, Q_0\} = \left\{\left[\frac{N_2}{|K_0|}\right] P_B, \left[\frac{N_2}{|K_0|}\right] Q_B\right\}$ of $E[|K_0|]$.

4. We will use the fact that $\ker \phi_{C,0} = \ker \widehat{M}$, where $\widehat{M}$ describes the action of $\widehat{\phi}$ on $E[|K_0|]$. That is, run Algorithm 6.2 with input $\{\widehat{b_1}, \ldots, \widehat{b_4}\}$, integer variables (w, x, y, z), $|K_0|$ and the basis $\{P_0, Q_0\}$ for $E[|K_0|]$. Let the output be the matrix $\widehat{M}$.
5. Find some vector $\left[\begin{smallmatrix}\alpha_0\\ \beta_0\end{smallmatrix}\right]$ which generates the kernel of $\widehat{M}(w_0, x_0, y_0, z_0)$. Set $K_0 = [\alpha_0] \cdot P_0 + [\beta_0] \cdot Q_0$.
6. We now find the kernel point K_1 in a similar process. Fix a basis $\{P_1, Q_1\} = \left\{\left[\frac{N_2}{|K_1|}\right] P_B, \left[\frac{N_2}{|K_1|}\right] Q_B\right\}$ of $E[|K_1|]$.
7. We will use the fact that $\ker \phi_{C,1} = \ker M$, where M describes the action of ϕ_C on $E[|K_1|]$. That is, run Algorithm 6.2 with input $\{b_1, \ldots, b_4\}$, integer variables (w, x, y, z), L_0 and basis $\{P_1, Q_1\}$ for $E[|K_1|]$. Let the output be the matrix M.
8. Find some vector $\left[\begin{smallmatrix}\alpha_1\\ \beta_1\end{smallmatrix}\right]$ which generates the kernel of $M(w_0, x_0, y_0, z_0)$. Set $K_1 = [\alpha_1] \cdot P_1 + [\beta_1] \cdot Q_1$.
9. Lastly we describe how to find the kernel point K_2. We perform a similar process to the last two kernel points, except we need to push the point through an isogeny (see Definition 5.1).
10. Run Algorithm 6.2 with $\{b_1, \ldots, b_4\}$, integer variables (w, x, y, z), and k. Let the output be the matrix M_k and the basis $\{P_k, Q_k\}$ for $E[k] \subset E(\overline{\mathbb{F}}_{p^2})$.
11. Find some vector $\left[\begin{smallmatrix}\alpha_k\\ \beta_k\end{smallmatrix}\right]$ which generates the kernel of $M_k(w_0, x_0, y_0, z_0)$.
12. Let $\phi_{C,1}$ be the isogeny from E with kernel $\langle K_1 \rangle$, and set $K_2 = \phi_1([\alpha_k] \cdot P_k + [\beta_k] \cdot Q_k)$.
13. Return K_0, K_1, K_2.

Steps 4 and 7 run in time $O(\log p)$, since N_2 is $\log p$-smooth and $E[N_2] \subset E(\mathbb{F}_{p^2})$. By Lemma 6.1, Step 10 will run in worst case time $\tilde{O}(k^3)$ and best case time $\tilde{O}(\sqrt{k})$ when k is $\log p$-smooth (assuming k does not divide N_2, in which case it is even better). □

Remark 6.4. The converse of Proposition 6.3 is true as well, in the sense that there exists an algorithm which outputs coefficients (w, x, y, z) upon input of a triangular kernel for ϕ_C with respect to N_2, and it has the same runtime. We do not state this converse algorithm, as will we not use it.

Proposition 6.3 will allow us to convert Oracle 5.9 (which returns an endomorphism in terms of a triangular kernel) to an oracle that returns the coefficients of an endomorphism in terms of a basis. The advantage of this becomes apparent when we look at the explicit description of the degree and eigenspaces of an endomorphism represented in terms of a basis.

More specifically, associated to any basis of the endomorphism ring is a 4-variable quadratic form which represents the degrees of the endomorphisms.

Lemma 6.5. *If $\phi_C = [w] \cdot b_1 + [x] \cdot b_2 + [y] \cdot b_3 + [z] \cdot b_4$ is any endomorphism given in terms of a basis $\{b_1, b_2, b_3, b_4\}$ of the endomorphism ring, then the degree*

of ϕ_C is given by the following quadratic form:

$$\begin{aligned} q(w,x,y,z) = \phi_C \circ \widehat{\phi}_C = [&w^2 \deg b_1 + x^2 \deg b_2 + y^2 \deg b_3 + z^2 \deg b_4 \\ &+ wx \operatorname{Tr}(b_1\widehat{b}_2) + wy \operatorname{Tr}(b_1\widehat{b}_3) + wz \operatorname{Tr}(b_1\widehat{b}_4) + xy \operatorname{Tr}(b_2\widehat{b}_3) \\ &+ xz \operatorname{Tr}(b_2\widehat{b}_4) + yz \operatorname{Tr}(b_3\widehat{b}_4)]. \end{aligned}$$

From Proposition 6.3 and Lemma 6.5 we see that an oracle which outputs solutions to certain quadratic forms may be used instead of an oracle that outputs triangular kernels. This new oracle will be presented at the end of Section 6.2 (see Oracle 6.11). Next we explore what the eigenspace conditions of Oracle 5.9 become if we use this basis description of endomorphisms.

6.2 Quadratic Forms from Eigenspaces of Endomorphisms

In the last subsection we showed that, if $\mathrm{End}(E)$ is known, then instead of representing endomorphisms by triangular kernels, we can represent them in terms of a basis of $\mathrm{End}(E)$. We saw that this basis representation allows for a simple description of the degree and the action of the endomorphism on the set of Alice's possible private points.

In this subsection, we explore when the eigenspace requirements (Conditions 3 and 4) of Oracle 5.9 are satisfied, assuming the endomorphism is described in terms of a basis. We do this by first analyzing the eigenspace of a random matrix on $E[N_1]$. We simplify our calculations by assuming that N_1 is a prime power, although a similar analysis should work if N_1 is any $\log p$-smooth number.

The following notation will be useful in the main theorem of this subsection.

Notation 6.6. Let ℓ and e be fixed positive integers. Normally, in SIDH and in SIKE, $\ell^e = 2^a$ or $\ell^e = 3^b$. Let $\left[\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right]$ denote a matrix with entries in $\mathbb{Z}/\ell^e\mathbb{Z}$. Given an eigenvector $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ of this matrix and a fixed small prime ℓ, we use the following notation in the remainder of this document:

- ν denotes the largest natural number such that $\ell^\nu \mid \beta$ and $\ell^\nu \mid \delta - \alpha$.
- β', ϵ' are the numbers such that $\beta = \ell^\nu \beta'$ and $\delta - \alpha = \ell^\nu \epsilon'$.
- ξ is the largest natural number such that $\ell^\xi \mid \epsilon' - 2\beta' r$ and $\xi \leq \frac{e-\nu}{2}$.
- $\zeta = e - \nu - \xi$.

Remark 6.7. In Theorem 6.8 we will describe the ℓ^e-eigenspace of a matrix M. We will see in the proof of Theorem 6.8, that the definitions of ξ and ζ are independent of the choice of eigenvector $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ of M. In other words, ξ and ζ are defined with respect to M and prime ℓ.

Theorem 6.8 shows that if there is an eigenvector of a matrix and the associated ζ is small, then there are many eigenvectors.

Theorem 6.8. *Suppose there is an eigenvector $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ of a matrix $\left[\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right]$ in $E[\ell^e]$.*

- *If $\ell \mid \beta'$, then $\left[\begin{smallmatrix}1\\ \kappa\end{smallmatrix}\right]$ is an eigenvector if and only if it has the form $\left[\begin{smallmatrix}1\\ r+c\ell^{\zeta}\end{smallmatrix}\right]$ for some $c \in \mathbb{Z}$.*
- *If $\ell \nmid \beta'$, then $\left[\begin{smallmatrix}1\\ \kappa\end{smallmatrix}\right]$ is an eigenvector if and only if it has the form $\left[\begin{smallmatrix}1\\ r+c\ell^{\zeta}\end{smallmatrix}\right]$ or $\left[\begin{smallmatrix}1\\ -r+\epsilon'(\beta')^{-1}+c\ell^{\zeta}\end{smallmatrix}\right]$ for some $c \in \mathbb{Z}$.*

Proof. Note that $\left[\begin{smallmatrix}1\\ r\end{smallmatrix}\right]$ is an eigenvector of the matrix $\left[\begin{smallmatrix}\alpha & \beta\\ \gamma & \delta\end{smallmatrix}\right]$ over $\mathbb{Z}/\ell^e\mathbb{Z}$ if and only if $\gamma + (\delta - \alpha)r - \beta r^2 \equiv 0 \pmod{\ell^e}$, see Lemma A.1. Now, suppose $\left[\begin{smallmatrix}1\\ r+x\end{smallmatrix}\right]$ is also an eigenvector. This implies

$$\gamma + (\delta - \alpha)r - \beta r^2 \equiv 0 \pmod{\ell^e}$$

and

$$\gamma + (\delta - \alpha)(r + x) - \beta(r + x)^2 \equiv 0 \pmod{\ell^e}.$$

Subtracting these two equations shows that $\left[\begin{smallmatrix}1\\ r+x\end{smallmatrix}\right]$ is an eigenvector if and only if x satisfies:

$$(\delta - \alpha)x - \beta(2rx + x^2) \equiv 0 \pmod{\ell^e}.$$

This is equivalent to

$$(\delta - \alpha - \beta(2r + x))x \equiv 0 \pmod{\ell^e}. \tag{3}$$

and also,

$$(\epsilon' - \beta'(2r + x))x \equiv 0 \pmod{\ell^{e-\nu}}. \tag{4}$$

Suppose $\ell \mid \beta'$. Then $\ell \nmid \epsilon'$, and hence a vector of the form $\left[\begin{smallmatrix}1\\ r+x\end{smallmatrix}\right]$ is an eigenvector if and only if it has the form $x \equiv 0 \pmod{\ell^{e-v}}$. This is equivalent to saying $\left[\begin{smallmatrix}1\\ r+x\end{smallmatrix}\right]$ has the form $\left[\begin{smallmatrix}1\\ r+c\ell^{\zeta}\end{smallmatrix}\right]$, since $\xi = 0$.

Suppose $\ell \nmid \beta'$. Then Equation (4) is equivalent to

$$(\epsilon'(\beta')^{-1} - (2r + x))x \equiv 0 \pmod{\ell^{e-\nu}}. \tag{5}$$

Further suppose x is a solution to Equation (5). One of the following two cases holds:

$$x \equiv -2r + \epsilon'(\beta')^{-1} \pmod{\ell^{\zeta}} \text{ and}$$
$$x \equiv 0 \pmod{\ell^{\xi}},$$

or

$$x \equiv 0 \pmod{\ell^{\zeta}} \text{ and}$$
$$x \equiv -2r + \epsilon'(\beta')^{-1} \pmod{\ell^{\xi}}.$$

Thus the eigenvector $\left[\begin{smallmatrix}1\\ r+x\end{smallmatrix}\right]$ must have the form given in the theorem.

Conversely, suppose that $x = c\ell^{\zeta}$. By the definition of ξ, we have that $x \equiv 0 \equiv -2r + \epsilon'\beta'^{-1} \pmod{\ell^{\zeta}}$. Thus x is a solution to Equation (5).

Now suppose that $x = -2r + \epsilon'\beta'^{-1} + c\ell^{\zeta}$, then $x \equiv 0 \pmod{\ell^{\xi}}$. Thus x is a solution to Equation (5). □

Remark 6.9. Theorem 6.8 proves that for a matrix

1. the number of ℓ^e-eigenvectors is $\ell^{e-\zeta} = \ell^{\nu+\xi}$ or $2\ell^{e-\zeta} = 2\ell^{\nu+\xi}$, and

2. the probability that a random point of order ℓ^e is an eigenvector is $\ell^{-\zeta}$ or $\frac{1}{2}\ell^{-\zeta}$ (depending on if $\ell \mid \beta'$).

Given a basis $\{b_0, b_1, b_2, b_3\}$ for an endomorphism ring, Lemma 6.1 shows that the action of the set of endomorphisms

$$\{[w] \cdot b_1 + [x] \cdot b_2 + [y] \cdot b_3 + [z] \cdot b_4 \mid w, x, y, z \in \mathbb{Z}\}$$

on $E[\ell^e]$ can be given a 2×2-matrix $M(w, x, y, z)$ whose entries are linear in the four variables. Thus, we are interested in the values (w_0, x_0, y_0, z_0) which the matrix $M(w_0, x_0, y_0, z_0)$ has many eigenvectors. In other words, we want $M(w_0, x_0, y_0, z_0)$ to have at least one eigenvector and ζ to be small. We will now replace Conditions 3 and 4 of Oracle 5.9 with more concrete conditions. This allows us to restate Theorem 5.11 with an oracle that has a more concrete output.

Remark 6.10. In order to use of Theorem 6.8 to derive the following oracle/reduction, we require $N_1 = \ell^e$. For a more general statement simply replace Conditions 4 and 5 in Oracle 6.11 with a condition that eigenspace of ϕ_C separates S (the space that potentially contains Alice's private point) into two large subsets (as in Oracle 5.9).

Oracle 6.11.
Input: E, p, N_2, a set $S \subseteq E[\ell^e]$, a quadratic form $q(w, x, y, z)$, a 2×2-matrix $M(w, x, y, z)$ acting on $E[\ell^e]$ whose entries are linear in the four variables, an integer $K \leq \ell^e$, and $s \in [0, e]$.
Output: Integers (w_0, x_0, y_0, z_0), with no common divisor, satisfying the equation $q(w_0, x_0, y_0, z_0) = Lk$, such that the following constraints hold:

1. $k \leq K$,
2. $L \mid (N_2)^2$,
3. $\gcd(k, \ell) = 1$,
4. $M(w_0, x_0, y_0, z_0)$ has at least one eigenvector, and
5. $M(w_0, x_0, y_0, z_0)$ has $\zeta < s$, (where ζ is defined with respect to M and ℓ),

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

The next theorem is analogous to Theorem 5.11 and constitutes the second main result of this work. It states that endomorphisms associated to the outputs of Oracle 6.11 can be used to gain information about Alice's private key r_A.

Theorem 6.12. *Suppose we are given*

1. *a starting supersingular elliptic curve $E(\mathbb{F}_{p^2})$ such that $p = N_1 N_2 - 1$ for N_1 coprime to N_2, where $N_1 = \ell^e$ and N_2 is $\log p$-smooth,*
2. *bases $\{P_A, Q_A\}$ of $E[N_1]$ and $\{P_B, Q_B\}$ of $E[N_2]$,*
3. *a basis $\{b_1, b_2, b_3, b_4\}$ of $\mathrm{End}(E)$,*
4. *the image of an N_1-degree isogeny $E_A = \phi_A(E)$,*
5. *the action $\phi_A|_{E[N_2]}$ with respect to $\{P_B, Q_B\}$, and*

6. *access to Oracle 6.11 (denoted $\mathcal{O}$), and for an overwhelming fraction of sets S, the oracle $\mathcal{O}$ will succeed for a non-negligible fraction of $K \leq \ell^e$, and $s \in [0, e]$ (the input to $\mathcal{O}$).*

Then there exists a (classical) algorithm which outputs r_A, where $\ker \phi_A = \langle R_A \rangle$, *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 $\tilde{O}(K^3 \cdot m)$. Further, if the k's output from $\mathcal{O}$ are all* $\log p$*-smooth, then the runtime is $\tilde{O}(\sqrt{K} \cdot m)$.*

Theorem 6.12 may initially look more complicated than the main theorem, Theorem 5.11, because there are more conditions. However, the result is actually simpler because, inputting bases allows us to reduce the oracle to linear algebra and solving a quadratic form.

Proof. To prove this theorem, we can appeal to Theorem 5.11. We do so by describing the procedure to turn Oracle 6.11 into Oracle 5.9, and showing that the steps required can be performed efficiently.

