One-way functions and malleability oracles

Kutas, Péter; Merz, Simon-Philipp; Petit, Christophe; Weitkaemper, Charlotte

DOI:
10.1007/978-3-030-77870-5_9

Document Version
Peer reviewed version

Citation for published version (Harvard):
Kutas, P, Merz, S-P, Petit, C & Weitkaemper, C 2021, One-way functions and malleability oracles: hidden shift attacks on isogeny-based protocols. in A Canteaut & F-X Standaert (eds), *Advances in Cryptology – EUROCRYPT 2021: 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17–21, 2021, Proceedings, Part I.* Lecture Notes in Computer Science, vol. 12696, Springer, pp. 242-271, 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, 17/10/21. https://doi.org/10.1007/978-3-030-77870-5_9

One-way functions and malleability oracles: Hidden shift attacks on isogeny-based protocols

Péter Kutas[1], Simon-Philipp Merz[2], Christophe Petit[3,1], and Charlotte Weitkämper[1]

[1] University of Birmingham, UK
[2] Royal Holloway, University of London, UK
[3] Université libre de Bruxelles, Belgium

Abstract. The supersingular isogeny Diffie-Hellman (SIDH) is a post-quantum key exchange protocol based on the presumed hardness of computing an isogeny between two supersingular elliptic curves given some additional torsion point information. Unlike other isogeny-based protocols, SIDH has been widely believed to be immune to subexponential quantum attacks because of the non-commutative structure of the endomorphism rings of supersingular curves.

We contradict this belief in this paper. More precisely, we highlight the existence of an abelian group action on the SIDH key space, and we show that for sufficiently *unbalanced* and *overstretched* SIDH parameters, this action can be efficiently computed using the torsion point information revealed in the protocol. This reduces the underlying hardness assumption to an instance of the hidden shift problem which can be solved in quantum subexponential time.

We formulate our attack in a new framework allowing the inversion of one-way functions in quantum subexponential time provided a malleability oracle with respect to some commutative group action. This framework unifies our new attack with earlier subexponential quantum attacks on isogeny-based protocols, and it may be of further interest for cryptanalysis.

1 Introduction

The hardness of solving mathematical problems such as integer factorization or the computation of discrete logarithms in finite fields and elliptic curve groups guarantees the security of most currently deployed cryptographic protocols. However, these classical problems can be solved efficiently using quantum algorithms. Quantum computers with sufficient processing power to threaten cryptographic primitives currently in use do presumably not yet exist, but progress is being made in quantum computing. The possibility of large scale quantum computers and the need for long-term security in some applications necessitate the development of quantum-secure cryptographic algorithms.

Different approaches to attain quantum-resistance are based on lattices, codes, multivariate polynomials over finite fields, and elliptic curve isogenies. Within the

field of post-quantum cryptography, isogeny-based cryptography is a relatively new area which is particularly interesting due to the small key sizes required. The main underlying problem in this branch of post-quantum cryptography is to find an isogeny $\varphi : E_1 \to E_2$ between two given isogenous elliptic curves E_1 and E_2 over some finite field $\mathbb{F}_q$.

An early isogeny-based cryptographic system utilizing *ordinary* elliptic curves was proposed by Couveignes, but at first only circulated privately [9]. Meanwhile, the first construction using *supersingular* curves was a hash function developed by Charles, Lauter and Goren [6]. Later, Rostovtsev and Stolbunov independently rediscovered and further developed Couveignes' construction [29]. In 2010, Childs, Jao and Soukharev [7] showed how to break this scheme in quantum subexponential time using a reduction to an instance of the injective abelian hidden shift problem. While this attack is tolerable for sufficiently large parameters, the main drawback of the Couveignes-Rostovtsev-Stolbunov (CRS) construction is its unacceptable lack of speed. Adapting the CRS scheme to supersingular elliptic curves, Castryck et al. managed to eliminate most of the performance issues allowing for larger practical parameters when introducing CSIDH [5].

The attack due to Childs, Jao and Soukharev crucially relies on the commutativity of the ideal class groups acting on the endomorphism rings of the relevant elliptic curves over $\mathbb{F}_q$. This motivated Jao and De Feo [18] to consider the full isogeny graph for supersingular elliptic curves whose endomorphism rings are maximal orders in a quaternion algebra (in particular, the endomorphism rings are non-commutative). The result of their work, the *Supersingular Isogeny Diffie-Hellman* (SIDH) key agreement scheme, underlies the SIKE submission to NIST's post-quantum standardization process [1, 17].

The hard problem SIDH is based on is to find an isogeny between two isogenous curves, further given the images of certain torsion points under this isogeny. The supply of this additional public information has fueled cryptanalytic research which aims to recover secret information when parameters are sufficiently overstretched [4, 22, 26]. However, folklore widespread amongst cryptographers assumes that due to SIDH's non-commutative nature there is no quantum attack reducing the SIDH problem to an abelian hidden shift problem. In particular, many believe that no reasonable variant of Childs-Jao-Soukharev's attack applies in the supersingular case [18, p. 18, Section 5].

Our contributions. We provide a new quantum attack on overstretched SIDH which uses a reduction of the underlying computational problem to an injective abelian hidden shift problem. This can be solved in quantum subexponential time and thus disproves the folkloric belief mentioned above.

The idea underlying our attack is to construct endomorphisms on the starting curve of the SIDH instance which act freely and transitively on a set containing the secret. Forcing these endomorphisms to be of a certain degree, we exploit the torsion point information supplied in SIDH to compute some information associated to the action of the endomorphisms on the secret without knowing

the latter. Solving a hidden shift problem on the function identifying the endomorphisms with the associated information then reveals the secret.

While this attack does not threaten SIDH with balanced parameter sets as originally proposed by Jao and De Feo [18] and used in SIKE [17], it shows that an attack using a hidden shift algorithm is possible despite the non-commutative nature of SIDH.

We describe our new attack as a special instance of a general framework. This allows us to unify other quantum attacks on isogeny-based schemes such as the one due to Childs, Jao and Soukharev [7] constructing isogenies between ordinary curves or a similar application of Kuperberg's hidden shift algorithm to CSIDH [5], which has recently been improved by Bonnetain-Schrottenloher [3] and by Peikert [25].