Algorithm 6.13.
Input: The input to Oracle 6.11, $(E, p,$ $\log p$-smooth N_2, bases $\{P_A, Q_A\}$ and $\{P_B, Q_B\}$ a set $S \subset E[\ell^e]$, an integer $K \leq \ell^e$, $\rho \in (0, 1/2]$ where $\rho \in \left[\frac{1}{f(\lambda)}, \frac{1}{2}\right]$ for a polynomial f), and access to $\mathcal{O}$.
Output: The output to Oracle 5.9 ($\ker_\triangle \phi_C$ of a cyclic endomorphism ϕ_C subject to the same constraints as Oracle 5.9, or $\perp$ if no such endomorphism ϕ_C exists).

Notice that both the oracles output $\perp$ at the same time, by our definition of the output.

1. Use Algorithm 6.2 with input $E, \{b_1, \ldots, b_4\}$, integers variables (w, x, y, z), ℓ^e, and the basis $\{P_A, Q_A\}$ for $E[\ell^e]$. Let the output be the matrix $M(w, x, y, z)$.
2. Compute the quadratic form given in Lemma 6.5 associated to the basis $\{b_1, b_2, b_3, b_4\}$.
3. Set $s = \min\left\{e, \max\left\{\log_\ell\left(\frac{1}{\rho}\right), \log_\ell\left(\frac{1}{1-\rho}\right)\right\}\right\}$. If Oracle 6.11 instantiated at
$$\mathcal{O}(E, p, N_2, S, q(w, x, y, z), M(w, x, y, z), K, s)$$
succeeds, then label the output as (w_0, x_0, y_0, z_0). Otherwise output $\perp$.
4. Use Proposition 6.3 with input $E, \{b_1, b_2, b_3, b_4\}$, basis $\{P_B, Q_B\}$ of $E[N_2]$, and (w_0, x_0, y_0, z_0) to get a triangular kernel $\ker_\triangle \phi_C$ of
$$\phi_C = [w_0] \cdot b_1 + [x_0] \cdot b_2 + [y_0] \cdot b_3 + [z_0] \cdot b_4.$$
5. Output $\ker_\triangle \phi_C$.

Now that we have an algorithm to convert Oracle 6.11 into Oracle 5.9, we will convert Algorithm 5.12 for Theorem 5.11 to an algorithm for Theorem 6.12. Specifically, each call to Oracle 5.9 by Algorithm 5.12\Theorem 5.11 can be replaced with Algorithm 6.13 (which calls Oracle 6.11 as a subroutine).

Algorithm 6.13 runs in worst case time $\tilde{O}(K^3)$ and best case time $\tilde{O}(\sqrt{K})$ by Proposition 6.3. We now give an argument to show that this choice of s will

give a solution (w_0, x_0, y_0, z_0) so that $M(w_0, x_0, y_0, z_0)$ roughly satisfies the the size conditions of Oracle 5.9 (Steps 3 and 4). To simplify our exposition we will consider the situation where the ratio of the eigenspace to search space is $\ell^{e-\zeta}$ to ℓ^e. Observe that the condition on ρ in Oracle 5.9, combined with the formula for the number of eigenvectors from Remark 6.9 leads to the following:

$$\rho, 1-\rho < \frac{\ell^{e-\zeta}}{\ell^e} = \ell^{-\zeta}.$$

Rearranging, we see that if $s = \max\left\{\log_\ell\left(\frac{1}{\rho}\right), \log_\ell\left(\frac{1}{1-\rho}\right)\right\}$, then $s > \zeta$. Note that this does not guarantee the conditions on $|S|$ from Oracle 5.9, only that the size is correct. □

Notation 6.14. In later sections we say that s is *close* to e to mean that

$$s \geq \min\left\{e, \max\left\{\log_\ell\left(\frac{1}{\rho}\right), \log_\ell\left(\frac{1}{1-\rho}\right)\right\}\right\},$$

for an implicit ρ given in terms of a polynomial f.

Instead of using the abstract language of endomorphisms given in Theorem 5.11, Theorem 6.12 is framed in terms of Oracle 6.11, which finds solutions to a particular set of 4-variable linear and quadratic equations. This new language makes the requirements on Oracle 6.11 more accessible.

7 Instantiating the Oracle for $j = 1728$

The goal of this section will be to instantiate Oracle 6.11 at an elliptic curve of particular interest, namely the elliptic curve with j-invariant 1728, whose endomorphism ring is known. This was the starting curve for the Round 1 SIKE submission to the NIST post-quantum standardization process [6]. As this elliptic curve is closely related to the Round 2 starting curve [2], which is presently a candidate for standardization, our analysis is still relevant.

To instantiate the oracle, we need to investigate a quadratic form that describes the degree of endomorphisms on this curve (Section 7.1), and discuss the number of eigenvectors of an endomorphism of this curve (Section 7.2). Finally, we will show that our methods do not give a practical attack on Bob's private key in SIKE (Section 7.3). The choice to look at Bob's key is done because his torsion subgroup has a particularly convenient basis, but similar analysis could be done for Alice's key.

7.1 The Quadratic Form for $j = 1728$

We fix the starting curve to be the above mentioned SIKE curve. We do this because it is the NIST Round 1 starting curve. As the Round 2 starting curve [2] is adjacent on the 2-isogeny graph to this curve, the endomorphism rings of these two starting curves are almost the same, so our analysis is relevant. We chose the Round 1 elliptic curve due to the simplicity of the description of its endomorphism ring.

Notation 7.1. For the next two sections, we let E_0 denote the elliptic curve having j-invariant 1728 over $\mathbb{F}_p$, where $p = 2^a 3^b - 1$ for some integers $a \geq 2$ and $b \geq 1$.

The endomorphism ring of $\mathrm{End}(E_0)$ is an excellent candidate for applying the reduction in Theorem 5.11 as it has many endomorphisms of small degree. $\mathrm{End}(E_0)$ is known, and can be described in terms of the following two functions.

Notation 7.2. Let $\pi : E_0 \to E_0$ denote the *Frobenius map* $\pi(x, y) = (x^p, y^p)$, and $\iota : E_0 \to E_0$ denote the *distortion map* $\iota(x, y) = (-x, iy)$.

Lemma 7.3. *The endomorphism ring of E_0 is*

$$\mathrm{End}(E_0) \cong \mathbb{Z}[1] \oplus \mathbb{Z}\iota \oplus \mathbb{Z}\tfrac{[1]+\pi}{2} \oplus \mathbb{Z}\tfrac{\iota+\iota\pi}{2}.$$

Proof. Since $p \equiv 3 \pmod 4$, we can apply Proposition 4.2 of [18] where $J = \pi$ and $I = \iota$ to find:

$$\mathrm{End}(E_0) \cong \mathbb{Z}\tfrac{[1]+\pi}{2} \oplus \mathbb{Z}\tfrac{\iota+\iota\pi}{2} \oplus \mathbb{Z}\pi \oplus \mathbb{Z}\iota\pi.$$

The statement follows by applying a change of basis. □

To find endomorphisms of a particular degree, we look for solutions to the quadratic form associated to the endomorphism's degree (as in Lemma 6.5). By direct substitution in Lemma 6.5, we get the following.

Lemma 7.4. *Suppose $\phi_C = [w] \cdot [1] + [x] \cdot \iota + [y] \cdot \frac{[1]+\pi}{2} + [z] \cdot \frac{\iota+\iota\pi}{2}$ for some integers (w, x, y, z). Then the degree of ϕ_C is given by the following quadratic form*

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

The proof of Lemma 7.4 is straightforward and given in Appendix A.3.

We now show how to find a slightly nicer form for q. A straightforward computation gives the following result in the case where N_2 is a power of 2, although there is a similar result for more general N_2.

Lemma 7.5. *If a 4-tuple (w_0, x_0, y_0, z_0) is a solution to*

$$w^2 + x^2 + \left(\tfrac{p+1}{4}\right)(y^2 + z^2) + wy + xz = 2^m k, \tag{6}$$

then the 4-tuple $(2w_0 + y_0, 2x_0 + z_0, y_0, z_0)$ is a solution to

$$w^2 + x^2 + p(y^2 + z^2) = 2^{m+2} k. \tag{7}$$

Conversely, if a 4-tuple (w_0, x_0, y_0, z_0) is a solution to Equation (7), then the 4-tuple $(w_0 - y_0, x_0 - z_0, 2y_0, 2z_0)$ is a solution to Equation (6).

From now on, we will focus on the quadratic form given in the left-hand side of Equation (7) in Lemma 7.5.

Notation 7.6. Let $q_0(w, x, y, z) = w^2 + x^2 + p(y^2 + z^2)$. This is the degree of the quadratic form associated to a subring $\mathcal{R} = \langle 1, \iota, \pi, \iota\pi \rangle \subset \mathrm{End}(E_0)$.

To find endomorphisms of the degree required by Oracle 6.11, it suffices to find solutions to

$$q_0(w_0, x_0, y_0, z_0) = 2^{m+2}k.$$

We will often drop the $+2$ for ease of description.

Remark 7.7. It is not hard to find a solution to this equation if we ignore the constraint that the resulting endomorphism should have many eigenvectors. This can be done by first fixing a large $m \leq 2a$ and a small k. Randomly choose y and z of different parity until x and w can be chosen via Cornacchia's algorithm [8]. This is demonstrated in Example A.7.

The next subsection gives conditions on endomorphisms on $E_0[\ell^e]$ having many eigenvectors.

7.2 The Main Reduction for $j = 1728$

In this subsection, we reverse the torsion subgroups of Alice and Bob, $E[N_1]$ and $E[N_2]$, compared to the SIKE torsion subgroups for ease of description. More specifically, we instantiate Oracle 6.11 at E_0 using the quadratic form $q_0(w, x, y, z)$, where $N_1 = 3^{239}$ and $N_2 = 2^{372}$. Then Theorem 6.12 (when it calls Oracle 6.11 with $N_1 = 3^{239}$ and $N_2 = 2^{372}$) can be thought of as computing Bob's private key in SIKE.

We proceed by describing the action of ϕ_C on $E_0[N_1]$ in terms of a particular basis of $E_0[3^{239}]$. The basis of $E_0[\ell^e]$ is chosen so that the distortion map ι and the Frobenius map act nicely with respect to this basis.

Definition 7.8. A *convenient basis* of $E_0[\ell^e]$ is a basis $\{P, Q\}$ where

$$Q = \iota(P) = \pi(P).$$

Remark 7.9. A convenient basis may not always exist (see Lemma A.5). For example, no such basis exists for $E_0[2^{372}]$, although other simplified bases exist for $E_0[2^{372}]$. For $E_0[3^{239}]$ a convenient basis does exist, see Example A.6. For simplicity, in the next two subsections, we consider the situation where Bob's basis, $\{P_B, Q_B\}$, is a convenient basis (i.e., where $N_1 = 3^{239}$).

The following lemma shows that if a convenient basis does exist, then there is a simple matrix description of ϕ_C on $E_0[\ell^e]$.

Lemma 7.10. *If there exists a convenient basis $\{P, \pi(P)\} = \{P, \iota P\}$ of $E_0[\ell^e]$, then the endomorphism of $\mathcal{R}$*

$$\phi_C = [w] \cdot [1] + [x] \cdot \iota + [y] \cdot \pi + [z] \cdot \iota\pi$$

acts as the matrix

$$M(w, x, y, z) = \left[\begin{smallmatrix} w-z & -x+y \\ x+y & w+z \end{smallmatrix}\right]$$

on $E_0[\ell^e]$ with respect to this basis.

Proof. As $\iota^2(P) = [-1] \cdot P$ and $\pi^2(P) = P$, the result follows. □

Notation 7.11. Let $M_0(w,x,y,z) = \left[\begin{smallmatrix} w-z & -x+y \\ x+y & w+z \end{smallmatrix}\right]$.

We now instantiate the generic Oracle 6.11 into a new oracle acting on the SIKE parameters. Inputting the matrix M_0 and q_0 into Oracle 6.11, gives the following oracle.

Oracle 7.12.
Input: E_0 with $j(E_0) = 1728$, p, N_2, a set $S \subseteq E_0[N_1]$, an integer $K \leq N_1 = 3^{239}$, and $s \in [0, 239]$.
Output: Integers (w_0, x_0, y_0, z_0) with no common divisors, satisfying $w_0^2 + x_0^2 + p(y_0^2 + z_0^2) = Lk$ such that:

1. $k \leq K$,
2. $L \mid (2^{372})^2$,
3. $\gcd(k, 3) = 1$,
4. $\left[\begin{smallmatrix} w_0-z_0 & -x_0+y_0 \\ x_0+y_0 & w_0+z_0 \end{smallmatrix}\right]$ has at least one eigenvector, and
5. $\left[\begin{smallmatrix} w_0-z_0 & -x_0+y_0 \\ x_0+y_0 & w_0+z_0 \end{smallmatrix}\right]$ has $\zeta < s$, (where ζ is defined with respect to M and $\ell = 3$),

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

The following proposition shows that the security assumption of SIKE/SIDH relies on the hardness of constructing Oracle 7.12 for K smaller than sub-exponential in the security parameter. More specifically, there should be very few endomorphisms

$$\phi_C = [w_0] \cdot 1 + [x_0] \cdot \iota + [y_0] \cdot \pi + [z_0] \cdot \iota\pi \in \mathcal{R}$$

which satisfy Conditions 1 to 5 of Oracle 7.12 (with $K = \text{poly}(\log p)$), because each such endomorphism has the potential to reveal information about Bob's or Alice's private key.

Proposition 7.13. *Oracle 7.12 is an instantiation of Oracle 6.11 at E_0, q_0, and M_0.*

Proof. This result follows from making following substitutions into Oracle 6.11. Note that Oracle 6.11 does not require q or M to be associated to $\text{End}(E_0)$, so it will accept q_0 and M_0 of $\mathcal{R}$.

1. the elliptic curve $E = E_0$,
2. $N_1 = 3^{239}$,
3. $N_2 = 2^{372}$,
4. the quadratic form $q(w,x,y,z) = w^2 + x^2 + \left(\frac{p+1}{4}\right)(y^2 + z^2) + wy + xz$,
5. the matrix $M_0(w,x,y,z) = \left[\begin{smallmatrix} w-z & -x+y \\ x+y & w+z \end{smallmatrix}\right]$, (see Lemma 7.10), and
6. set $\rho = \min\left\{\frac{1}{3^\zeta}, 1 - \frac{1}{3^\zeta}\right\}$ (where ζ is defined with respect to M and $\ell = 3$), and $s = \min\left\{239, \max\left\{\log_3\left(\frac{1}{\rho}\right), \log_3\left(\frac{1}{1-\rho}\right)\right\}\right\}$. □

Remark 7.14. Using this more concrete Oracle 7.12, Theorem 6.12 can now target Bob's private key in the NIST Round 1 SIKE starting curve with j-invariant 1728. Note that we use q_0 and M_0 for $\mathcal{R}$, instead of the entirety of $\text{End}(E_0)$ as in Steps 1 and 2 of Algorithm 6.13, but this is equivalent by Lemma 7.10 assuming there is a convenient basis.

7.3 Characterizing Large Eigenspaces for $j = 1728$

The condition for an endomorphism having a large number of eigenvectors splits naturally into two conditions. We investigate these two conditions separately and then concurrently, to show that even together, the two conditions are not enough to affect the security of SIKE. In other words, if Theorem 6.12 calls Oracle 7.12 (instead of calling Oracle 6.11, as discussed in Remark 7.14), then Theorem 6.12 runs in an impractical amount of time.

Remark 7.15. Remark 6.9 shows that the following conditions are the two conditions which we could place on a matrix M with entries in $\mathbb{Z}/\ell^e\mathbb{Z}$ to attain a large number of eigenvectors (that is, to ensure ζ is close to 0):

1. *Divisibility condition:* ν is large, preferably close to e, (where ν is the largest natural number such that $\ell^\nu \mid \beta$ and $\ell^\nu \mid \delta - \alpha$).
2. *Modular condition:* ξ is large, preferably close to $\frac{e-\nu}{2}$, (where ξ is the largest natural number such that $\ell^\xi \mid \frac{(\delta-\alpha)-2\beta r}{\ell^\nu}$ and $\xi \leq \frac{e-\nu}{2}$).

We investigate the weaker of these conditions, the modular condition, first. We will show that if Oracle 7.12 only outputs endomorphisms that satisfy this modular condition, then the algorithm from Remark 7.14 runs in an impractical amount of time. More specifically, we show the modular condition is too weak to use on its own in any setting.

The following theorem tells us that it is reasonably easy to find endomorphisms that satisfy the modular condition.

Theorem 7.16. *Suppose we have a convenient basis* $\{P, \pi(P)\} = \{P, \iota(P)\}$ *for* $E[\ell^e]$. *Suppose that there is an eigenvector* $\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right]$ *of a matrix* $\left[\begin{smallmatrix}\alpha & \beta\\ \gamma & \delta\end{smallmatrix}\right]$ *in* $E_0[\ell^e]$. *Then* $\nu + \xi \geq \frac{e}{2}$ *if and only if* $4y^2 + 4z^2 \equiv 4x^2 \pmod{\ell^e}$.

Proof. Since $\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right]$ is an eigenvector, we see that r satisfies

$$\gamma + (\delta - \alpha)r - \beta r^2 \equiv 0 \pmod{\ell^e}.$$

We can use the ring version of the quadratic formula on this equation to obtain

$$-2\beta r \equiv -(\delta - \alpha) \pm \sqrt{(\delta - \alpha)^2 + 4\beta\gamma} \pmod{\ell^e}.$$

Rearranging the terms, we get

$$((\delta - \alpha) - 2\beta r)^2 \equiv (\delta - \alpha)^2 + 4\beta\gamma \pmod{\ell^e}. \tag{8}$$

By definition of ν and ξ,

$$(\delta - \alpha) - 2\beta r \equiv 0 \pmod{\ell^{\nu+\xi}}.$$

Suppose that $\nu + \xi \geq \frac{e}{2}$. Rewriting the modular condition, we have that $(\delta - \alpha) - 2\beta r \equiv f\ell^{\nu+\xi} \pmod{\ell^e}$, for some $f \in \mathbb{Z}$. Substituting this in the Equation (8), we obtain,

$$(f\ell^{\nu+\xi})^2 \equiv (\delta - \alpha)^2 + 4\beta\gamma \pmod{\ell^e},$$

implying that

$$(\delta - \alpha)^2 + 4\beta\gamma \equiv 0 \pmod{\ell^e},$$

since $\nu + \xi \geq \frac{e}{2}$. Using the matrix representation of the endomorphism, $\left[\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right] = \left[\begin{smallmatrix} w-z & -x+y \\ x+y & w+z \end{smallmatrix}\right]$, our equation transforms to,

$$(2z)^2 + 4(x+y)(-x+y) \equiv 0 \pmod{\ell^e}.$$

This implies that

$$4y^2 + 4z^2 \equiv 4x^2 \pmod{\ell^e}.$$

Conversely, suppose that $4y^2 + 4z^2 \equiv 4x^2 \pmod{\ell^e}$. Then, using the matrix representation as before in the ring version of the quadratic formula for r, we obtain

$$\begin{aligned} -2\beta r &\equiv -(\delta - \alpha) \pm \sqrt{(\delta-\alpha)^2 + 4\beta\gamma} \pmod{\ell^e} \\ &\equiv -2z \pm \sqrt{4z^2 + 4(y^2 - x^2)} \pmod{\ell^e} \\ &\equiv -2z + f\ell^c \pmod{\ell^e}, \end{aligned}$$

where $(f\ell^c)^2 = 0 \pmod{\ell^e}$ for some $c \geq \frac{e}{2}$ and $f \in \mathbb{Z}$. We substitute this as follows,

$$(\delta - \alpha) - 2\beta r \equiv 2z - 2z + f\ell^c \pmod{\ell^e} \equiv 0 \pmod{\ell^c}.$$

By the definition of ν and ξ, $\ell^{\nu+\xi}$ is the largest power of ℓ that divides $(\delta - \alpha) - 2\beta r$. Hence, $\nu + \xi \geq c \geq \frac{e}{2}$. □

Remark 7.17. Although Theorem 7.16 allows us to easily construct solutions where ξ is almost $\frac{e-\nu}{2}$ (see Example A.8), it is not clear how to obtain solutions with a small value of k. If such an endomorphism with a small k were to be found, then we could gain information about Bob's private key (in $E[N_1]$). However, even if endomorphisms with a small k could be found, the following proposition, Proposition 7.18, states that using endomorphisms that satisfy only the modular condition does not yield an algorithm with a practical runtime.

Proposition 7.18. *Endomorphisms which only satisfy the modular condition (that is, $\nu = 0$, ξ is large, $\ell^\xi = \frac{(\delta-\alpha)-2\beta r}{\ell^\nu}$, and $\xi \leq \frac{e-\nu}{2}$), have eigenspace no larger than $2\ell^{e/2}$.*

Proof. Since $\nu = 0$, it directly follows that $\xi \leq \frac{e-\nu}{2} = \frac{e}{2}$ by definition. The result follows from Remark 6.9. □

Corollary 7.19. *If Oracle 7.12 outputs endomorphisms that only satisfy the modular condition, the information entropy $H(\rho)$ gained from one output of the oracle is at most $2e\ell^{-\frac{e}{2}} \log \ell$.*

Proof. See Appendix A.2.

Result 7.20. *If Oracle 7.12 outputs endomorphisms that only satisfy the modular condition, then by Corollary 7.19, the parameters of SIKE are impractical to attack by our method of instantiating Oracle 6.11 as Oracle 7.12 in Theorem 6.12.*

We will now consider the divisibility condition, in particular, the implication the divisibility condition has on (w_0, x_0, y_0, z_0) for an endomorphism

$$\phi_C = [w_0] \cdot 1 + [x_0] \cdot \iota + [y_0] \cdot \pi + [z_0] \cdot \iota\pi \in \mathcal{R}.$$

Lemma 7.21. *Suppose the matrix*

$$M_0 = \left[\begin{smallmatrix} w_0-z_0 & -x_0+y_0 \\ x_0+y_0 & w_0+z_0 \end{smallmatrix}\right] = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}.$$

has at least one eigenvector of the form $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$. *If* $\ell \neq 2$, *then* $\ell^\nu \mid x_0, y_0, z_0$ *if and only if* $\ell^\nu \mid \beta, \delta - \alpha$.

Proof. The forward direction is straightforward. Conversely, if $\ell^\nu \mid \delta - \alpha$, then $\ell^\nu \mid z_0$. Since $\ell^\nu \mid \beta$, showing that $\ell^\nu \mid \gamma$ would give $\ell^\nu \mid x_0, y_0$. From Lemma A.1, we have

$$\beta r^2 + (\alpha - \delta)r - \gamma \equiv 0 \pmod{\ell^e}.$$

Then, since $\ell^\nu \mid \beta, \delta - \alpha$, we see that $\ell^\nu \mid \gamma$. □

If N_1 and N_2 are prime powers of 2 and 3, then there are no endomorphisms with $y = z = 0$ that have non-trivial eigenspaces (see Corollary A.3). Thus Lemma 7.22 is relevant for all desirable endomorphisms in the SIKE setting. It provides an obstruction to finding endomorphisms that satisfy the divisibility condition.

Lemma 7.22. *Assume the following:*

1. $E_0(\mathbb{F}_{p^2})$ *has* j*-invariant* 1728, *where* $p = N_1 N_2 - 1$,
2. $\phi_C = [w_0] \cdot [1] + [x_0] \cdot \iota + [y_0] \cdot \frac{[1]+\pi}{2} + [z_0] \cdot \frac{\iota+\iota\pi}{2}$ *is an endomorphism on* E_0 *of degree* Lk *where* $L \mid (N_2)^2$,
3. *at least one of* y_0 *and* z_0 *is nonzero, and*
4. $\ell^\mu \mid x_0, y_0, z_0$ *for a positive integer* μ.

Then k *has a lower bound of approximately* $\ell^{2\mu}$.

Proof. By Lemma 7.4, (w_0, x_0, y_0, z_0) is a solution to

$$w^2 + x^2 + \left(\tfrac{p+1}{4}\right)(y^2 + z^2) + wy + xz = Lk. \tag{9}$$

By Lemma 7.5, $(w_1, x_1, y_1, z_1) = (2w_0 + y_0, 2x_0 + z_0, y_0, z_0)$ is a solution to

$$w^2 + x^2 + p(y^2 + z^2) = 4Lk,$$

and it is clear that $\ell^\mu \mid x_0, y_0, z_0$ implies that $\ell^\mu \mid x_1, y_1, z_1$. Therefore, we can define (w_2, x_2, y_2, z_2) such that $(w_2, \ell^\mu x_2, \ell^\mu y_2, \ell^\mu z_2) = (w_1,\ x_1,\ y_1,\ z_1)$. Then (w_2, x_2, y_2, z_2) is a solution to

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

This implies:

$$\ell^{2\mu} p(y^2 + z^2) \leq 4Lk.$$

The statement of Lemma 7.22 follows from the fact that $L \mid (N_2)^2 \approx p$. □

Remark 7.23. Having $\ell^\nu \mid x, y, z$ where ν is close to e is the best way to get an endomorphism ϕ_C with enough eigenvectors so that there is a reasonably high probability that a random point in $E_0[\ell^e]$ is an eigenvector of ϕ_C. However, Lemma 7.22 and Proposition A.2 show that requiring ν to be close to e forces to k to be impractically large in the SIKE/SIDH setting.

Result 7.24. *If Oracle 7.12 outputs endomorphisms that only satisfy the divisibility condition, then by Lemma 7.22, the parameters of SIKE are impractical to attack by our method of instantiating Oracle 6.11 as Oracle 7.12 in Theorem 6.12.*

We now consider endomorphisms that satisfy both division and modular conditions.

Proposition 7.25. *The expected runtime for Algorithm 6.13 when applied to 2-party SIKE (with the parameters as above) is at least* $\ell^{\frac{e}{2}}$.

Proof. Let ν and ξ be defined as in Notation 6.6 with respect to M. The modular condition is optimal at $\xi = \frac{e-\nu}{2}$, which would imply there are $\ell^{\nu+\xi} = \ell^{\frac{e+\nu}{2}}$ eigenvectors. There would need to be at least $\ell^{\frac{e-\nu}{2}}$ endomorphisms to cover the space of possible eigenvectors.

The expected event is that after $\frac{1}{2}\ell^{\frac{e-\nu}{2}}$ many calls to the oracle, a kernel-fixing endomorphism is found. (Of course, our algorithm will likely require more than one kernel-fixing endomorphism, but even just finding one requires approximately $\frac{1}{2}\ell^{\frac{e-\nu}{2}}$ many calls to the oracle.) Hence, if m is the expected number of calls to the oracle (as in Theorem 5.11), we have $m \geq \frac{1}{2}\ell^{\frac{e-\nu}{2}}$. By Lemma 7.21 and Lemma 7.22, k is at least $\ell^{2\nu}$. Thus the expected runtime for Algorithm 6.13 has the lower bound

$$m\sqrt{k} \geq \ell^{\frac{e-\nu}{2}}\sqrt{k} \geq \ell^{\frac{e-\nu}{2}}\ell^{\nu} = \ell^{\frac{e+\nu}{2}}.$$

This is optimal when $\nu = 0$. □

Result 7.26. *By Proposition 7.25, the runtime of our approach applied to SIKE has a lower bound of* $\ell_1^{\frac{e_1}{2}}$. *This is similar to well-known classical attacks.*

Remark 7.27. Figure 8 depicts the different sets of endomorphisms in the SIKE case. Recall Notation 6.14.

- Black area: endomorphisms satisfying only the divisibility condition (ν is close to e).
- Blue ellipse: endomorphisms with many eigenvectors (ζ is close to 0). These endomorphisms with many eigenvectors are said to satisfy the mixed condition; either divisibility condition holds strongly, or the modular condition holds and the divisibility condition holds (weakly).
- Purple area: endomorphisms which have the desired degree conditions (with small k).

By Result 7.26, the blue and purple areas in Figure 8 do not overlap; that is, there are no desirable endomorphisms, and so our methods are infeasible in the SIKE setting. This result holds against Bob's private key, but a similar analysis suggests our methods cannot be used to find Alice's private key. Section 8 will provide similar figures for alternative settings.

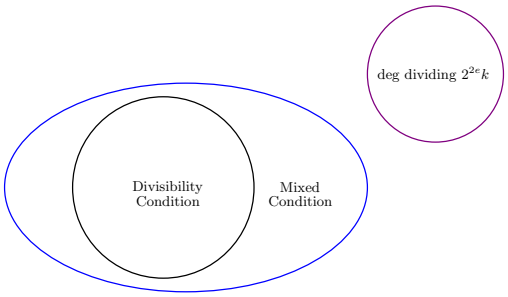

Fig. 8: Endomorphisms in the SIKE setting

We detail some examples in the Appendix that satisfy a subset of these constraints. Example A.7 presents an endomorphism with small k degree while Example A.8 presents an endomorphism satisifying the modular condition alone. However, as Figure 8 illustrates, there do not exist endomorphisms in the SIKE setting that satisfy all the conditions necessary for our methods to yield an efficient attack.

In this subsection we proved the negative result that Remark 7.14 does not provide a practical algorithm to attack Bob's key in the SIKE setting. However, there is more potential to create a practical oracle to use in Theorem 6.12 for supersingular isogeny-based algorithms in different settings, as will be shown in the following section.

8 Alternate Settings

In the previous sections, we were interested in fields of prime characteristic of the form $p = N_1N_2 - 1$, where $N_1 \approx N_2$, as this is what is needed for SIKE. However, in the literature, there are supersingular isogeny-based n-party algorithms that use primes of the form $p = N_1 \cdot \ldots \cdot N_n - 1$, where $N_1 \approx N_2 \approx \cdots \approx N_n$ [3]. In Sections 8.1 and 8.2 we will consider the 3-party and 4-party case, respectively. In Section 8.3 we will consider the unbalanced case, where the prime has the form $p = N_1N_2 - 1$, where $N_1 \approx N_2^{n-1}$ for some $n \geq 2$.

8.1 3-Party Setting

As was the case in the 2-party setting with Result 7.20, in the 3-party setting the modular condition alone does not provide endomorphisms with a large enough percentage of eigenvectors to make Remark 7.14 practical. Thus, we need endomorphisms that satisfy the divisibility condition.

Notation 8.1. We introduce the following notation specific to the 3-party setting:

- Let $p = N_1N_2N_3 - 1$, where $p = N_1N_2N_3 - 1$, for $N_1 \approx N_2 \approx N_3$.
- Let $N_1 = \ell^e$ and N_2, N_3 be $\log p$-smooth.
- Let E_0 be the elliptic curve with j-invariant 1728 over $\mathbb{F}_{p^2}$.
- Each of the parties will generate a private isogeny ϕ_1, ϕ_2, and ϕ_3, from an element of full order in $E_0[N_1], E_0[N_2]$, and $E_0[N_3]$, respectively.

Without loss of generality, we will consider an attacker trying to find the first party's private key in $E_0[N_1]$. We still have the following:

- The endomorphisms has the form $\phi_C = [w] \cdot [1] + [x] \cdot \iota + [y] \cdot \frac{[1]+\pi}{2} + [z] \cdot \frac{\iota+\iota\pi}{2}$.
- The degree of ϕ_C is $q(w,x,y,z) = w^2 + x^2 + \left(\frac{p+1}{4}\right)(y^2+z^2) + wy + xz$, and as shown in Lemma 7.5, solutions to q can be translated to and from similar solutions to $q_0(w,x,y,z) = w^2 + x^2 + p(y^2+z^2)$.
- For simplicity we will also assume there is a convenient basis $\{P_1, Q_1\}$ for $E_0[N_1]$. Thus ϕ_C acts as the matrix
$$M_0(w,x,y,z) = \left[\begin{smallmatrix} w-z & -x+y \\ x+y & w+z \end{smallmatrix}\right]$$
on $E_0[N_1]$ with respect to $\{P_1, Q_1\}$.
- ν and ξ are defined as in Notation 6.6 with respect to M_0.

Analogous to Lemma 7.22, the following lemma provides an obstruction to finding endomorphisms that satisfy the divisibility condition. In particular, although endomorphisms satisfying the divisibility conditions might exist, ν is not close to e.