This framework might be of interest beyond isogeny-based cryptography. To define one of the key properties required, we introduce the notion of a *malleability oracle* for a function with respect to some group action. Under some additional assumptions, access to this oracle is sufficient to compute preimages of the function via solving an injective hidden shift problem.

Outline. In Section 2, we provide an overview of the notation used, we recall some mathematical background for isogeny-based cryptography and we review some quantum algorithms used in our attack. In Section 3, we present our general framework, namely sufficient conditions for computing preimages of one-way functions via reduction to a hidden shift problem and give a presentation of our new attack on overstretched SIDH in Section 4. In Section 5, we additionally instantiate our general framework with the attack of Childs, Jao and Soukharev as well as the application of quantum hidden shift algorithms to CSIDH. We conclude the paper in Section 6 with a discussion on potential improvements and future work.

2 Preliminaries

In this section, we introduce terminology and notation, and we recall relevant background on isogeny-based protocols and quantum algorithms.

2.1 Terminology

We call a function $\mu : \mathbb{N} \to \mathbb{R}$ *negligible* if for every positive integer c there exists an integer N_c such that $|\mu(x)| < \frac{1}{x^c}$ for every $x > N_c$. We call an algorithm *efficient* if the execution time is bounded by a polynomial in the security parameter of the underlying cryptographic scheme. Given any function, we take having *oracle access* to this function to mean that it is feasible to evaluate the function at any possible element in an efficient way. In particular, we assume that the oracle acts like a black box such that one query with an element from the domain outputs the corresponding value of the function.

Further, we call a function $f : \{0,1\}^* \to \{0,1\}^*$ *one-way*, if f can be computed by a polynomial time algorithm, but for all polynomial time randomized

algorithms F, all positive integers c and all sufficiently large $n = \text{length}(x)$, $\Pr[f(F(f(x))) = f(x)] < n^{-c}$, where the probability is taken over the choice of x from the discrete uniform distribution on $\{0,1\}^n$, and the randomness of F.

2.2 Mathematical background on isogenies

For more complete introductions to elliptic curves and to isogeny-based cryptography we refer to Silverman [31] and De Feo [10], respectively.

Let $\mathbb{F}_q$ be a finite field of characteristic p. In the following we assume $p \geq 3$ and therefore an elliptic curve E over $\mathbb{F}_q$ can be defined by its short Weierstrass form

$$E(\mathbb{F}_q) = \{(x,y) \in \mathbb{F}_q^2 \mid y^2 = x^3 + Ax + B\} \cup \{\mathcal{O}_E\}$$

where $A, B \in \mathbb{F}_q$ and $\mathcal{O}_E$ is the point $(X : Y : Z) = (0 : 1 : 0)$ on the associated projective curve $Y^2Z = X^3 + AXZ^2 + BZ^3$. The set of points on an elliptic curve is an abelian group under the "chord and tangent rule" with $\mathcal{O}_E$ being the identity element. The *j-invariant* of an elliptic curve is $j(E) = 1728\frac{4A^3}{4A^3+27B^2}$ and there is an isomorphism of curves $f : E_0 \to E_1$ if and only if $j(E_0) = j(E_1)$.

Given two elliptic curves E_0 and E_1 over a finite field $\mathbb{F}_q$, an *isogeny* is a non-constant rational map $\phi : E_0 \to E_1$ which is also a group homomorphism from $E_0(\overline{\mathbb{F}_q})$ to $E_1(\overline{\mathbb{F}_q})$, or equivalently, a rational map for which $\phi(\mathcal{O}_{E_0}) = \mathcal{O}_{E_1}$. Two curves are called *isogenous* if there exists an isogeny between them. The *degree* of an isogeny ϕ is its degree as a rational map. For separable isogenies, the degree is also equal to the number of elements in the kernel of ϕ. Note that we will always consider the separable case in the following.

Since an isogeny defines a group homomorphism $E_0 \to E_1$, its kernel is a subgroup of E_0. Conversely, any subgroup $S \subset E_0$ determines a (separable) isogeny $\phi : E_0 \to E_1$ with $\ker \phi = S$ and $E_1 = E_0/S$.

An *endomorphism* of an elliptic curve E defined over $\mathbb{F}_q$ is an isogeny defined over an extension of $\mathbb{F}_q$ mapping E onto itself. The set of endomorphisms of E together with the zero map forms a ring under pointwise addition and function composition. This ring is the *endomorphism ring* of E, denoted $\text{End}(E)$, and it is isomorphic either to an order in a quaternion algebra and E is called *supersingular*, or to an order in an imaginary quadratic field and E is referred to as an *ordinary* curve [31]. An isogeny between two curves having the same endomorphism ring is called a *horizontal* isogeny.

For any isogeny $\phi : E_0 \to E_1$, there exists another isogeny $\hat{\phi}$, called the *dual isogeny*, satisfying $\phi \circ \hat{\phi} = \hat{\phi} \circ \phi = [\deg(\phi)]$. Therefore, the property of being isogenous is an equivalence relation on the set of isomorphism classes of elliptic curves defined over $\mathbb{F}_q$.

2.3 Hard homogeneous spaces and CSIDH

Recall the notion of Couveignes' *hard homogeneous spaces* (HHS) [9], a finite commutative group action for which some operations are easy to compute and others are hard.

Instances of Couveignes' hard homogeneous spaces can be constructed using elliptic curve isogenies and have been the basis of one branch of isogeny-based cryptography which uses the group action we will describe in the following.

Denote the set of all isomorphism classes over $\overline{\mathbb{F}_q}$ of isogenous curves with n points and endomorphism ring $\mathcal{O}$ by $\mathrm{Ell}_{q,n}(\mathcal{O})$, and represent the isomorphism class of a curve E in $\mathrm{Ell}_{q,n}(\mathcal{O})$ by the j-invariant $j(E)$. Any horizontal isogeny $\varphi : E \to E_{\mathfrak{b}}$ between curves in $\mathrm{Ell}_{q,n}(\mathcal{O})$ is determined by E and $\ker \varphi$ up to isomorphism. This kernel corresponds to an ideal $[\mathfrak{b}]$ in $\mathcal{O}$. Since principal ideals in $\mathcal{O}$ correspond to isomorphisms, ideals that are equivalent in the ideal class group of $\mathcal{O}$, $\mathrm{Cl}(\mathcal{O})$, induce the same isogeny up to isomorphism. Hence, we have a well-defined group action

$$\begin{aligned} \cdot : \mathrm{Cl}(\mathcal{O}) \times \mathrm{Ell}_{q,n}(\mathcal{O}) &\to \mathrm{Ell}_{q,n}(\mathcal{O}), \\ ([\mathfrak{b}], j(E)) &\mapsto j(E_{\mathfrak{b}}), \end{aligned}$$

which is free and transitive ([33], Thm. 4.5, and erratum Thm. 4.5 of [30]).