Lemma 8.2. *Assume the following:*

1. *$E_0(\mathbb{F}_{p^2})$ has j-invariant 1728, where $p = N_1N_2N_3 - 1$ and $N_1 = \ell^e$, for $N_1 \approx N_2 \approx N_3$,*
2. *$\phi_C = [w_0] \cdot [1] + [x_0] \cdot \iota + [y_0] \cdot \frac{[1]+\pi}{2} + [z_0] \cdot \frac{\iota+\iota\pi}{2}$ is an endomorphism on E_0 of degree Lk, where $L \mid (N_2N_3)^2$,*
3. *at least one of y_0 and z_0 is nonzero, and*
4. *$\ell^\mu \mid x_0, y_0, z_0$ for a positive integer μ.*

Then k has a lower bound of approximately $\ell^{2\mu-e} > 1$.

Proof. If the point (w_0, x_0, y_0, z_0) is a solution to
$$w^2 + x^2 + \left(\tfrac{p+1}{4}\right)(y^2+z^2) + wy + xz = Lk', \tag{10}$$
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
$$w^2 + x^2 + p(y^2+z^2) = Lk,$$
where $k = 4k'$. 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
$$w^2 + \ell^{2\mu}x^2 + \ell^{2\mu}p(y^2+z^2) = Lk.$$
The statement follows from the fact that $p \approx \ell^{3e}$ and $L \leq \ell^{4e}$. □

We now attempt to optimize Algorithm 6.13 in the 3-party case.

Proposition 8.3. *The expected runtime of Algorithm 6.13 in the 3-party case is at least $\ell^{\frac{e}{4}}$.*

Proof. Consider an endomorphism $\phi_C = [w_0] \cdot [1] + [x_0] \cdot \iota + [y_0] \cdot \frac{[1]+\pi}{2} + [z_0] \cdot \frac{\iota+\iota\pi}{2}$ on E_0 of degree Lk, where $L \mid N_2N_3$. If there is a convenient basis $\{P_1, Q_1\}$ for $E_0[\ell]$, then ϕ_C acts as the matrix M_0 with respect to $\{P_1, Q_1\}$. Let ν and ξ be defined as in Notation 6.6 with respect to M_0.

Given a fixed ν, the best we can hope for from the modular condition is $\xi = \frac{e-\nu}{2}$, which implies there are $\ell^{\nu+\xi} = \ell^{\frac{e+\nu}{2}}$ eigenvectors. Therefore, at least $\ell^{\frac{e-\nu}{2}}$ endomorphisms are needed to cover the space of possible eigenvectors.

We expect Algorithm 6.13 to require $\frac{1}{2}\ell^{\frac{e-\nu}{2}}$ calls to the oracle before a kernel-fixing endomorphism is found. (Of course, our algorithm will likely require more than one kernel-fixing endomorphism, but even just finding one requires approximately $\frac{1}{2}\ell^{\frac{e-\nu}{2}}$ many calls to the oracle.) Hence, if m is the expected number of calls to the oracle (as in Theorem 5.11), we have $m \geq \frac{1}{2}\ell^{\frac{e-\nu}{2}}$. By Lemma 8.2, k is at least $\ell^{2\nu-e}$. Thus the runtime for Algorithm 6.13 has the lower bound

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

The optimal value of ν is $\frac{e}{2}$ which gives an approximate runtime as $\ell^{\frac{e}{4}}$. □

The best known quantum attacks in the 3-party case until now has been $O\left(\ell^{\frac{e}{3}}\right)$, and Result 8.4 suggests our methods could possibly give a better attack than even quantum meet-in-the-middle.

Result 8.4. *By Proposition 8.3, the runtime of our approach in the 3-party case has a lower bound of $\ell^{\frac{e}{4}}$. In the best case for our work, where k is consistently $\log p$-smooth (and therefore Algorithm 6.13 has runtime approximately $\sqrt{K} \cdot m$), the same optimal value as Proposition 8.3 of $\ell^{\frac{e}{4}}$ can be obtained.*

Remark 8.5. In order for our methods to produce a competitive attack in the 3-party case, we expect at least $\ell^{\frac{e}{4}}$ desirable endomorphisms must be found where k is small. This seems impractical, but it is possible that enough desirable endomorphisms (with k small) could be found to decrease the security of the 3-party group key exchange.

Remark 8.6. Figure 9 depicts the different endomorphisms in the 3-party setting, an analogue of Figure 8.

- Black circle: endomorphisms which satisfy the divisibility condition.
- Blue ellipse: endomorphisms with many eigenvectors.
- Purple circle: endomorphisms which have the desired degree conditions (with small k).
- Green area: Result 8.4 tells us that desirable endomorphisms could exist, giving us an improved attack in the 3-party case (even though such endomorphisms do not exist in the 2-party, that is, the SIKE setting, see Figure 8).

Desirable endomorphisms which only satisfy the division condition still do not exist (the blue ellipse and the purple circle do not intersect).

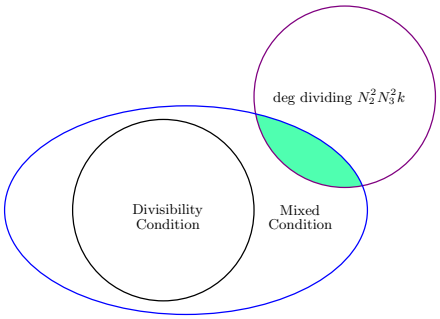

Fig. 9: Endomorphisms in the 3-party setting

8.2 4-Party Setting

There is no setting, including the 4-party setting, where the modular condition alone provides endomorphisms with a large enough percentage of eigenvectors to make Theorem 6.12 practical. When it comes to the divisibility condition, in the 4-party case there is no analogy to Lemmas 7.22 and 8.2.

In this subsection we will use the obvious adaption of Notation 8.1 for the 4-party setting. Assuming there is a convenient basis $\{P_1, Q_1\}$ for $E_0[N_1]$, we get the same matrix $M_0(w, x, y, z)$ and quadratic form $q_0(w, x, y, z)$. We will consider an attacker trying to find the first party's private key in $E_0[N_1]$.

For the 4-party scheme to be secure, the following problem must be hard.

Problem 8.7. Let p be a prime number of the form $p = N_1N_2N_3N_4 - 1$. Suppose there is a convenient basis of $E[N_1]$. Find (w_0, x_0, y_0, z_0) that satisfies

$$w^2 + \ell^{2\nu}x^2 + p\ell^{2\nu}(y^2 + z^2) = Lk$$

where $\nu + \xi$ is close to e, $\gcd(k, \ell) = 1$, $k = 4k'$ is small, and $L \mid (N_2N_3N_4)^2$.

Remark 8.8. When it comes to the divisibility condition, in the 4-party case, there is no restriction on the number of eigenvectors an endomorphism can have that is analogous to the restrictions given in Lemmas 7.22 and 8.2. However, it is likely that endomorphisms exist that satisfy only the divisibility condition. We postpone the analysis of this case to Section 8.3.

Remark 8.9. Figure 10 depicts the different endomorphisms in the 4-party setting, and is an analogue of Figures 8 and 9.

- Black circle: endomorphisms which satisfy the divisibility condition.
- Blue ellipse: endomorphisms with many eigenvectors.
- Purple circle: endomorphisms which have the desired degree conditions (with small k).
- Green area: Remark 8.8 tells us that desirable endomorphisms (the green area in Figure 9) could exist which give us an offline attack in the 4-party case (even though such endomorphisms do not exist in the 2-party, that is, the SIKE setting, see Figure 8). If enough of these endomorphisms have ν close enough to e and small enough k, then the private key could be found in polynomial time.

Additionally, desirable endomorphisms which only satisfy the division condition are possible (the intersection between the black and purple circles). In Section 8.3, we give a heuristic argument stating that it is, in fact, likely that some desirable endomorphisms exist in the 4-party situation.

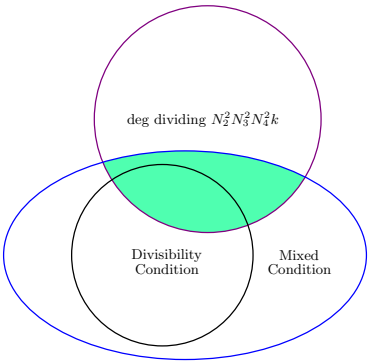

Fig. 10: Endomorphisms in the 4-party setting

8.3 Unbalanced Setting

We now consider the general n-unbalanced case, and then we use this analysis to determine where our results begin to outperform the best known attacks.

We use the field $\mathbb{F}_{p^2}$, with $p = N_1N_2 - 1$, where $N_1^{n-1} \approx N_2$. We use the obvious adaptation of Notation 8.1. Assuming there is a convenient basis $\{P_1, Q_1\}$ for $E[N_1]$, we get the same matrix $M_0(w, x, y, z)$ and quadratic form $Q_0(w, x, y, z)$.

We can now analyze the relationship needed between N_1 and N_2 to allow desirable endomorphisms to exist which satisfy the divisibility condition (ν is close to e). The analogous quadratic form to that given in Lemma 7.22 is

$$w^2 + N_1^2x^2 + N_1^2p(y^2 + z^2) = N_2^2k.$$

This implies

$$w^2 + N_1^2x^2 + N_1^3N_2(y^2 + z^2) \approx N_2^2k.$$

Suppose we impose the restriction of $k = O(\log^r p)$ for some constant r. Then w and x will not dominate the asymptotics. Thus,

$$N_1^3(y^2 + z^2) \approx N_2k$$

and $N_1^3 \in \tilde{O}(N_2)$. Therefore, $N_1^2 \not\approx N_2$. However, $N_1^3 \approx N_2$ does not provide an obstruction, (like with Lemmas 7.22 and 8.2). This lack of an obstruction when $N_1^3 \approx N_2$ is similar to the lack of obstruction in a 4-party scheme.

For the unbalanced scheme to be secure, we need it to be hard to find many desirable endomorphisms. Hence, similar to Problem 8.7 in the 4-party case, the following problem must be hard.

Problem 8.10. Let p be a prime number of the form $p = N_1N_2 - 1$, where $N_1^3 \approx N_2$. Find (w_0, x_0, y_0, z_0) such that

$$w^2 + \ell^{2\nu}x^2 + \ell^{2\nu}p(y^2 + z^2) = Lk$$

for small k, $\gcd(k, \ell) = 1$, ν close to e, and $L \mid N_2^2$.

We have the following heuristic argument for why we expect solutions to Problem 8.10 to exist.

Remark 8.11. Consider the following case where $p = N_1N_2 - 1$ and $N_1^3 = (3^b)^3 \approx 2^a = N_2$ (the more general case is similar). Recall that the supersingular ℓ-isogeny graph is a Ramanujan graph with exactly $\frac{p+13}{12}$ vertices [13].

We consider isogenies of degree some multiple of 2^{2a}. With this size of degree, we have passed the mixing number (that is, the codomain of the isogeny has any supersingular j-invariant with probability at least $\frac{6}{p+13}$), which is $\frac{\log \frac{p+13}{6}}{\log 3/2\sqrt{2}}$-many steps in the 2-graph [13]. Thus we are guaranteed the existence of an endomorphism of degree $2^{2a}k$, and we expect there to be at around $\frac{2^{2a}k}{p} \approx p^{\frac{1}{2}}k$ endomorphisms with this degree.

Now we address the relationship between the quantity of endomorphisms and the size of their eigenspaces to better understand the impact of our results in this setting. By increasing k we expect to get many endomorphisms. However, the requirement of many eigenvectors increases the coefficients in the quadratic form. By expecting x, y, z to all be divisible by approximately 3^b it is unlikely that there are solutions in the SIKE setting. Yet, by assuming $3^{3b} \approx 2^a$ and $k \in O(3^{b/2})$, it again becomes reasonable to expect some solutions where x, y, z are divisible by approximately 3^b. This follows from the fact that there are approximately k isogenies of degree k and approximately 3^b isogenies of degree bounded above by $3^{\frac{b}{2}}$. Therefore, we have a heuristic argument why Problem 8.7 has solutions.

As the 4-party case is similar to the unbalanced case where $N_1^3 \approx N_2$, we could give a similar (although slightly more involved) heuristic argument to the one given in Remark 8.11 that suggests there likely exist desirable endomorphisms in the 4-party setting. We have yet to construct these desirable endomorphisms in the 4-party setting, but if found, then Theorem 6.12 will lead to a reduction of security. If enough desirable endomorphisms are found, this could result in a polynomial time attack for recovering private keys in the 4-party setting.

8.4 Summary of our Results

We conclude this section with a summary of our analysis and how our results compare with the best attacks currently known on the different settings.

- 2-*party*. A mixed approach of the modular and divisibility condition does not produce an attack that is better than previously studied attacks. Recall that the best known classical attack runs in $O(N_1^{\frac{1}{2}})$ [1] and in $O(N_1^{\frac{1}{3}})$ for the best quantum one [14], [24].
- *Unbalanced 2-party*. We show a potentially better exponential attack on an unbalanced 2-party key establishment where $p = N_1N_2 - 1$ whenever $N_2 > N_1^2$, and a potential polynomial attack whenever $N_2 > N_1^3$. The only possibly better classical attack in this setting comes from [17], where the authors obtain a polynomial attack in the largely unbalanced case where $\log N_2 \in O(\log^2 N_1)$.

- 3-*party*. A mixed approach of the modular and divisibility conditions could potentially produce a $O(N_1^{\frac{1}{4}})$ exponential attack in the worst case, which is a significant improvement over the known generic attacks (which are the same as in the 2-party case).
- 4-*party*. The divisibility condition alone could be enough to produce a polynomial time attack, much better than the generic attacks (again, the same as in the 2-party case). Desirable endomorphisms likely exist that could be used to weaken the security of 4-party isogeny-based protocols.

9 Using Endomorphisms with Almost-Eigenvectors

Note: This section and the following section are still under revision. Since they are relevant to the other results in our paper, we considered it important to include them.

In this section we wish to address one case in which Algorithm 5.12 is inefficient. Specifically, we consider the case where the endomorphisms found do not have enough eigenvectors (in the sense that ρ is too small), even if they are very efficient endomorphisms (in the sense that k is small). If the resulting values of k associated to each of these endomorphisms are small enough, it might still be possible to use the method described in this section to transform them into more desirable endomorphisms. This method works for endomorphisms that have many eigenvectors in $E[N]$, for some large divisor N of N_1, (although possibly a smaller number of eigenvectors for N_1).

We start in Section 9.1, by discussing endomorphisms that preserve (and almost preserve) Alice's kernel. In Section 9.2 we discuss how this theory can be used to create a new oracle and associated reduction theorem.

9.1 Almost-Invariant Kernels

In this subsection we consider another well-known method to construct endomorphisms on E_A. We will see that this other construction gives us an alternative way of viewing the endomorphism ψ_C. In the next subsection, we will use this viewpoint to relax the condition on the endomorphisms Oracle 5.9 outputs, by making certain endomorphisms (with eigenspaces that are slightly too small) more desirable.

Given an endomorphism ϕ_C on E, there is a natural endomorphism of E_A induced by ϕ_A, namely $\phi_A \circ \phi_C \circ \widehat{\phi}_A$, as shown in Figure 11.

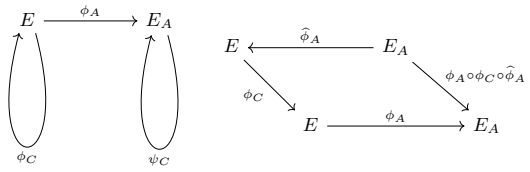

Fig. 11: Endomorphism induced by ϕ_A

In fact, ψ_C and $\phi_A \circ \phi_C \circ \widehat{\phi}_A$ are scalar multiples of each other if and only if $\ker \phi_A$ is invariant under ϕ_C. To show this, we begin by giving an explicit description of the kernel of $\widehat{\phi}_A$.

Lemma 9.1. *Suppose* $\{P, Q\}$ *forms a basis of* $E[N]$*, and* $\phi : E \to E'$ *denotes a separable isogeny where* $\ker \phi = \langle P \rangle$*. Then* $\phi(Q)$ *generates the kernel of the dual isogeny* $\widehat{\phi}$*. Moreover, if* $\phi(R)$ *generates* $\ker \widehat{\phi}$*, then* $\{P, R\}$ *forms a basis of* $E[N]$*.*

Proof. Let ϕ' be an isogeny with domain E' and kernel $\langle \phi(Q) \rangle$. Then $\langle P, Q \rangle \subset \ker(\phi' \circ \phi)$, since

$$\begin{array}{ccccc} P & \xrightarrow{\phi} & \mathcal{O} & \xrightarrow{\phi'} & \mathcal{O} \\ \mathrm{Q} & \xrightarrow{\phi} & \phi(Q) & \xrightarrow{\phi'} & \mathcal{O} \end{array}$$

Thus

$$\ker[N] = E[N] = \langle P, Q \rangle \subset \ker(\phi' \circ \phi).$$

As

$$|\ker[N]| = N^2 = \deg(\phi' \circ \phi) = |\ker(\phi' \circ \phi)|,$$

it follows that $[N] \cong \phi' \circ \phi$. By [21, §III, Theorem 6.1 (a)], this implies ϕ' is the dual of ϕ.