Given two elliptic curves E_0, E_1 in $\mathrm{Ell}_{q,n}(\mathcal{O})$ up to isomorphism, it is in general assumed to be hard to find an isogeny $\varphi : E_0 \to E_1$.

A similar construction can be performed with endomorphism rings of supersingular curves. This occurrence of hard homogenous spaces is used for the *Commutative SIDH* (CSIDH) protocol [5] proposed for post-quantum non-interactive key exchange. Since the endomorphism rings of such curves are orders in a quaternion algebra, they are non-commutative and hence yield a group action with less desirable properties than in the construction for ordinary curves. Therefore, Castryck et al. suggest restricting the endomorphism ring to the subring of $\mathbb{F}_p$-rational endomorphisms which is an order in an imaginary quadratic field, and as such commutative. Again, the ideal class group of this order $\mathcal{O}$, $\mathrm{Cl}(\mathcal{O})$, acts on $\mathrm{Ell}_p(\mathcal{O})$, the set of all isomorphism classes of supersingular isogenous curves over $\mathbb{F}_p$ with $\mathbb{F}_p$-rational endomorphism ring (isomorphic to) $\mathcal{O}$.

Given that the set $\mathrm{Ell}_p(\mathcal{O})$ is non-empty, the group action is free and transitive (see [5], Thm. 7, summarizing results from [33], [30]), and can be used to perform a Diffie-Hellman-type key exchange. Note that this construction is strictly speaking not an instance of a HHS, as was pointed out by De Feo-Meyer [12].

There have been multiple proposals to attack concrete parameter suggestions for CSIDH with quantum algorithms. Peikert [25] uses Kuperberg's collimation sieve algorithm to solve the hidden shift instance with quantum accessible classical memory and subexponential quantum time, a strategy independently also explored by Bonnetain-Schrottenloher [3].

2.4 SIDH

We recall the *Supersingular Isogeny Diffie-Hellman* (SIDH) protocol which was introduced by Jao and De Feo in [18] and forms the basis of the *Supersingular Isogeny Key Encapsulation* (SIKE) mechanism [17] which has been submitted to NIST's post-quantum competition.

Fix some supersingular elliptic curve E_0 over a field $\mathbb{F}_{p^2}$, where p is a prime, and let N_1 and N_2 be two smooth integers coprime to p with $(N_1, N_2) = 1$. Further choose some points $P_A, Q_A, P_B, Q_B \in E_0$ such that P_A and Q_A generate the N_1-torsion of E_0, $E_0[N_1]$, and similarly, $\langle P_B, Q_B \rangle = E_0[N_2]$. Then the protocol is as follows:

1. Alice chooses a random cyclic subgroup of $E_0[N_1]$ generated by a point of the form $A = P_A + [x_A]Q_A$ and Bob chooses some random cyclic subgroup of $E_0[N_2]$ generated by $B = P_B + [x_B]Q_B$.
2. Alice then computes her secret isogeny $\varphi_A : E_0 \to E_0/\langle A \rangle$ and Bob computes his secret isogeny $\varphi_B : E_0 \to E_0/\langle B \rangle$.
3. Alice sends the curve $E_A := E_0/\langle A \rangle$ and the two points $\varphi_A(P_B), \varphi_A(Q_B)$ to Bob while Bob sends $\big(E_B := E_0/\langle B \rangle, \varphi_B(P_A), \varphi_B(Q_A)\big)$ to Alice.
4. Alice and Bob both compute the shared secret curve $E_{AB} := E_0/\langle A, B \rangle$ using the given torsion information. Alice obtains E_{AB} as $E_{AB} = E_B/\langle \varphi_B(A) \rangle = E_B/\langle \varphi_B(P_A) + [x_A]\varphi_B(Q_A) \rangle$, Bob proceeds analogously to compute the same curve.

For SIDH, one chooses the prime p of the form $p = N_1 N_2 f - 1$ with N_1 and N_2 being powers of 2 and 3, respectively. As the above protocol is vulnerable to adaptive attacks (see e.g., [14]), SIKE applies a variant of the Fujisaki-Okamoto transformation due to Hofheinz, Hövelmanns and Kiltz [16] to standard SIDH. To ensure that both Alice and Bob enjoy the same level of security, the recommended parameter sets for SIDH and SIKE suggest balanced parameters, i.e., $N_1 \approx N_2$.

The active attack on standard SIDH presented by Galbraith-Petit-Shani-Ti [14] utilizes the additional information on torsion points to recover a secret key through multiple executions of the protocol with malformed messages. Further, the given torsion point information is exploited in Petit's passive attack [26] on a non-standard variant of SIDH with unbalanced and comparatively large torsion parameters. The requirements on unbalancedness and size of parameters have recently been improved upon by Kutas et al. [22] who additionally show that, even with balanced parameters, there exist certain primes which facilitate an effective torsion point attack on SIDH.

For our quantum attack to work, we need to relax the balancedness condition of standard SIDH and require one of N_1 and N_2 to be larger than the other by a certain factor. In particular, we need $N_1 N_2 \gg p$ which prohibits choosing p as suggested by Jao-De Feo. We call this variant of SIDH *overstretched*. Note that this variant of SIDH is still polynomial time as long as N_1 and N_2 are powersmooth numbers, albeit much slower in practice than with the suggested parameters.

SIDH is believed to be immune to subexponential quantum attacks [1, 17, 18]. In particular, it has been claimed and been widely accepted that no reasonable variant of Childs et al.'s attack [7] exists for SIDH [18, p.18, Section 5]. Yet, we will show in this paper how to reduce SIDH with overstretched parameters to an abelian hidden shift problem.

2.5 Quantum algorithms for hidden shift problems

We first recall what is meant when two functions are said to be shifts of each other, or equivalently that these two functions hide a shift.

Definition 2.1. *Let $F_0, F_1 \colon G \to X$ be two functions defined on some group G, such that there exists some $s \in G$ satisfying $F_0(g) = F_1(g \cdot s)$ for all $g \in G$. The* hidden shift problem *is the problem of finding s given oracle access to the functions F_0 and F_1.*

We now proceed to give an overview of how this problem is solved in quantum subexponential time given that the functions satisfy some additional properties. To this end, we first describe a quantum property testing algorithm which determines whether two functions are shifts of one another with high probability, and then summarize the quantum algorithms solving the injective abelian hidden shift problem in the following.

Testing for the hidden shift property. Assume we are given two functions $F_0, F_1 : G \to X$ mapping a finite abelian group G (with exponent $k \in \mathbb{Z}^+$) to some finite set X. Before trying to immediately solve a hidden shift instance on F_0 and F_1, we want to decide whether these functions actually satisfy the hidden shift promise fully or at least for a large proportion of the elements in the domain. For a concise overview of the technique of property testing, we refer the reader to [24].

Here, we can use a special case of Friedl et al.'s testing algorithm [13] which utilizes quantum Fourier sampling and has a query complexity of $\mathcal{O}(k \log(|G|)/\delta)$, where $0 < \delta < 1$ parametrizes how rigorous the algorithm is in rejecting samples. Any input pair of hidden shift functions will be accepted by the tester with full certainty while a pair which disagrees with functions hiding a shift on at least $2|G|\delta$ values is rejected with some constant probability.

Finding the hidden shift. Multiple approaches utilizing quantum computation have been proposed to solve the hidden shift problem efficiently. Some of these works have considered different group structures as well as variations on the promise.

The first quantum subexponential algorithm is due to Kuperberg [20] and reduces the hidden shift problem to the hidden subgroup problem in the dihedral group $D_G \simeq \mathcal{C}_2 \ltimes G$, i.e., to finding a subgroup of D_G such that a function obtained from combining the input functions of the hidden shift problem is constant exactly on its cosets. It requires quantum subexponential time, namely $2^{\mathcal{O}(\sqrt{\log |G|})}$ quantum queries, for a finite abelian group G. A modification of this method proposed by Regev [28] reduces the memory required by Kuperberg's approach (from super-polynomial to polynomial) while keeping the running time quantum subexponential. Another, slightly faster algorithm, the collimation sieve, using polynomial quantum space was proposed later by Kuperberg [21]. In this variant, parameter tradeoffs between classical and quantum running time and quantumly accessible memory are possible.

These algorithms suggested for the hidden shift problem where the group G is abelian generally begin by producing some random quantum states, each with an associated classical "label", by evaluating the group action on a uniform superposition over the group G. For this generation of states, oracle access to the two functions F_0 and F_1 is needed. Then, the hidden shift s is extracted bitwise through performing measurements on specific quantum states (i.e., ones with desirable labels) which are generated from the random states via some sieve algorithm.

3 Malleability oracles and hidden shift attacks

In this section, we introduce the notion of a *malleability oracle* for a one-way function. Under some conditions, such an oracle allows the computation of preimages of given elements in quantum subexponential time by reduction to the hidden shift problem.

3.1 Malleability oracles

Recall the definition of a free and transitive group action.

Definition 3.1. *Let G be a group with neutral element e, and let I be a set. A* (left) group action *φ of G on I is a function $\varphi\colon G \times I \to I$, $(g, x) \mapsto \varphi(g, x)$, that satisfies $\varphi(e, x) = x$ for all x, and $\varphi(gh, x) = \varphi(g, \varphi(h, x))$ for all $x \in I$ and $g, h \in G$.*

The group action is called transitive *if and only if I is non-empty and for every pair of elements $x, y \in I$ there exists $g \in G$ such that $\varphi(g, x) = y$. The group action is called* free *if and only if $\varphi(g, x) = x$ implies $g = e$.*

From now on we will denote the action of a group element $g \in G$ on a set element $i \in I$ by $g \cdot i$.

Next, we define an oracle capturing the main premise required for our attack to compute preimages of one-way functions.

Definition 3.2. *Let $f : I \to O$ be an injective (one-way) function, let $\mathcal{A}_G$ be the action of a group G on I and let $g \cdot i$ denote the image of $g \in G$ acting on $i \in I$. A* malleability oracle *for G at $o := f(i)$ provides the value of $f(g \cdot i)$ for any input $g \in G$, i.e., the malleability oracle evaluates the map*

$$g \mapsto f(g \cdot i).$$

We call the function f malleable*, if a malleability oracle is available at every $o \in f(I)$.*

In Section 4 we show how a polynomial time malleability oracle can be constructed in the context of SIDH with overstretched parameters, and in Section 5 we see that in other contexts it may arise naturally.

3.2 Reduction to hidden shift problem

Given a malleability oracle at $o = f(i)$, computing a preimage of o reduces to a hidden shift problem.

Theorem 3.3. *Let $f : I \to O$ be an injective (one-way) function and let G be a group acting transitively on I. Given a malleability oracle for G at $o := f(i)$, a preimage of o can be computed by solving a hidden shift problem.*

Proof. Given $o \in f(I)$, our goal is to compute i such that $f(i) = o$. Let k be an arbitrary but fixed element in I and define

$$F_k \ : \ G \to O \ , \ \theta \mapsto f(\theta \cdot k).$$

Note that i is unique since f is an injective function, thus F_i is well-defined. Moreover, the malleability oracle allows us to evaluate the function F_i on any $\theta \in G$, since $F_i(\theta) = f(\theta \cdot i)$.

Fix some arbitrary $j \in I$. Since we know j, we can evaluate F_j on any group element θ by evaluating $f(\theta \cdot j)$ through simply computing the group action. Due to the transitivity of the group action of G, there exists $\sigma \in G$ such that $i = \sigma \cdot j$. Since for all $\theta \in G$

$$F_i(\theta) = f(\theta \cdot i) = f(\theta\sigma \cdot j) = F_j(\theta\sigma),$$

the functions F_j and F_i are shifts of each other.

Solving the hidden shift problem for F_i and F_j therefore allows us to recover σ, and thus to compute $i = \sigma \cdot j$. □

The following corollary will be used in our attack on overstretched SIDH.