To see the second statement, notice that $\phi(R) = [\lambda'] \cdot \phi(Q)$ for some λ' relatively prime to N. Thus, $R = [\lambda'] \cdot Q + [\lambda] \cdot P$ for some $[\lambda] \cdot P \in \ker \phi$. Therefore, $\langle P, Q \rangle = \langle P, R \rangle$. □

While Lemma 9.1 described $\ker \widehat{\phi}_A$, Lemma 9.2 describes $\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A)$, as illustrated in the commutative diagram in Figure 12. Understanding the kernel of $\phi_A \circ \phi_C \circ \widehat{\phi}_A$ will allow us to relate ψ_C and $\phi_A \circ \phi_C \circ \widehat{\phi}_A$.

Lemma 9.2. *Suppose* $\ker(\phi_A) = \langle P_A + [r_A] \cdot Q_A \rangle = \langle R_A \rangle$*,* $\ker(\phi_C) = \langle R_C \rangle$*, and* $\gcd(N_1, \deg \phi_C) = 1$*. Then*

$$\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A) = \langle \phi_A(Q_A), \phi_A(R_C), S \rangle,$$

for any $S \in E_A[(N_1)^2]$ *such that* $\langle \widehat{\phi}_A(S) \rangle = \langle \widehat{\phi}_C(R_A) \rangle$*.*

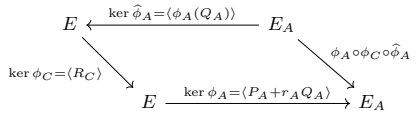

Fig. 12: Endomorphism induced by ϕ_A with Kernels

Proof. We can see that $\langle \phi_A(Q_A), \phi_A(R_C), S \rangle \subseteq \ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A)$, since

$$\begin{array}{ccccccc} \phi_A(Q_A) & \xrightarrow{\widehat{\phi}_A} & [N_1] \cdot Q_A = \mathcal{O} & \xrightarrow{\phi_C} & \mathcal{O} & \xrightarrow{\phi_A} & \mathcal{O} \\ \phi_A(R_C) & \xrightarrow{\widehat{\phi}_A} & [N_1] \cdot R_C & \xrightarrow{\phi_C} & \mathcal{O} & \xrightarrow{\phi_A} & \mathcal{O} \\ S & \xrightarrow{\widehat{\phi}_A} & \widehat{\phi}_A(S) = \widehat{\phi}_C(R_A) & \xrightarrow{\phi_C} & [N_1] \cdot R_A & \xrightarrow{\phi_A} & \mathcal{O} \end{array}$$

Now, we finish the proof by showing that the subgroup has the same order as the kernel and is thus equal to it. Note that

$$\begin{aligned}|\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A)| &= \deg \phi_A \deg \phi_C \deg \widehat{\phi}_A \\ &= (N_1)^2 \deg \phi_C.\end{aligned}$$

By the assumption $\gcd(N_1, \deg \phi_C) = 1$, we obtain that $|\langle \phi_A(R_C) \rangle| = \deg \phi_C$. By the first isomorphism theorem of $\widehat{\phi}_A$ restricted to $\langle \phi_A(Q_A), S \rangle$,

$$\begin{aligned}|\langle \phi_A(Q_A), S \rangle| &= |\ker \widehat{\phi}_A| \cdot |\widehat{\phi}_A(\langle \phi_A(Q_A), S \rangle)| \\ &= \deg \phi_A \cdot |\langle \widehat{\phi}_A(S) \rangle| \\ &= \deg \phi_A \cdot |\langle \widehat{\phi}_C(R_A) \rangle| \\ &= N_1 \cdot |\langle R_A \rangle| \\ &= (N_1)^2.\end{aligned}$$

Hence,

$$\begin{aligned}|\langle \phi_A(Q_A), \phi_A(R_C), S \rangle| &= |\langle \phi_A(R_C) \rangle| \cdot |\langle \phi_A(Q_A), S \rangle| \\ &= N_1^2 \deg \phi_C.\end{aligned}$$

□

We can now use this knowledge of $\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A)$ to show $\phi_A \circ \phi_C \circ \widehat{\phi}_A$ is essentially $[N_1] \cdot \psi_C$, as seen in Figure 13.

Theorem 9.3. *Let ψ_C be the isogeny with kernel $\langle \phi_A(R_C) \rangle$, and* $\gcd(\deg \phi_C, N_1) = 1$. *Then* $\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A) = \ker([N_1] \cdot \psi_C)$ *if and only if* $\phi_C(\ker \phi_A) = \ker \phi_A$. *Furthermore, when this holds,* $\phi_A \circ \phi_C \circ \widehat{\phi}_A \cong [N_1] \cdot \psi_C$.

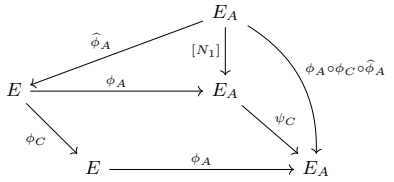

Fig. 13: ψ_C and the endomorphism induced by ϕ_A

Proof. We will start by considering $\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A)$. By Lemma 9.2 we know that

$$\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A) = \langle \phi_A(Q_A), \phi_A(R_C), S \rangle,$$

for any $S \in E_A[(N_1)^2]$ such that $\widehat{\phi}_A(S) = [\lambda] \cdot \widehat{\phi}_C(R_A)$ for some natural number λ, where $\gcd(\lambda, N_1) = 1$. Since $\gcd(N_1, \deg \phi_C) = 1$, we find

$$\begin{aligned}\ker(\phi_A \circ \phi_C \circ \widehat{\phi}_A) &= \langle \phi_A(Q_A), S \rangle \oplus \langle \phi_A(R_C) \rangle \\ &= \langle \phi_A(Q_A), S \rangle \oplus \ker \psi_C, \qquad (11)\end{aligned}$$

where $\oplus$ denotes a direct sum. On the other hand, since $\gcd(N_1, \deg \phi_C) = 1$, we have the group decomposition:

$$\ker([N_1] \cdot \psi_C) = E_A[N_1] \oplus \ker \psi_C. \qquad (12)$$

Equations (11) and (12) imply that it suffices to show that $\langle \phi_A(Q_A), S\rangle = E_A[N_1]$ if and only if $\ker \phi_A$ is invariant under ϕ_C.

Suppose $\phi_C(\ker \phi_A) = \ker \phi_A$. Since $\gcd(N_1, \deg \phi_C) = 1$, this implies $\ker \phi_A = \widehat{\phi}_C(\ker \phi_A)$, or equivalently, $\widehat{\phi}_C(R_A) = [\lambda'] \cdot R_A$ for some λ', where $\gcd(\lambda', N_1) = 1$. By the definition of S this implies $\widehat{\phi}_A(S) = [\lambda\lambda'] \cdot R_A$. Since $\widehat{\widehat{\phi}}_A = \phi_A$, $\widehat{\phi}_A(S)$ generates $\ker \widehat{\widehat{\phi}}_A$. By Lemma 9.1, not only $\phi_A(Q_A) = \ker \widehat{\phi}_A$, but also $\{\phi_A(Q_A), S\}$ forms a basis of $E_A[N]$.

Conversely, suppose $\langle \phi_A(Q_A), S\rangle = E_A[\deg \phi_A]$. By Lemma 9.1, $\langle \widehat{\phi}_A(S)\rangle = \ker \widehat{\widehat{\phi}}_A = \ker \phi_A$. Thus $\widehat{\phi}_A(S) = [\lambda'] \cdot R_A$, for some λ', where $\gcd(\lambda', N_1) = 1$. As $\widehat{\phi}_A(S) = [\lambda] \cdot \widehat{\phi}_C(R_A)$, this implies $[\lambda] \cdot \widehat{\phi}_C(R_A) = [\lambda'] \cdot R_A$. Since $\gcd(N_1, \deg \phi_C) = 1$, this proves $\phi_C(\ker \phi_A) = \ker \phi_A$. □

Theorem 9.3 explains why our methods work. In particular, if Alice's private point is invariant under an endomorphism ϕ_C, then $\phi_A \circ \phi_C \circ \widehat{\phi}_A$ is a scalar multiple of an endomorphism ψ_C on E_A, which means ψ_C exists. However, Theorem 9.3 also suggests a generalization for the case where Alice's private point is *almost* invariant under ϕ_C. We will explore this generalization now, along with its ramifications.

Moreover, if N_2 is $\log p$-smooth, for example if it has the form $N_2 = \ell^e$, then the proof of Theorem 9.3 can be generalized to give us Theorem 9.6. In particular, if the kernel of ϕ_A is almost-invariant under ϕ_C, then there is an endomorphism of E_A that has almost the same degree as ϕ_C. Before proving this statement, we require additional notation and definitions.

Definition 9.4. We call a point R of order $N_1 = \ell^e$ an *almost-eigenvector* (by ℓ^{e-d} for some $d \in \{0, \ldots, e\}$) of an endomorphism ϕ on E if

$$\phi([\ell^{e-d}] \cdot R) = [\lambda] \cdot ([\ell^{e-d}] \cdot R)$$

for some integer λ satisfying $\gcd(\lambda, \deg \phi) = 1$. This is equivalent to saying $[\ell^{e-d}] \cdot R \in \mathsf{Eig}_{\ell^d} \phi$.

Notation 9.5. Suppose $N_1 = \ell^e$. For the remainder of this section we fix $d \in \{0, \ldots, e\}$. Let ϕ_A^d denote the isogeny on E with kernel $\langle [\ell^{e-d}] \cdot R_A \rangle$, and let E_A^d denote the image of ϕ_A^d. Let ψ_C^d denote the isogeny on E_A^d with kernel $\langle \phi_A^d(R_C)\rangle$, and let E_{CA}^d denote the image of ψ_C^d, see Figure 14.

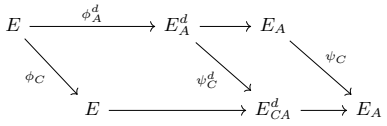

Fig. 14: Partial isogenies

With this notation in mind, we now generalize Theorem 9.6 to almost-eigenvectors.

Theorem 9.6. *Suppose* $\gcd(\deg \phi_C, N_1) = 1$*, where* $N_1 = \ell^e$ *for some natural number* e*, and let* $d \in \{0, \ldots, e\}$*. Then* $\ker(\phi_A^d \circ \phi_C \circ \widehat{\phi}_A^d) \cong [\ell^d] \cdot \ker(\psi_C^d)$ *if and only if* $\phi_C([\ell^{e-d}] \cdot \ker \phi_A) = [\ell^{e-d}] \cdot \ker \phi_A$*.*

This proof is similar to the proof for Theorem 9.3, and the concept is illustrated in Figure 15.

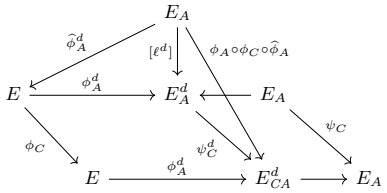

Fig. 15: Almost-Invariant kernels

Next we use Theorem 9.6 to generalize our reduction given in Theorem 5.7.

9.2 Almost-Eigenvectors

In previous sections, we always assumed that Alice's kernel was invariant under the endomorphism ϕ_C. However, Theorem 9.6 suggests that it might be possible to use endomorphisms where Alice's private point is an almost-eigenvector. More technically, we are interested in endomorphisms that have many eigenvectors in $E[N]$, for some large divisor N of N_1. This gives a natural generalization of our earlier discussions. We start with a generalization of Theorem 5.7.

Theorem 9.7. *Suppose we are given*

1. *a starting supersingular elliptic curve $E(\mathbb{F}_{p^2})$ such that $p = N_1 N_2 - 1$ for coprime $N_1 = \ell^e$ and $\log p$-smooth N_2,*
2. *the image of an N_1-degree isogeny $E_A = \phi_A(E)$ which has kernel $\langle R_A \rangle$,*
3. *the action of ϕ_A restricted to the N_2-torsion points $\phi_A|_{E[N_2]}$*
4. *$d \in \{0, \ldots, e\}$, and*
5. *triangular kernel $\ker_\triangle \phi_C$ of an Lk-degree endomorphism ϕ_C in $\mathbb{F}_{p^2}$ where*
   (a) $\gcd(k, N_1) = 1$, *and*
   (b) $L \mid (N_2)^2$.

Then there exists a (classical) algorithm which runs in time $\tilde{O}(\ell^{e-d} k^3)$ and that decides whether $[\ell^{e-d}] \cdot R_A \in \mathsf{Eig}_{\ell^d} \phi_C$ or $[\ell^{e-d}] \cdot R_A \notin \mathsf{Eig}_{\ell^d} \phi_C$ with overwhelmingly high probability. Further, if k is $\log p$-smooth, then the runtime is $\tilde{O}\left(\ell^{e-d}\sqrt{k}\right)$.

We first describe such an algorithm and then analyze its runtime as a proof of the theorem.

Algorithm 9.8.
Input: E_A, $\phi_A|_{E[N_2]}$, $\ker_\triangle \phi_C = (K_0, K_1, K_2)$, a number $d \in \{0, \ldots, e\}$, and a natural number k.
Output: True if $[\ell^{e-d}] \cdot R_A \in \mathsf{Eig}_{\ell^d} \phi_C$, and False if $[\ell^{e-d}] \cdot R_A \notin \mathsf{Eig}_{\ell^d} \phi_C$.

1. Calculate all the possible ℓ^{e-d} isogenies from E_A. Denote them ϕ_i.
2. For each isogeny ϕ_i, calculate $\phi_i(E_A), \phi_i(\phi_A(P_B))$ and $\phi_i(\phi_A(Q_B))$.

3. For each isogeny ϕ_i, run Algorithm 5.8 with $\phi_i(E_A), \phi_i(\phi_A(P_B)), \phi_i(\phi_A(Q_B))$, (K_0, K_1, K_2) and k. If it returns True, then return True.
4. Return False.

Proof. Since the worst case runtime (when k is prime) of Algorithm 5.8 is $\tilde{O}(k^3)$, the worst case runtime of Algorithm 9.8 is $\tilde{O}(\ell^{e-d}k^3)$. □

We can now generalize Oracle 6.11, to produce almost-eigenvectors:

Oracle 9.9.
Input: N_2, a set $S \subseteq E[\ell^d]$ (where $d \in \{0, \ldots, e\}$), a quadratic form $q(w, x, y, z)$, a 2×2-matrix $M(w, x, y, z)$ acting on $E[\ell^d]$ whose entries are linear in the four variables, an integer $K \leq \ell^d$, and $r \in \{0, \ldots, d\}$
Output: $(w_0, x_0, y_0, z_0) \in \mathbb{Z}^4$ satisfying $q(w_0, x_0, y_0, z_0) = Lk$, subject to the constraints:

1. $k \leq K$,
2. $L \mid (N_2)^2$,
3. $\gcd(k, \ell) = 1$,
4. $M(w_0, x_0, y_0, z_0)$ has at least one ℓ^{e-d}-eigenvector, and
5. $M(w_0, x_0, y_0, z_0)$ has $\zeta < r$, (where ζ is defined with respect to M and ℓ),

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

Using Algorithm 9.8 and Oracle 9.9 gives us a more general version of Theorem 5.11.

The endomorphisms with many ℓ^d-eigenvectors are actually the same endomorphisms as those with many almost-eigenvectors. The following proposition makes this formal.