Corollary 3.4. *Let $f : I \to O$ be an injective (one-way) function and let G be a finitely generated abelian group acting freely and transitively on I. Given a malleability oracle for G at $o := f(i)$, a preimage of o can be computed in quantum subexponential time.*

Proof. To obtain a hidden shift instance solvable by a subexponential quantum algorithm such as Kuperberg's, we only have to show that for every $k \in I$ the function $F_k(\theta) = f(\theta \cdot k)$ is injective. Then the claim follows from Theorem 3.3 and the discussion in Section 2.5.

Suppose that $F_k(g) = F_k(h)$ for some $g, h \in G$. This means $f(g \cdot k) = f(h \cdot k)$. Since f is injective and the group action is free, we get $g = h$. □

4 Attack on overstretched SIDH in quantum subexponential time

Despite the non-commutative nature of SIDH, we show in this section that one can find an abelian group action on its private key space. Moreover for sufficiently *overstretched* SIDH parameters, the torsion point information allows us to build

a malleability oracle under this group action. This gives rise to an attack using quantum subexponential hidden shift algorithms as proposed in our framework in Section 3.2.

This section is organized as follows: We first sketch our approach to exploit the torsion point information in Section 4.1. We then solve two technical issues in Sections 4.2 and 4.3. These issues require to tweak our general approach slightly, and we summarize the resulting algorithm in Section 4.4. Finally in Section 4.5, we present a hybrid approach to combine guessing part of the secret and computing the remaining part using our attack framework; this allows to extend the attack to further parameter sets.

Throughout this section, we use the following notation. Let $p \equiv 3 \pmod 4$ be prime, let E_0 be the supersingular elliptic curve with j-invariant 1728, given by the equation $y^2 = x^3 + x$, and let $\mathcal{O}_0 = \mathrm{End}(E_0)$ be its endomorphism ring. Note that $\mathcal{O}_0$ is well-known. More precisely, it is the $\mathbb{Z}$-module generated by $1, \iota, \frac{1+j}{2}$ and $\frac{\iota+j}{2}$, where ι denotes the non-trivial automorphism of E_0 mapping $(x, y) \mapsto (-x, iy)$ and j is the Frobenius endomorphism.

Remark 4.1. The attack we describe can be expanded to other curves with known endomorphism rings that are close to E_0, such as the curve used in the updated parameters of SIKE for the second round of NIST's post-quantum standardisation effort [1], by guessing the action on torsion points.

4.1 Overview of the attack

Let I be the set of cyclic subgroups of E_0 of order N_1, and let O be the set of j-invariants of all supersingular curves that are N_1-isogenous to E_0. Let f be the function sending any element of I to the j-invariant of the codomain of its corresponding isogeny, i.e.,

$$f : I \to O, \quad K \mapsto j(E_0/K). \tag{1}$$

The function f can be efficiently computed on any input using Vélu's formulae [32], provided N_1 is sufficiently smooth and that the N_1-torsion is defined over a sufficiently small extension field of $\mathbb{F}_p$. In SIDH, the latter is achieved by choosing $N_1 | p - 1$, but for sufficiently powersmooth N_1 this is true more generally.

On the other hand, inverting f amounts to finding an isogeny of degree N_1 from E_0 to a curve in a given isomorphism class, or equivalently to finding the subgroup of E_0 defining this isogeny. The conjectured hardness of this problem is at the heart of isogeny-based cryptography.

In the SIDH protocol, additional torsion point information is transmitted publicly as part of the exchange, and thus also given to potential attackers. For the security proof it is assumed that a variant of the following problem [18] is hard when $N_1 \approx N_2$.

Problem 4.2. *Let p be a large prime, let N_1 and N_2 be two large powersmooth coprime integers, and let $K \in I$ be a cyclic subgroup of order N_1 of E_0 chosen*

uniformly at random. Let $\varphi : E_0 \to E_0/K$. Given the supersingular curves E_0 and E_0/K together with the restriction of φ to $E_0[N_2]$, compute K.

Our attack exploits the information provided by the restriction of the secret isogeny to $E_0[N_2]$ to construct a malleability oracle for f at the (unknown) secret. This gives rise to an attack on overstretched SIDH following the framework of Section 3.

Let G be a subgroup of $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$. Then G induces a group action on I given by

$$\mathcal{A}_G : G \times I \to I \ , \ (\theta, K) \mapsto \theta(K).$$

Indeed, the degree of any non-trivial representative θ is coprime to N_1 and thus preserves the order of any generator of K.

By applying f, we identify a group action $\mathcal{A}_G$ of G on I with a map $G \times I \to O$. With a slight abuse of terminology, we will refer to the image under f of an orbit of G in I as an *orbit of G in O* in the following.

Note that the full group $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$ is not abelian. Our attack will require an abelian subgroup G acting on I such that G acts freely and transitively on the orbit of the secret key under this group action, as well as one element in this orbit. This leads to the following task.

Task 4.3. *Let $K \in I$ be any cyclic subgroup of E_0 of order N_1 chosen uniformly at random and let $\varphi : E_0 \to E_A := E_0/K$. Compute an element $L \in I$ and an abelian subgroup G of $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$, such that G acts freely and transitively on the orbit $G \cdot L$, f is injective on $G \cdot L$ and E_A is contained in $f(G \cdot L)$ in O.*

We solve this task in Section 4.2. More precisely, we partition I into three subsets restricted to which f is injective and give abelian groups that induce the required action on these subsets. One of these subsets of I always contains the secret K.

In order to apply our general framework from Section 3, it remains to construct a malleability oracle for f at $j(E_0/K)$ for any secret $K \in I$. To construct this oracle, we use both the torsion point information provided in the SIDH protocol and a solution to the following task.

Task 4.4. *Given an endomorphism $\theta \in G$ of degree coprime to N_1 and an integer N_2 coprime to N_1, compute an endomorphism θ' of degree N_2 such that θ and θ' induce the same action on the set I of cyclic subgroups of $E_0[N_1]$ of order N_1.*

An algorithm solving a small variation of this task when using sufficiently large N_2 and unbalanced N_1 and N_2 is presented in Section 4.3.

The following lemma results from the coprimality of $\deg(\theta)$ and N_1 when considering an SIDH key exchange instance with "secret" isogenies θ and φ as displayed in Figure 1.

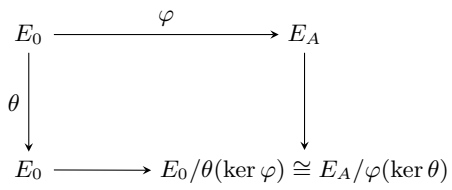

Fig. 1. SIDH key exchange instance with isogenies φ and the endomorphism θ.

Lemma 4.5. *Let $\varphi : E_0 \to E_A$ be an isogeny of degree N_1 and let $\theta \in End(E_0)$ be of degree coprime to N_1. Then $E_A/\varphi(\ker\theta)$ is isomorphic to $E_0/\theta(\ker\varphi)$.*

Let N_3 be the degree of θ. We cannot compute the curve $E_0/\theta(\ker\varphi)$ in general without the knowledge of the isogeny φ or its action on the N_3-torsion. However, we can compute the curve if we find an endomorphism θ' of degree N_3' such that θ and θ' have the same action on the N_1-torsion and $\varphi|_{E_0[N_3']}$ is known. This is the motivation behind Task 4.4 as we know the action of φ on the N_2-torsion in Problem 4.2. A solution to this task yields a malleability oracle for f with respect to the previously described group action of G on I in the SIDH setting.

This strategy to implement the malleability oracle is sketched in Algorithm 1, and Proposition 4.30 will prove its correctness.

Algorithm 1: Computation of $f(\theta(K))$, given $f(K)$ and $\theta \in G$

Input: Let $\varphi : E_0 \to E_A := E_0/K$ be an isogeny of degree N_1, let N_2 be coprime to N_1 and $G \subset (\mathcal{O}_0/N_1\mathcal{O}_0)^*$ one of the groups provided in Section 4.2. The input is E_0, $f(K) = j(E_A)$, $\varphi|_{E_0[N_2]}$ and $\theta \in G$.
Output: $f(\theta(K)) = j(E_0/\theta(K))$.
1 Compute endomorphism θ' of degree N_2 having the same action as θ on cyclic N_1-order subgroups of $E_0[N_1]$ using the lifting procedure of Section 4.3;
2 Using the knowledge of φ on $E_0[N_2]$, determine $\varphi(\ker\theta')$;
3 Compute $f(\theta(K)) = E_0/\theta(K) = E_A/\varphi(\ker\theta')$;
4 **return** $f(\theta(K)) = j(E_0/\theta(K))$

For parameters that allow us to construct a malleability oracle, we can then solve Problem 4.2 underlying SIDH-like protocols via a reduction to an injective abelian hidden shift problem using the framework introduced in Section 3.2.

Informal result 4.6. *Suppose the parameters allow the solution of Task 4.4 efficiently, then Problem 4.2 can be solved in quantum subexponential time.*

We use the remainder of this section to prove this result more formally under certain assumptions. To this end, we first give solutions to Task 4.3 and, for some parameters, to Task 4.4. Then we construct a malleability oracle using the torsion point information provided in SIDH and the solution for Task 4.4.

Apart from some technical details that we will address in the following, the informal result follows from Corollary 3.4. An overview of the attack is depicted in Algorithm 2.

Algorithm 2: Solving SIDH's underlying hardness assumption via an abelian hidden shift problem

Input: Let $\varphi : E_0 \to E_A := E_0/K$ be an isogeny of degree N_1 and P, Q a basis of $E_0[N_2]$. The input is E_0, E_A, $\varphi(P)$, $\varphi(Q)$.

Output: K defining φ.

1 Compute an abelian group $G \subset (\mathcal{O}_0/N_1\mathcal{O}_0)^*$ acting freely and transitively on the orbit $G(K)$ and one $J \in G(K) \subset I$;

2 Define $F_K : G \to O, g \mapsto f(g(K))$ and $F_J : G \to O, g \mapsto f(g(J))$;

3 Compute injective abelian hidden shift $\theta \in G$ of F_K and F_J, i.e., $\theta \in G$ such that $F_K(g) = F_J(\theta g)$ for all $g \in G$, with algorithm such as Kuperberg's. To this end, one evaluates F_K using Alg. 1 and F_J using the knowledge of J;

4 **return** $K := \theta(J)$

4.2 A free and transitive group action

Recall that E_0 is the supersingular curve with j-invariant 1728, given by the equation $y^2 = x^3 + x$. In this section we provide a solution to Task 4.3. We will do so by partitioning I into three orbits under the free and transitive action of abelian subgroups of $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$. Moreover, we show that restricting the function f to any of the three subsets yields an injective function. One of these three options will then always be a solution to Task 4.3.

For simplicity, we treat N_1 as a power of two in this section, but the results generalize to any power of a small prime. A generalization to powers of 3 is sketched in Appendix B.

Definition 4.7. *Let d be a positive integer. We say a supersingular elliptic curve E is* at distance *d from E_0 if there exists a separable isogeny ϕ with cyclic kernel of degree d from E_0 to E.*

Recall that f maps I to the set of j-invariants of curves at distance N_1 from E_0 due to the well-known correspondence between separable isogenies of degree N_1 and cyclic subgroups of $E_0[N_1]$ of order N_1. However, the correspondence is not necessarily one-to-one. In particular, if E_0 has a non-scalar endomorphism of degree N_1^2, then that endomorphism can be decomposed as $\hat{\tau_1} \circ \tau_2$ where τ_1 and τ_2 are non-isomorphic isogenies of degree N_1 from E_0 to the same curve E. For small enough N_1, the following lemma shows that two kernels correspond to the same curve if and only if they are linked by the automorphism $\iota : E_0 \to E_0, (x, y) \mapsto (-x, iy)$.

Lemma 4.8. *Suppose that $N_1^2 < \frac{p+1}{4}$. Then the only endomorphisms of degree N_1^2 of E_0 are $N_1 \cdot [1]$ and $N_1 \cdot \iota$, where $\iota : E_0 \to E_0, (x, y) \mapsto (-x, iy)$ is the non-trivial automorphism.*

Proof. Due to the condition that $N_1^2 < \frac{p+1}{4}$, an endomorphism θ of degree N_1^2 lies in $\mathbb{Z}[\iota]$. Let $\theta = a + b\iota$ for some $a, b \in \mathbb{Z}$. Then the degree of θ is $a^2 + b^2$. Now we have to prove that the only ways to decompose N_1^2 as a sum of two squares are trivial, i.e., $N_1^2 = N_1^2 + 0^2 = 0^2 + N_1^2$.