Proposition 9.10. *An endomorphism ϕ_C has $\ell^{\nu+\xi}$ (or $2\ell^{\nu+\xi}$ if $\ell \nmid \beta'$), many ℓ^e-eigenvectors if and only if it has $\ell^{\nu+\xi+e-d}$ (or $2\ell^{\nu+\xi+e-d}$ if $\ell \nmid \beta'$) many almost-eigenvector by ℓ^{e-d}, where $\nu + \xi \leq d$ and $d \leq e$.*

Proof. Recall $\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right] \in E[\ell^e]$ is an eigenvector of $\left[\begin{smallmatrix}\alpha & \beta\\ \gamma & \delta\end{smallmatrix}\right]$ if and only if it satisfies Equation (3) from Theorem 6.8, namely,

$$(\delta - \alpha - \beta(2r + x))x \equiv 0 \pmod{\ell^e}. \tag{13}$$

Thus $\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right] \in E[\ell^d]$ is an almost-eigenvector by ℓ^{e-d} of $\left[\begin{smallmatrix}\alpha & \beta\\ \gamma & \delta\end{smallmatrix}\right]$ if and only if it satisfies

$$(\delta - \alpha - \beta(2r + x))x \equiv 0 \pmod{\ell^d}. \tag{14}$$

By Theorem 6.8, if $\ell \mid \beta'$, then $\left[\begin{smallmatrix}1\\\kappa\end{smallmatrix}\right]$ is an almost-eigenvector by ℓ^{e-d} if and only if it has the form $\left[\begin{smallmatrix}1\\r+c\ell^{d-\nu-\xi}\end{smallmatrix}\right]$ for some $c \in \mathbb{Z}$. If $\ell \nmid \beta'$, then $\left[\begin{smallmatrix}1\\\kappa\end{smallmatrix}\right]$ is an almost-eigenvector by ℓ^{e-d} if and only if it has the form $\left[\begin{smallmatrix}1\\r+c\ell^{d-\nu-\xi}\end{smallmatrix}\right]$ or $\left[\begin{smallmatrix}1\\-r+\epsilon'(\beta')^{-1}+c\ell^{d-\nu-\xi}\end{smallmatrix}\right]$ for some $c \in \mathbb{Z}$.

Conversely, following the proof of Theorem 6.8, we get that $x = c\ell^{d-\nu-\xi}$ and $x = -2r + \epsilon'(\beta')^{-1} + c\ell^{d-\nu-\xi}$ are solutions to Equation (14), for any $c \in \mathbb{Z}$. □

Remark 9.11. Proposition 9.10 shows that by using the almost-eigenspace (by ℓ^{e-d}), we essentially increase the number of (almost) eigenvectors by ℓ^{e-d}. Hence, in Theorem 9.7, for endomorphisms whose eigenspace is not quite as large as we wish, by using almost-eigenspaces we can essentially increase the size of the eigenspace by ℓ^{e-d} at the cost of slowing the algorithm down by ℓ^{e-d}. Therefore, in Theorem 5.11 in the case $N_1 = \ell^e$ we can essentially increase the size of the eigenspace by ℓ^{e-d} (as long as $\rho\ell^{e-d} < \frac{1}{2}$), at the cost of increasing the runtime by ℓ^{e-d}.

The results in this section have generalized our techniques in order to increase the space of eigenvectors for an endomorphism. These methods do not decrease the runtime of Theorem 5.11. However, it does accommodate for the possible scenario when the endomorphisms found do not have sufficiently many eigenvectors, even if they have an efficiently computable degree (namely, k is small).

While this section focused on making the attack more applicable, the next section will instead aim to reduce the number of oracle calls to theorems like Theorem 6.12.

10 Learning Secret Torsion Information

Recall that our reductions rely on the repeated use of Theorem 5.7. However, Theorem 5.7 not only determines (with overwhelming probability) whether $R_A \in \mathsf{Eig}_{N_1}(\phi_C)$ or $R_A \notin \mathsf{Eig}_{N_1}(\phi_C)$, but in the case where $R_A \in \mathsf{Eig}_{N_1}(\phi_C)$, it also constructs the map ψ_C.

In this section, we propose a method which makes use of the explicit endomorphism ψ_C on E_A to determine the image of more torsion points under ϕ_A other than the points given in Alice's public key. We will discuss the idea underlying our methods in the simplest case, that is, when the endomorphisms have the easiest form to work with.

Suppose that $\phi_C(\ker \phi_A) = \ker \phi_A$. Recall that, by Theorem 9.3, $[N_1] \cdot \psi_C = \phi_A \circ \phi_C \circ \widehat{\phi}_A$, see Figure 16.

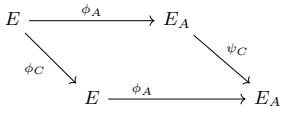

Fig. 16: Commutative diagram where ϕ_C fixes $\ker \phi_A$

Thus we expect ψ_C to act on the torsion points of E in a similar manner to how ϕ_C acts on the torsion points of E_A. We will start by showing the (well-known) fact that $\phi_A : E \to E_A$ takes a basis for $E[N]$ to a basis of $E_A[N]$ (assuming $\gcd(N, p) = \gcd(N, N_1) = 1$).

Lemma 10.1. *Let $\phi : E \to E_A$ be a separable isogeny and $\{P, Q\}$ be a basis for $E[N]$, where N, $\deg \phi$ and p are pairwise coprime, then $E_A[N] = \langle \phi(P), \phi(Q) \rangle$.*

Proof. Recall $E[N] \cong \mathbb{Z}/N\mathbb{Z} \times \mathbb{Z}/N\mathbb{Z}$, hence $|E[N]| = N^2$. As $|E[N]|$ and $\deg \phi$ are relatively prime, $\ker \phi \cap E[N] = \{\mathcal{O}\}$. By the first isomorphism theorem on the restriction map $\phi|_{E[N]}$, we find that

$$\operatorname{Im} \phi|_{E[N]} \cong \frac{E[N]}{\ker \phi|_{E[N]}} \cong E[N].$$

There is only one subgroup of E_A of this form, namely $E_A[N]$. Thus, $\phi(E_A[N]) \cong E_A[N]$. Therefore, the isogeny ϕ restricted to $E[N]$ acts as an (isomorphic) linear transformation to $E_A[N]$. The result follows. □

In Proposition 10.2 we show that (under the same conditions as in Lemma 10.1), the action of ϕ_C on $E[N]$ is similar (that is, conjugate as a linear transformation) to the action of ψ_C on $E[N]$. This is depicted in Figure 17 (where $\gcd(\deg \phi_C, p \cdot N \cdot N_1) = 1$).

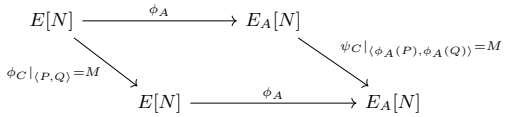

Fig. 17: Action of ϕ_C and ψ_C on torsion subgroups

Proposition 10.2. *Suppose the following:*

1. $\{P, Q\}$ *is a basis for* $E[N]$.
2. $\phi_A \circ \psi_C = \phi_C \circ \phi_A$.
3. $N \geq 2$, *and* N_1, N, p, *and* $\deg \phi_C$ *are all pairwise coprime.*

If $\phi_C|_{\langle P,Q \rangle} = M$ *for some matrix* $M \in \mathrm{GL}_2(\mathbb{Z}/N\mathbb{Z})$, *then*

$$\psi_C|_{\langle \phi_A(P), \phi_A(Q) \rangle} = M.$$

Proof. By Lemma 10.1, we know that $E_A[N] = \langle \phi_A(P), \phi_A(Q) \rangle$. We demonstrate that $\psi_C|_{\langle \phi_A(P), \phi_A(Q) \rangle} = M$. Let

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

Then,

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

Similarly,

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

We conclude that

$$\phi_A \circ \phi_C \circ \widehat{\phi}_A\Big|_{\langle \phi_A(P), \phi_A(Q) \rangle} = (N_1)M.$$

Since the diagram in Figure 16 commutes, we have

$$\phi_A \circ \phi_C = \psi_C \circ \phi_A,$$

$$\phi_A \circ \phi_C \circ \widehat{\phi}_A = \psi_C \circ \phi_A \circ \widehat{\phi}_A,$$

and since $[N_1] = \phi_A \circ \widehat{\phi}_A$ we conclude that

$$\phi_A \circ \phi_C \circ \widehat{\phi}_A = \psi_C \circ [N_1].$$

Therefore,

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

Since N_1 is invertible modulo N, we can divide both sides of the above equality by N_1 and obtain the desired result. □

Proposition 10.2 says that if ϕ_C preserves $\ker \phi_A$, and ϕ_A maps a basis $\langle P, Q\rangle$ to a basis $\langle \phi_A(P), \phi_A(Q)\rangle$, then the matrix of ϕ_C with respect to $\langle P, Q\rangle$ equals the matrix of ψ_C with respect to $\langle \phi_A(P), \phi_A(Q)\rangle$. However, we are interested in the converse direction:

- Does the equation $\phi_C|_{\langle P,Q\rangle} = \psi_C|_{\langle P',Q'\rangle}$ reveal any information about the relationship between $\langle P, Q\rangle$ and $\langle P', Q'\rangle$?
- Furthermore, can we use this relationship to deduce information about the action of ϕ_A on $E[N]$?

We answer these questions in the affirmative in the case where the action is given as a diagonal matrix, see Figure 18.

Proposition 10.3. *Suppose the following:*

1. *we are given an endomorphism ϕ_C on E, with* $\gcd(\deg \phi_C, pN_1) = 1$,
2. $\phi_C(\ker \phi_A) = \ker \phi_A$, *and*
3. *the action of ϕ_C on $E[N]$ with respect to a basis $\{P, Q\}$ is a diagonal matrix D (which is not a scalar multiple of the identity) for some natural number N, where* $\gcd(N, pN_1 \deg \phi_C) = 1$.

Then we can determine $\phi_A(P)$ and $\phi_A(Q)$ up to scalar multiples.

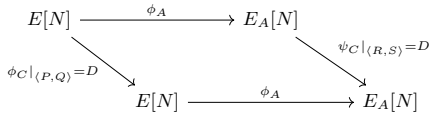

Fig. 18: Action of ϕ_C and ψ_C on torsion subgroups

Proof. Since $\phi_C(\ker \phi_A) = \ker \phi_A$, ψ_C exists by Proposition 4.7. By Proposition 10.2, the action of $\phi_C|_{E[N]}$ is diagonalizable, and moreover, $\psi_C|_{E_A[N]}$ is similar (as a matrix) to D. Thus, there exists a basis $\{R, S\}$ of $E_A[N]$ such that $\psi_C|_{E_A[N]}$ with respect to $\{R, S\}$ is also D.

Let $D = \left[\begin{smallmatrix} d_1 & 0 \\ 0 & d_2 \end{smallmatrix}\right]$. Proposition 10.2 states that ψ_C acts as D with respect to the basis $\{\phi_A(P), \phi_A(Q)\}$. This means that $\phi_A(P)$ and $\phi_A(Q)$ are eigenvectors of D of eigenvalue d_1 and d_2, respectively. But we also have that, R and S are eigenvectors of D of eigenvalue d_1 and d_2, respectively. This implies $\phi_A(P) = [\alpha] \cdot R$ and $\phi_A(Q) = [\beta] \cdot S$ for some $\alpha, \beta \in (\mathbb{Z}/N\mathbb{Z})^*$.

□

Proposition 10.3 tells us that we can almost learn the action of ϕ_A on arbitrary subgroups $E[N]$ (under some reasonable coprimality restrictions of N) when there is a basis for which ϕ_C acts as a diagonal matrix. The following Remark 10.4 shows that it is easy to choose a torsion group $E[N]$ for some basis such that ϕ_C acts as a diagonal matrix.

Remark 10.4. Given a random matrix M , it is often possible to find a number N such that the reduction of M modulo N is similar to a non-scalar diagonal matrix. More specifically, suppose there exists N that divides a non-diagonal entry and the diagonal entries are not equal mod N. Let M' denote the reduction of M modulo N. By our choice of N, the matrix M' is triangular, and the diagonal entries are not equal. As the diagonal entries are not equal, this triangular matrix can be diagonalized.

To our knowledge, ψ_C alone is not enough to uniquely determine $\phi_A|_{E[N]}$, (even if we use techniques involving the Weil pairing [28], as is done in the proof of the next proposition). However, in certain cases, two endomorphisms ψ_C and ψ'_C are enough to almost uniquely determine $\phi_C|_{E[N]}$, which we discuss now.

Proposition 10.5. *Suppose the following:*

1. *we are given two endomorphisms ϕ_C and ϕ'_C on E,*
2. *$\phi_C(\ker \phi_A) = \ker \phi_A$ and $\phi'_C(\ker \phi_A) = \ker \phi_A$,*
3. *ϕ_C acts on $E[N]$ as a diagonal matrix D with respect to a basis $\{P, Q\}$,*
4. *ϕ'_C acts on $E[N]$ as a diagonal matrix D' with respect to a basis $\{P', Q'\}$,*
5. *$\deg \phi_C, \deg \phi'_C$ each are pairwise coprime with N, N_1 and p, and*
6. *$P' = [\gamma] \cdot P + [\delta] \cdot Q$ for some integers $\gamma, \delta \in (\mathbb{Z}/N\mathbb{Z})^*$.*

Then we can determine $(\phi_A(P), \phi_A(Q))$ up to a sign.

Proof. We describe the algorithm that outputs $(\phi_A(P), \phi_A(Q))$ up to sign given the conditions as described in Proposition 10.5 followed by a justification of its correctness and analysis of its runtime.

Algorithm 10.6.
Input: E, E_A, N, bases $\{P, Q\}, \{P', Q'\}$ of $E[N]$, endomorphisms ψ_C, ψ'_C with action on $E_A[N]$ equal to D, D' respectively.
Output: $\pm(\phi_A(P), \phi_A(Q))$.

1. Find a basis $\{R, S\}$ of $E_A[N]$ such that the action of ψ_C on $E_A[N]$ with respect to a basis $\{R, S\}$ is D.

2. Find a basis $\{R', S'\}$ of $E_A[N]$ such that the action of ψ'_C on $E_A[N]$ with respect to a basis $\{R', S'\}$ is D'.
3. Determine γ and δ such that $P' = [\gamma] \cdot P + [\delta] \cdot Q$.
4. Without loss of generality (by switching R' and S' if necessary) it is possible to find γ' and $\delta' \in (\mathbb{Z}/N\mathbb{Z})^*$ such that $R' = [\gamma'] \cdot R + [\delta'] \cdot S$.
5. Evaluate the Weil pairings $g = e_N(P, Q)$ and $h = e_N(R, S)$.
6. Solve the discrete log for x in $g^{N_1} = h^x$.
7. Solve $\alpha^2 \equiv \frac{\delta\gamma'}{\delta'\gamma} x \pmod{N}$ for α.
8. Set $\beta \equiv \alpha^{-1} x \pmod{N}$.
9. Return $\phi_A(P) = [\alpha] \cdot R$, $\phi_A(Q) = [\beta] \cdot S$.

We now justify the correctness of our algorithm. Observe that,

$$\phi_A(P') = [\gamma] \cdot \phi_A(P) + [\delta] \cdot \phi_A(Q) = [\gamma\alpha] \cdot R + [\delta\beta] \cdot S,$$

and also, by Proposition 10.3, there is some integer ϵ such that

$$\phi_A(P') = [\epsilon] \cdot R' = [\epsilon\gamma'] \cdot R + [\epsilon\delta'] \cdot S.$$

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:

$$e_N(\phi_A(P), \phi_A(Q)) = e_N(P, Q)^{N_1} = g^{N_1},$$

and,

$$e_N(\phi_A(P), \phi_A(Q)) = e_N(R, S)^{\alpha\beta} = h^x.$$

Therefore, $x = \alpha\beta$ and can be computed by solving discrete log as described in Step 6. Substituting for β from the previous equation in terms of α, we get the expression for α^2 in Step 7. Hence, we obtain two solutions $\pm(\alpha, \beta)$ and the result follows. □

Intuitively, Proposition 10.5 states that if two endomorphisms of E are found that each fix the subgroup $\ker \phi_A$, then it is possible to learn the action of ϕ_A up to a sign on any torsion subgroup where the hypotheses of Proposition 10.5 are satisfied. In particular, either the pair $(\phi_A(P), \phi_A(Q))$ or the pair $(-\phi_A(P), -\phi_A(Q))$ is found.

We now present a procedure to relax the assumption of Oracle 6.11.