Let $N_1 = 2^k$, and we prove the statement by induction on k. For $k = 1$ the statement is trivial. Suppose that $k > 1$ and that $N_1^2 = a^2 + b^2$. Then a and b cannot both be odd as N_1^2 is divisible by four. If they are both even, then dividing by four yields a decomposition of $(N_1/2)^2 = (a/2)^2 + (b/2)^2$. By the induction hypothesis, this decomposition is trivial implying that N_1^2 can also only be decomposed in a trivial way. □

Corollary 4.9. *Suppose that $N_1^2 < \frac{p+1}{4}$. Let ϕ and ϕ' be two isogenies of degree N_1 from E_0 to a curve E. Then either* $\ker\phi = \ker\phi'$ *or* $\ker\phi = \iota(\ker\phi')$.

Proof. Consider the endomorphism $\tau = \hat{\phi}' \circ \phi$ of E_0. The degree of τ is N_1^2, so $\tau = N_1$ or $\tau = N_1 \cdot \iota$ by Lemma 4.8. In the former case, the isogenies ϕ and ϕ' are identical by the uniqueness of the dual. In the latter case, we have $\ker\phi = \iota(\ker\phi')$. □

Consider again the function $f : I \to O$ as defined in (1) sending a cyclic subgroup of $E_0[N_1]$ of order N_1 to the j-invariant of the corresponding elliptic curve, and note that it is not injective. However, an element in the image of f has precisely one preimage if the kernel of the corresponding isogeny is fixed by the automorphism ι.

To solve Task 4.3, we partition the set I of N_1-order subgroups of $E_0[N_1]$ in a way that f restricted to those subsets is injective. Furthermore, we define a free and transitive group action on each of these subsets.

Let P be a point such that $\{P, \iota(P)\}$ is a basis of $E[N_1]$, and define

$$S_P := \{E/\langle P + \alpha\iota(P)\rangle \text{ with } 2|\alpha\}. \tag{2}$$

We show that f restricted to S_P is injective.

Proposition 4.10. *Let $j(E_0) = 1728$, let $P \in E_0[N_1]$ be such that $\{P, \iota(P)\}$ is a basis of $E_0[N_1]$. Suppose that $N_1^2 < \frac{p+1}{4}$. Then S_P is a set of pairwise non-isomorphic curves.*

Proof. We apply Corollary 4.9. It is clear that $P + \alpha\iota(P)$ and $P + \alpha'\iota(P)$ are not scalar multiples of each other if $\alpha \neq \alpha'$ because $P, \iota(P)$ generate $E_0[N_1]$. It remains to show that for any even α, α', the points $P + \alpha\iota(P)$ and $-\alpha' P + \iota(P)$ are not scalar multiples of each other. Suppose that there exists an odd λ such that

$$P + \alpha\iota(P) = \lambda(-\alpha' P + \iota(P)).$$

Note that we can restrict to odd λs as the order of both points is N_1. Since $\{P, \iota(P)\}$ is a basis of the N_1-torsion, this implies that $1 \equiv -\lambda\alpha' \pmod{N_1}$. However, this cannot happen since α' is even. This contradiction concludes the proof. □

Our next goal is to define a free and transitive group action on S_P. The set S_P does not include all elliptic curves at distance N_1 from E_0, i.e., all curves in $f(I)$. Nonetheless, we first restrict ourselves to S_P and define the free and transitive group action for the remaining curves later when we address the cases where α is odd.

Every curve at distance N_1 from E_0 is of the form $E_0/\langle P + \alpha\iota(P)\rangle$ for some $\alpha \in \mathbb{Z}/N_1\mathbb{Z}$. This follows from the observation that the curves $E_0/\langle \beta_1 P + \beta_2 \iota(P)\rangle$ and $E_0/\langle -\beta_2 P + \beta_1 \iota(P)\rangle$ are isomorphic if their kernels are linked by ι.

Recall that E_0 is a curve with well-known endomorphism ring, and we are interested in the endomorphisms that are of degree coprime to N_1. While there are infinitely many such endomorphisms, we are only concerned with their action on $E_0[N_1]$, i.e., we are looking at the group $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$ which is finite and can be mapped injectively into $GL_2(\mathbb{Z}/N_1\mathbb{Z})$. Furthermore, we are only concerned with the action of the endomorphisms on I, i.e., on cyclic subgroups of $E_0[N_1]$ of order N_1, and we can therefore identify even more endomorphisms with each other by the following lemma.

Lemma 4.11. *Let $\theta = a + b\iota + cj + dk$ and $\theta' = a' + b'\iota + c'j + d'k$, where ι denotes the non-trivial automorphism of E_0, j the Frobenius endomorphism and $k := ij$, and let I be the set of cyclic subgroups of $E_0[N_1]$ of order N_1. Then $\theta(K) = \theta'(K)$ for every $K \in I$ if and only if there exists some $\lambda \in (\mathbb{Z}/N_1\mathbb{Z})^*$ such that*

$$(a, b, c, d) \equiv \lambda(a', b', c', d') \pmod{N_1}.$$

Proof. Considering the respective restrictions to $E_0[N_1]$, two endomorphisms are equal if they lie in the same class in $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$. Moreover, let θ_1, θ_2 be two endomorphisms such that $\theta_1 = [\lambda]\theta_2$ for some integer λ, and let P be an element of order N_1. Since scalar multiplication commutes with any endomorphism, it is easy to see that $\theta_1(P)$ and $\theta_2(P)$ generate the same subgroup in $E_0[N_1]$ if and only if λ is coprime to N_1. □

Now, we are ready to give a solution to Task 4.3 if $K \in I$ lies in S_P.

Proposition 4.12. *Let G be the group of equivalence classes of elements*

$$\{a + b\iota \mid a \text{ odd}, b \text{ even}\} \subset \mathbb{Z}[\iota] \subset End(E_0)$$

where we identify two elements if and only if they differ by multiplication by an odd scalar modulo N_1. Then G is an abelian group, and it acts freely and transitively on S_P.