Oracle 10.7.
Input: $\phi_C, \phi'_C \in \mathrm{End}(E)$ and N as in the statement of Proposition 10.5, and the input to Oracle 6.11.
Output: The same output of Oracle 6.11, except Step 2 is replaced with

$$L \mid (N \cdot N_2)^2.$$

Theorem 10.8. *The statement of Theorem 6.12 holds when Oracle 6.11 is replaced with Oracle 10.7 when N is $\log p$-smooth.*

Proof. We need to show that given the input to Oracle 10.7 and access to Oracle 6.11, we can construct the output to Oracle 10.7.

By Proposition 10.5 we can construct the pair $\pm(\phi_A(P), \phi_A(Q))$ for basis $\{P, Q\}$ of $E[N]$. Thus we are interested in endomorphisms with triangular kernels with respect to NN_2 (instead of triangular kernels with respect to N_2). Next, similar to the proof of Proposition 6.3, we can transform the quadratic form solution into a triangular kernel, but now say with NN_2-torsion components $K_{N,1}$ and $K_{N,0}$. Traversing through the reductions, we see that we must modify Algorithm 5.8 to accommodate for the new factor of N. Consider the following modified Step 3 of Algorithm 5.8:

3.′ Compute the isogenies $\psi_{N,0}$ and $\psi_{N,1}$ with respective kernels $\langle \psi_{C,0} \circ \phi_A(K_{N,0}) \rangle$ and $\langle \psi_{C,1} \circ \phi_A(K_{N,1}) \rangle$, and set codomain curves to be $E_{CNA,0}$ and $E_{CNA,1}$, respectively. For all k-isogenies from $E_{CNA,0}$, check if their codomain has j-invariant $j(E_{CNA,1})$.

□

If the torsion images were found completely (instead of only up to sign), then this approach could be combined with [17] to efficiently solve the CSSI problem. We leave this for future work.

11 Recommendations

The work given in this paper, and by Petit in [17], demonstrate that the hardness assumptions of specific instances of the CSSI problem differ from the ℓ^e-isogeny problem. In particular, by fixing a starting elliptic curve, SIKE and most of the other protocols also fix a particular instantiation of the CSSI problem with a known endomorphism ring.

The best use of our methods on the ℓ^e-isogeny problem is given in Theorem 4.9, but Remark 4.10 demonstrates some of its limitations. However, Theorem 5.11 gives a more practical result; it reduces the CSSI problem to the problem of finding desirable endomorphisms. Our next main result, Theorem 6.12, reduces a particular instantiation of the CSSI problem (at a supersingular elliptic curve with known endomorphism ring) to the problem of finding solutions to a simple quadratic form.

In [10], the authors prove that if endomorphism rings of both the starting curve and ending curve are known, then it is possible to construct an isogeny between them. Knowing only the endomorphism ring of the starting curve is not known to decrease the security in the case of SIKE, but even without concrete attacks, being more conservative is not unwarranted.

One solution to thwart any potential attacks exploiting this knowledge is to initialize the protocol with a supersingular elliptic curve with an unknown endomorphism ring. However, known methods for constructing a supersingular

elliptic curve also give a description of its endomorphism ring, so it is not immediately clear how to effectively modify SIKE to start with a *fixed* supersingular curve with an unknown endomorphism ring. For example, using an isogeny path from a supersingular elliptic curve with a known endomorphism ring gives the endomorphism ring of the new supersingular elliptic curve (at least to whomever computed the isogeny).

We propose to slightly alter the SIKE protocol, by requiring Alice to randomize the starting curve each time by taking a random walk from the initial SIKE curve as her first step. In the spirit of many recent works in post-quantum cryptography, we humorously propose the name *Supersingular Isogeny Two-party Handshake* (SITH) for our variant. By performing this change, the endomorphism ring of the new starting curve is only known to Alice. Figure 19 illustrates the steps for this new algorithm.

To prevent a GPST-style attack, SIKE uses the Fujisaki-Okamoto transform, which requires the other party, Bob, to divulge his encryption randomness seed to Alice, see [16]. As a result, it does not matter if Alice has additional secret information about the starting elliptic curve E_R, as Alice has no interest in attacking Bob (since she already obtains all of his secret entropy). More importantly, in our proposed modification, neither Bob nor any eavesdroppers have any information about the endomorphism ring of E_R.

11.1 Supersingular Isogeny Two-party Handshake (SITH)

Figure 19 depicts the SITH proposal. The only differences between SITH and SIKE are the key generation procedure and the resulting public key. We briefly present these differences.

For key generation, Alice now computes an isogeny ϕ_R whose kernel is in $E[2^a 3^b]$, and whose degree is large enough to ensure cryptographic security. Then she computes $E_R = \phi_R(E)$, and bases $\langle P_A, Q_A \rangle = E_R[2^a]$, and $\langle P_B, Q_B \rangle = E_R[3^b]$.

After this, Alice chooses a random number $0 \leq r_A < 2^a$. She calculates the isogeny ϕ_A on E_R whose kernel is generated by $P_A + [r_A] \cdot Q_A$ and also calculates $E_A = \phi_A(E_R), \phi_A(P_B)$ and $\phi_A(Q_B)$.

More specifically, key generation now involves the computation of

$$\phi_R(E), P_A, Q_A, P_B, Q_B, \phi(E_R), \phi_A(P_B), \phi_A(Q_B)$$

instead of $\phi_A(E)$, $\phi_A(P_B)$, $\phi_A(Q_B)$ as in SIKE.

Alice's public key is now

$$(E_R, P_A, Q_A, P_B, Q_B, E_A, \phi_A(P_B), \phi_A(Q_B))$$

instead of $E_A, \phi_A(P_B), \phi_A(Q_B)$, namely more than twice as long as in SIKE.

Note that there is a time/memory trade-off that can be considered with respect to key generation. In the version we presented, Alice sends the generated points P_A, Q_A, P_B, Q_B. Naturally, these points could also be independently generated by Bob. The best option depends on the computational resources of the entities running such a protocol.

The rest of the steps, namely, key generation for Bob, encapsulation and decapsulation are the same as in SIKE, with the starting curve being now E_R instead of E.

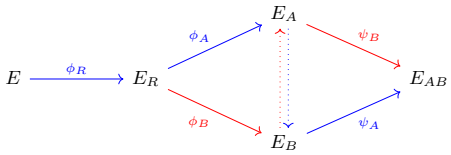

Fig. 19: Supersingular Isogeny Two-party Handshake (SITH)

12 Conclusion and Future Work

In this work, we have investigated a connection between solving the Computational Supersingular Isogeny (CSSI) problem and finding endomorphisms of a certain type of degree with large eigenspaces. While endomorphisms have been exploited in cryptanalytic efforts before (see [17]), our methods are fundamentally different from those used previously.

We presented a generic reduction from SIDH-based protocols with arbitrary number of parties to an oracle which returned endomorphisms with large eigenspace and certain bounded degree. As part of this reduction, we have introduced the notion of triangular decomposition of endomorphisms to circumvent relying on large extension fields, which may prove useful in future cryptanalytic efforts on isogeny-based cryptosystems.

More specifically, our approach exploits endomorphisms whose degrees have a large common divisor with N_1^2, and whose eigenspaces have a large intersection with the space of possible private keys. In the case where the endomorphism ring of the starting curve is known, we have shown that finding these specific endomorphisms is equivalent to finding solutions to a particular quadratic form. In certain cases, instead of solving the degree and eigenspace conditions separately, the eigenspace conditions can be combined to produce another quadratic form (whose solutions give us desirable endomorphisms).

In the case of 2-party SIDH/SIKE, we have shown that our reduction could not yield an attack that takes sub-exponential time. However, in Section 8, we have provided heuristic evidence that our reduction may yield faster attacks on multi-party CSSI-based cryptosystems than demonstrated previously – even for a small number of parties (possibly as small as 3 or 4). The relationship between quadratic forms and the security of CSSI-based cryptosystems (exploiting torsion images) had been investigated before in [17], but Petit's methods are not directly applicable to small multi-party cases, unless extra torsion information is revealed.

We also examined the tradeoff between the runtime of our reduction and the amount of information each desirable endomorphism provides. Finally, we presented a method of learning the images of the private isogeny on additional torsion points, which meant that our reduction required fewer desirable endomorphisms.

Based on our investigations, our recommendation is to avoid using starting curves in supersingular isogeny-based key establishments with known endomorphism rings, which mitigates any future attack based on our approach or Petit's.

Since we do not know of an efficient method to select a (universal) starting curve whose endomorphism ring is completely unknown, we recommend that the first step during key generation in SIKE, and related protocols, be changed to choosing a new starting curve. After this, the remaining steps of the key establishment can proceed unchanged.

12.1 Future Work

The avenue of attack explored in this work is far from complete, and leaves much to be explored and understood. Broadly speaking, it is important to understand the potentials and limitations of exploiting the knowledge of the endomorphism ring of the starting elliptic curve together with the knowledge of the action of an isogeny on a torsion subgroup. A complete investigation of these elements is necessary to thoroughly understand the security of supersingular isogeny-based cryptosystems. Specifically, the major elements we leave for future work are:

- Find endomorphisms with a certain type of degree with a large percentage of eigenvectors, in the unbalanced/multi-party cases. This amounts to finding solutions to $w^2 + Dx^2 + Dp(y^2 + z^2) = Lk$ for a large divisor D of N_1 and large divisor L of N_2^2. We have made a heuristic argument as to why we believe such solutions exist when there are at least 4-parties, but have no concrete solutions.
- If such endomorphisms can be found, investigate the distribution of the eigenspaces.
- Investigate how this attack would work in the unbalanced case for computationally feasible cases, where the prime p is small enough to work.
- Determine, for a fixed prime p, if there are certain particular starting elliptic curves that are more or less susceptible to our methods of attack. That is, what is the most appropriate choice of starting curve in a 2-party key exchange assuming we wish to have a fixed starting curve?
- Is there a more efficient reduction from SSDDH to finding solutions to our quadratic form Q. Perhaps combining these methods with other works, for example, that of Petit [17], could yield new cryptanalytic results.
- An in-depth comparison of sizes and performance between SIKE and SITH.

Acknowledgments. We thank Elena Bakos Lang, Edward Eaton, and Daniela Maftuleac for their helpful suggestions.

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. In Carlos Cid and Michael J. Jacobson Jr., editors, *Selected Areas in Cryptography – SAC 2018*, pages 322–343. Springer International Publishing, 2019.

2. Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, Carl Miller, Dustin Moody, Rene Peralta, Ray Perlner, Angela Robinson, Daniel Smith-Tone, and Yi-Kai Liu. NISTIR 8240 Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process. Technical report, National Institute of Standards & Technology, January 2019. `https://doi.org/10.6028/NIST.IR.8240`.
3. Reza Azarderakhsh, Amir Jalali, David Jao, and Vladimir Soukharev. Practical Supersingular Isogeny Group Key Agreement. *IACR Cryptology ePrint Archive*, 2019:330, 2019.
4. Jean-François Biasse, David Jao, and Anirudh Sankar. A Quantum Algorithm for Computing Isogenies between Supersingular Elliptic Curves. In Willi Meier and Debdeep Mukhopadhyay, editors, *Progress in Cryptology – INDOCRYPT 2014*, Lecture Notes in Computer Science, pages 428–442. Springer International Publishing, 2014.
5. Denis X Charles, Kristin E Lauter, and Eyal Z Goren. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113, 2009.
6. Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, and Daniel Smith-Tone. NISTIR 8105 Report on Post-Quantum Cryptography. Technical report, National Institute of Standards & Technology, April 2016. `https://doi.org/10.6028/NIST.IR.8105`.
7. Andrew Childs, David Jao, and Vladimir Soukharev. Constructing elliptic curve isogenies in quantum subexponential time. *Journal of Mathematical Cryptology*, 8(1):1–29, 2014.
8. Giuseppe Cornacchia. Su di un metodo per la risoluzione in numeri interi dellequazione $\sum_{h=0}^{n} C_h x^{n-h} y^h = P$. *Giornale di Matematiche di Battaglini*, 46:33–90, 1908.
9. Jean Marc Couveignes. Hard homogeneous spaces. *IACR Cryptology ePrint Archive*, 2006:291, 2006.
10. Kirsten Eisenträger, Sean Hallgren, Kristin Lauter, Travis Morrison, and Christophe Petit. Supersingular Isogeny Graphs and Endomorphism Rings: Reductions and Solutions. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018*, Lecture Notes in Computer Science, pages 329–368. Springer International Publishing, 2018.
11. Steven D Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 63–91. Springer, 2016.
12. David Jao, Reza Azarderakhsh, Matthew Campagna, Craig Costello, Amir Jalali, Brian Koziel, Brian LaMacchia, Patrick Longa, Michael Naehrig, Joost Renes, et al. Supersingular isogeny key encapsulation november 30, 2017. *NIST Round 1 Submissions for Post-Quantum Cryptography Standardization*, 2017.
13. 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.
14. 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*, Lecture Notes in Computer Science, pages 32–61. Springer International Publishing, 2019.
15. Kiran S. Kedlaya and Christopher Umans. Fast polynomial factorization and modular composition. *SIAM J. Comput.*, 40:1767–1802, 2008.

16. 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. In *Talk at NIST workshop on Cybersecurity in a Post-Quantum World: http://www. nist. gov/itl/csd/ct/post-quantum-crypto-workshop-2015. cfm*, volume 2, 2015.
17. Christophe Petit. Faster algorithms for isogeny problems using torsion point images. In *Advances in Cryptology – ASIACRYPT 2017*, pages 330–353. Springer, 2017.
18. Arnold Pizer. Ramanujan graphs. *AMS IP STUDIES IN ADVANCED MATHEMATICS*, 7:159–178, 1998.
19. Stephen Pohlig and Martin Hellman. An improved algorithm for computing logarithms over gf (p) and its cryptographic significance (corresp.). *IEEE Transactions on information Theory*, 24(1):106–110, 1978.
20. Alexander Rostovtsev and Anton Stolbunov. Public-key cryptosystem based on isogenies. *IACR Cryptology ePrint Archive*, 2006:145, 2006.
21. Joseph H Silverman. *The arithmetic of elliptic curves*, volume 106. Springer Science & Business Media, 2009.
22. Anton Stolbunov. Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves. *Adv. in Math. of Comm.*, 4(2):215–235, 2010.
23. Andrew Sutherland. On the evaluation of modular polynomials. *The Open Book Series*, 1(1):531–555, Nov 2013.
24. Seiichiro Tani. Claw finding algorithms using quantum walk. *Theoretical Computer Science - TCS*, 410, 08 2007.
25. David Urbanik and David Jao. New techniques for SIDH-based NIKE. *MathCrypt 2018, to be published in the Journal of Mathematical Cryptology*, 2018.
26. 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.
27. Jacques Vélu. Isogénies entre courbes elliptiques. *CR Acad. Sc. Paris.*, 273:238–241, 1971.
28. André Weil. Sur les fonctions algébriquesa corps de constantes fini. *CR Acad. Sci. Paris*, 210(592-594):149, 1940.

A Appendix

A.1 SIKE

The most important isogeny-based algorithm is SIKE, a KEM that was submitted to the NIST call for quantum-safe algorithms. Let $\mathbb{F}_q$ denote a finite field with q elements. Let E be an elliptic curve over $\mathbb{F}_q$.

Suppose that Alice and Bob would like to use SIKE to agree on a secret key. Refer to Figure 20 when reading the SIKE algorithm.

Public Parameters: Let $p = 2^a 3^b - 1$. Let E be a supersingular elliptic curve with j-invariant 1728. This is the starting curve for the NIST Round 1 version of SIKE. The Round 2 version uses a starting curve that is adjacent to this one on the 2-isogeny graph.

Key Generation: Alice chooses a random number $0 \leq r_A < 2^a$. She calculates the isogeny ϕ_A on E whose kernel is generated by $P_A + [r_A] \cdot Q_A$. She also calculates $E_A = \phi_A(E), \phi_A(P_B)$ and $\phi_A(Q_B)$.

Public Key: $E_A, \phi_A(P_B), \phi_A(Q_B)$.
Encapsulation: Similarly, Bob chooses a random $0 \leq r_B < 3^b$. He calculates $E_B = \phi_B(E), \phi_B(P_A)$ and $\phi_B(Q_A)$, where ϕ_B is an isogeny on E whose kernel is generated by $P_B + [r_B] \cdot Q_B$.

Bob also calculates the image E_{BA} of the isogeny ψ_B on E_A with kernel generated by $\phi_A(P_B) + [r_B] \cdot \phi_A(Q_B)$. He calculates the j-invariant of E_{BA}, which we denote j_{BA}, and uses this j-invariant to encrypt r_B.
Encapsulated Key: $E_B, \phi_B(P_A), \phi_B(Q_A), \mathrm{Enc}_{j_{AB}}(r_B)$.
Decapsulation: Alice computes the isogeny ψ_A on E_B whose kernel is generated by $\phi_B(P_A) + [r_A] \cdot \phi_B(Q_A)$. She calculates the j-invariant of E_{AB}. If both participants performed honestly, then $j(E_{AB}) = j(E_{BA})$. Alice uses $j(E_{AB})$ to decrypt r_B. She then derives $E'_B, \phi_B(P_A)', \phi_B(Q_A)'$ using r_B and compares them to Bob's encapsulated key to verify his honesty. Both parties can now use a key derivation function on $j(E_{AB})$ to get a shared symmetric key.

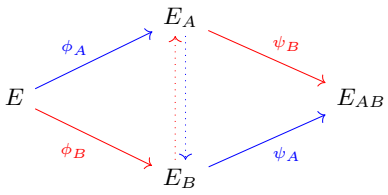

Fig. 20: SIKE

A.2 Proofs

Proof (Corollary 7.19). If Oracle 7.12 outputs endomorphisms satisfying only the modular condition, then by Proposition 7.18, the size of the eigenspace is bounded above by $2\ell^{\frac{e}{2}}$. Thus, $\rho \leq \frac{2\ell^{\frac{e}{2}}}{\ell^e} = 2\ell^{-\frac{e}{2}} \leq \frac{1}{2}$. Since H is monotonically increasing in $[0, \frac{1}{2}]$,

$$\begin{aligned} H(\rho) &\leq 2\ell^{-\frac{e}{2}} \log\left(\tfrac{1}{2}\ell^{\frac{e}{2}}\right) + (1 - 2\ell^{-\frac{e}{2}}) \log\left(\frac{1}{1 - 2\ell^{-\frac{e}{2}}}\right) \\ &= 2\ell^{-\frac{e}{2}} \log\left(\tfrac{1}{2}\ell^{\frac{e}{2}}(1 - 2\ell^{-\frac{e}{2}})\right) - \log\left(1 - 2\ell^{-\frac{e}{2}}\right) \\ &= 2\ell^{-\frac{e}{2}} \log\left(\tfrac{1}{2}\ell^{\frac{e}{2}} - 1\right) - \log\left(1 - 2\ell^{-\frac{e}{2}}\right) \end{aligned}$$

Observe that, for $x = \ell^{-\frac{e}{2}}$,

$$\lim_{x \to 0} \left(\frac{-\log(1 - 2x)}{2x \log\left(\frac{1}{2x} - 1\right)} \right) = 0.$$

So, for reasonably large e (such as in the SIKE setting),

$$-\log\left(1 - 2\ell^{-\frac{e}{2}}\right) \leq 2\ell^{-\frac{e}{2}} \log\left(\tfrac{1}{2}\ell^{\frac{e}{2}} - 1\right).$$

Hence, we can bound $H(\rho)$ by

$$\begin{aligned} H(\rho) &\leq 2(2\ell^{-\frac{e}{2}} \log\left(\tfrac{1}{2}\ell^{\frac{e}{2}} - 1\right)) \\ &\leq 4\ell^{-\frac{e}{2}} \log \ell^{\frac{e}{2}} \\ &\leq 2e\ell^{-\frac{e}{2}} \log \ell. \end{aligned}$$

□

Proof (Lemma 7.4). The isogeny $\phi_C \circ \widehat{\phi}_C$ is equal to $[\deg \phi_C]$. We will use the general fact that given isogenies ψ_1, ψ_2, it follows that $\widehat{\psi_1 + \psi_2} = \widehat{\psi}_1 + \widehat{\psi}_2$. Additionally, it is well-known that $\widehat{\iota} = -\iota$ and $\widehat{\pi} = -\pi$. This implies

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

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

Proof (Lemma 7.5). This proof involves basic algebraic manipulations. Suppose a 4-tuple (w_0, x_0, y_0, z_0) is a solution to

$$w^2 + x^2 + \left(\tfrac{p+1}{4}\right)(y^2 + z^2) + wy + xz = 2^m k.$$

As the left-hand side of the equation is a quadratic form, it follows that that $(2w_0, 2x_0, 2y_0, 2z_0)$ is a solution to

$$w^2 + x^2 + \left(\tfrac{p+1}{4}\right)(y^2 + z^2) + wy + xz = 2^{m+2} k.$$

Thus $(2w_0, 2x_0, y_0, z_0)$ is a solution to

$$w^2 + x^2 + 2wy + 2xz + (p+1)(y^2 + z^2) = 2^{m+2} k.$$

By completing the square we see that $(2w_0, 2x_0, y_0, z_0)$ is a solution to

$$(w+y)^2 + (x+z)^2 - y^2 - z^2 + (p+1)(y^2 + z^2) = 2^{m+2} k.$$

Thus $(2w_0 + y_0, 2x_0 + z_0, y_0, z_0)$ is a solution to

$$w^2 + x^2 + p(y^2 + z^2) = 2^{m+2} k.$$

Conversely, suppose (w_0, x_0, y_0, z_0) is a solution to

$$w^2 + x^2 + p(y^2 + z^2) = 2^m k,$$

then it is also a solution to

$$w^2 + x^2 - y^2 - z^2 + (p+1)(y^2 + z^2) = 2^m k.$$

This implies that $(w_0 - y_0, x_0 - z_0, y_0, z_0)$ is also a solution to

$$(w+y)^2 + (x+z)^2 - y^2 - z^2 + (p+1)(y^2 + z^2) = 2^m k.$$

Expanding the brackets implies $(w_0 - y_0, x_0 - z_0, y_0, z_0)$ is a solution to

$$w^2 + x^2 + 2wy + 2xz + (p+1)(y^2 + z^2) = 2^m k.$$

Therefore, $(w_0 - y_0, x_0 - z_0, 2y_0, 2z_0)$ is a solution to

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

Lemma A.1. *Note that* $\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right]$ *is an eigenvector of the matrix* $\left[\begin{smallmatrix}\alpha & \beta\\ \gamma & \delta\end{smallmatrix}\right]$ *over* $\mathbb{Z}/\ell^e\mathbb{Z}$ *if and only if* $\gamma + (\delta - \alpha)r - \beta r^2 \equiv 0 \pmod{\ell^e}$.

Proof. If $\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right]$ is an eigenvector, then

$$\left[\begin{smallmatrix}\alpha & \beta\\ \gamma & \delta\end{smallmatrix}\right]\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right] = \left[\begin{smallmatrix}\beta r+\alpha\\ \delta r+\gamma\end{smallmatrix}\right] = \lambda\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right]$$

for some $\lambda \in \mathbb{Z}/\ell^e\mathbb{Z}$. Thus

$$rs\lambda \equiv \delta r + \gamma \equiv \beta r^2 + \alpha r \pmod{\ell^e},$$

which implies

$$\gamma + (\delta - \alpha)r - \beta r^2 \equiv 0 \pmod{\ell^e}.$$

Conversely, suppose $\gamma + (\delta - \alpha)r - \beta r^2 \equiv 0 \pmod{\ell^e}$, then

$$\delta r + \gamma \equiv \alpha r + \beta r^2 \pmod{\ell^e}.$$

$$\left[\begin{smallmatrix}\alpha & \beta\\ \gamma & \delta\end{smallmatrix}\right]\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right] = \left[\begin{smallmatrix}\beta r+\alpha\\ \delta r+\gamma\end{smallmatrix}\right] = \left[\begin{smallmatrix}\beta r+\alpha\\ \alpha r+\beta r^2\end{smallmatrix}\right] = (\beta r + \alpha)\left[\begin{smallmatrix}1\\r\end{smallmatrix}\right].$$

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

We now propose the conditions under which there are no non-trivial endomorphisms with eigenvectors when $y = z = 0$. This helps to prove Remark 4.11 and our claim prior to Lemma 7.22.

Proposition A.2. *Let* ϕ_C *denote a endomorphism on* E *whose degree is less than* $\frac{p}{4}$. *Let* ℓ *be prime,* $e \in \mathbb{N}^+$. *Assuming* ϕ_C *is non-constant on* $E[\ell^e]$, *then* ϕ_C *has eigenvectors in* $E[\ell^e]$ *if and only if* $\ell \equiv 1 \pmod 4$.

Proof. By Lemma 7.3 we know that

$$\phi_C = [w] \cdot [1] + [x] \cdot \iota + [y] \cdot \tfrac{[1]+\pi}{2} + [z] \cdot \tfrac{\iota+\iota\pi}{2}.$$

By Lemma 7.4 we know that

$$\begin{aligned}\deg \phi_C &= w^2 + x^2 + \left(\tfrac{p+1}{4}\right)(y^2 + z^2) + wy + xz\\ &= \left(w + \frac{y}{2}\right)^2 + \left(x + \frac{z}{2}\right)^2 + \frac{p}{4}(y^2 + z^2).\end{aligned}$$

As $\deg \phi_C < \frac{p}{4}$ we see that $y = z = 0$, and hence $\phi_C = [w] + [x]\iota$. Let $\{P, \iota(P)\}$ be a basis for $E[\ell^e]$. Since $\iota^2(P) = -P$, ϕ_C on $E[\ell^e]$ acts as $\left[\begin{smallmatrix}w & -x\\ x & w\end{smallmatrix}\right]$ with respect to $\{P, \iota(P)\}$.

To find the eigenvectors notice that the characteristic polynomial of this matrix is $\lambda^2 - 2w\lambda + w^2 + x^2$. The eigenvalues are

$$\lambda = \frac{2w \pm \sqrt{4w^2 - 4(w^2 + x^2)}}{2} = w \pm x\sqrt{-1}.$$

We conclude that, in order for eigenvectors to exist, the number -1 has to be a quadratic residue modulo ℓ^e. In view of the restriction $\ell^e > 2$, this happens if and only if $\ell \equiv 1 \pmod 4$. □

Proposition A.2 shows that in the SIKE/SIDH setting there are no endomorphisms of small degree that have eigenvectors.

Corollary A.3. *The non-trivial endomorphisms with degree less than $\frac{p}{4}$ have no ℓ^e-eigenvectors in the SIKE/SIDH setting (where $\ell \in \{2,3\}$, as $\ell \not\equiv 1 \pmod 4$).*

Remark A.4. A similar statement to Corollary A.3 holds for almost eigenvectors, (see Definition 9.4 for a definition of almost eigenvectors).

Corollary A.3 means that Theorem 4.9 is not viable against Alice's or Bob's private key, as it will reveal a trivial amount of information.

A.3 Convenient Basis

Lemma A.5 is used in Example A.6 and Remark 7.9, where Example A.6 finds a convenient basis for the torsion group $E_0[3^{239}]$ and is also mentioned in Remark 7.9.

Lemma A.5. *Given a point P on E, then $\pi(P) = \iota P$ if and only if $P = (ui, v(1-i))$ for some $u, v \in \mathbb{F}_p$. Furthermore, this happens exactly when $u - u^3 = -2v^2$ for $u, v \in \mathbb{F}_p$.*

Proof. We start by proving the first statement. 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 $P = (ui, v(1-i))$.

Now, suppose $P = (ui, v(1-i))$ for some $u, v \in \mathbb{F}_p$. Then $\pi(P) = (-ui, v(1+i)) = (-(ui), (i)(v(1-i))) = \iota(P)$.

Next we prove the second statement. 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.

Lastly, suppose $P = (ui, v(1-i))$ for some $u, v \in \mathbb{F}_p$. Then

$$(ui)^3 + (ui) = (v(1-i))^2$$

implies that

$$-2v^2 = u - u^3. \qquad \square$$

Example A.6. (Referenced in Remark 7.9) We will now give an example of a basis of the form $\{P, \iota P\} = \{P, \pi(P)\}$ for the power of 3 that is used in SIKE. Consider the prime $p = 2^{372}3^{239} - 1$.

Let $P = (570556479520931242046188854123607496106560164061907466550384438849978105317427915430702852898083512987673860531431396211577428396652625840972800081121950020629423752030609821533127797086914328692076275754730669931633440054111 \cdot i,$
$170529679711570871948934177887081253029965384490684199363468409357092795616015244936108185015599117514645468695608544218109856248721923973057187123901749477184914154433309844493495718071811551302575552223407602899906055523667 +$
$101841880620577343810288340599797240683974242610583869707527212696975 8$

614506646360156677471154546759610990774292323165816709405572780063041998697334486940233979179268564560348986490401288096337567765953462758389897275959605316 4 · i).

Then P and $\pi(P)$ have order 3^{239} and satisfy the condition of Lemma A.5. This means $\{P, \pi(P)\}$ is a convenient basis for $E_0[3^{239}]$. Thus, by Lemma 7.10 with respect to this basis for $E_0[3^{239}]$, endomorphisms of the form

$$\phi_C = [w] \cdot [1] + [x] \cdot \iota + [y] \cdot \tfrac{[1]+\pi}{2} + [z] \cdot \tfrac{\iota+\iota\pi}{2}.$$

act as the matrix $\left[\begin{smallmatrix} w-z & -x+y \\ x+y & w+z \end{smallmatrix}\right]$ on $E_0[3^{239}]$. ◇

A.4 Solutions to Quadratic Equations

As mentioned in Remark 7.7, we can find endomorphisms by using Cornacchia's algorithm [8] to find solutions of the quadratic form in the SIKE setting.

Example A.7. (Referenced in Remark 7.7) Recall the prime used in SIKEp751 is $p = 2^{372}3^{239} - 1$. Given $p = 2^{372}3^{239} - 1$. Let

$$f(w, x, y, z) = w^2 + x^2 + \tfrac{p+1}{4}(y^2 + z^2) + wy + xz.$$

Using Cornacchia's algorithm [8] we found the solutions

$$f(w, x, y, z) = 2^{744}\,505$$

given by $y = 2, z = 0$,
$w = 190459273013886173789469058477389405072977989329809313036300027049913856737274269267420579193501276882631488221299$,
$x = 100934992803712252583499232003067684289983166377831353212224870329353965200785511814716526736324547178269212199 43$, and

$$f(w, x, y, z) = 3^{471}\,797$$

given by $y = 6, z = 2$,
$w = 4402443926454812109222547097306447497601705377932157897814852291811702805998992399170173986835853375357979497667429$,
$x = 353394034376007397128400742894114538724618805835457021608486949136846529962267603900382470795510648299515992269154$.
However, the associated endomorphisms do not have a large number of eigenvectors, hence they are not useful in Theorem 6.12. ◇

We now give an example of an endomorphism satisfying only the modular condition.

Example A.8. (Referenced in Remark 7.17) In the case of the SIKEp751 prime $p = 2^{372}3^{239} - 1$, it is reasonably easy to find matrices of the form $\left[\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right]$, where $3 \nmid \gamma, \delta - \alpha, \beta$, that have an eigenvector $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$, and satisfy the modular condition $(\delta - \alpha) - 2\beta r \equiv 0 \left(\mathrm{mod}\ 3^{\frac{240}{2}}\right)$. We found a matrix whose entries are $\alpha = 0, \gamma = 1, \delta = 1$, and
$\beta = 807311504499638501878806209128340478675593263006775463754863142944360806423365184661810647285478547377694955793200$
with an eigenvector $\left[\begin{smallmatrix} 1 \\ r \end{smallmatrix}\right]$ where

$r = 10764153393328513358384082788377873049007910173423672850064841905$
$924810752311535795490808630473047298369266077242 65.$
Solving $\begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} = \begin{bmatrix} w-z & -x+y \\ x+y & w+z \end{bmatrix}$ for w, x, y, z gives an endomorphism $\phi_C = [w] \cdot [1] + [x] \cdot \iota + [y] \cdot \frac{[1]+\pi}{2} + [z] \cdot \frac{\iota+\iota\pi}{2}$ with $3^{\frac{238}{2}}$ eigenvectors. Although this endomorphism has many eigenvectors, it does not have a high proportion of eigenvectors. Furthermore, for this example k is also impractically large, as $k = 269103834833212833959602069709446826225197754335591821251621047648120268807788394887270215761826182459231651931067.$ $\diamond$