Proof. It is easy to see that the endomorphisms in $\mathbb{Z}[\iota]$ of degree coprime to N_1 form an abelian subgroup of $\mathrm{End}(E_0)$. Using any basis for $E_0[N_1]$ of the form $\{P, \iota(P)\}$, we can write the elements of this subgroup as matrices of the form $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$, where $a^2 + b^2$ is coprime to N_1, a is odd and b is even. By identifying two endomorphisms $a_1 + b_1\iota$ and $a_2 + b_2\iota$ if there exists an integer λ coprime

to N_1 and an endomorphism δ such that $a_1 - \lambda a_2 + (b_1 - \lambda b_2) = N_1\delta$, which we can do by Lemma 4.11, we obtain G. As G is closed under multiplication and reduction modulo N_1, it is a subgroup of an abelian group and therefore abelian itself. Note that G contains all equivalence classes under Lemma 4.11 of endomorphisms of the form $a + b\iota$ for even b, independently of the chosen basis.

Recall that endomorphisms act on I, the set of cyclic order-N_1 subgroups of $E_0[N_1]$. To examine the orbit of an element in I, it is sufficient to look at the orbit of a generator of this cyclic group in I. We consider the orbit of P which has coordinates $(1, 0)$ with respect to our basis under the group action of G. The image of $(1, 0)$ under an element $\begin{pmatrix} 1 & b \\ -b & 1 \end{pmatrix}$ is $(1, -b)$. Inspecting the cyclic subgroups of E_0 these points generate, it becomes clear that the orbit of $G \cdot \langle P \rangle \subset I$ is in one-to-one correspondence with S_P. □

Dealing with an odd α. So far we have defined a free and transitive group action on S_P. However, when the secret kernel is generated by $P + \alpha\iota(P)$ with α odd, the corresponding curve is not contained in S_P. Now we handle the case where α is odd. One can show that the action of the previously defined group G on the set I has three orbits (see Appendix A for details). We have already seen that S_P is one orbit, but the odd α cases will split into two orbits. This means that the action of G cannot be free and transitive on these orbits, since the size of the orbits is smaller than the cardinality of the group. We avoid this issue by choosing a different (but similar) group of cardinality $N_1/4$ to act on the curves corresponding to odd α.

Let $Q = P + \iota(P)$. This first lemma shows when a linear combination of Q and $\iota(Q)$ is 0.

Lemma 4.13. *The linear combination $xQ + y\iota(Q) = 0$ holds if and only if $x \equiv y \pmod{N_1}$ and $x \equiv 0 \pmod{N_1/2}$.*

Proof. Observe that $xQ + y\iota(Q) = (x - y)P + (x + y)\iota(P)$ which equals zero exactly when $x \equiv y \pmod{N_1}$ and $x + y \equiv 0 \pmod{N_1}$, which is what we wanted to prove. □

Lemma 4.14. *Let $S_{P,1}$ and $S_{P,2}$ be defined in the following way:*

$$S_{P,1} := \left\{ E_0/\langle Q + \alpha\iota(Q)\rangle \mid \alpha \text{ even and } \alpha \in \left[0, \frac{N_1}{2} - 1\right] \right\},$$

$$S_{P,2} := \left\{ E_0/\langle Q + \alpha\iota(Q)\rangle \mid \alpha \text{ even and } \alpha \in \left[\frac{N_1}{2}, N_1 - 1\right] \right\}.$$

Then the curves in $S_{P,1}$ and $S_{P,2}$, respectively, are pairwise non-isomorphic.

Proof. Let α, α' be such that the corresponding curves are both in $S_{P,1}$, or $S_{P,2}$, respectively. First we show that $Q + \alpha\iota(Q)$ is an odd multiple of $Q + \alpha'\iota(Q)$ if and only if $\alpha \equiv \alpha' \pmod{N_1/2}$. Suppose there exists an odd λ such that

$$Q + \alpha\iota(Q) = \lambda(Q + \alpha'\iota(Q)).$$

Then Lemma 4.13 implies that $1-\lambda \equiv 0 \pmod{N_1/2}$ and $\alpha-\lambda\alpha' \equiv 0 \pmod{N_1/2}$ which implies $\alpha \equiv \alpha' \pmod{N_1/2}$. Now we have to show that $Q+\alpha\iota(Q)$ is never an odd multiple of $-\alpha Q + \iota(Q)$. Suppose there exists an odd λ such that

$$Q + \alpha\iota(Q) = \lambda(-\alpha' Q + \iota(Q)).$$

Lemma 4.13 implies that $1 + \alpha'\lambda \equiv \alpha - \lambda \equiv 0 \pmod{N_1/2}$. However, this leads to a contradiction, as $\alpha - \lambda \equiv 0 \pmod{N_1/2}$ implies λ is even while $1+\alpha'\lambda \equiv 0 \pmod{N_1/2}$ implies that λ must be odd. This shows that the curves in $S_{P,1}$ are pairwise non-isomorphic, and the same is true for curves in $S_{P,2}$. □

Our next goal is to give a free and transitive group action on $S_{P,1}$ and $S_{P,2}$. We start by defining the acting group.

Let us identify two endomorphisms $a + b\iota$ and $a' + b'\iota$ if there exists an odd $\lambda \in \mathbb{Z}/(N_1/2)\mathbb{Z}$ such that $a \equiv \lambda a' \pmod{N_1/2}$ and $b \equiv \lambda b' \pmod{N_1/2}$ and let us call the resulting group G_0. Let H be the subgroup of G_0 containing elements with even b.

Proposition 4.15. *H acts freely and transitively on $S_{P,1}$ and $S_{P,2}$.*

Proof. It is enough to show that H acts transitively on $S_{P,1}$ and $S_{P,2}$ as the cardinality of H, $S_{P,1}$ and $S_{P,2}$ is the same. We show that the orbit of $E/\langle Q\rangle$ contains every element in $S_{P,1}$. This follows from the fact that $(1+\alpha\iota)Q = Q+\alpha\iota(Q)$. Now we are left to show that H acts transitively on $S_{P,2}$. This follows from the fact that $(1+\alpha\iota)(Q + N_1\iota(Q)/2) = (1-\alpha N_1/2)Q + (\alpha + N_1/2)\iota(Q) = Q + (\alpha + N_1/2)\iota(Q)$ because α is even, so $(\alpha N_1/2)Q = 0$. □

What remains to be shown is that the curve $E/\langle P + \alpha\iota(P)\rangle$ with odd α is contained in either $S_{P,1}$ or $S_{P,2}$.

Proposition 4.16. *Let α be an odd integer. $E/\langle P + \alpha\iota(P)\rangle$ is contained in exactly one of $S_{P,1}$ or $S_{P,2}$.*

Proof. Observe that

$$P + \alpha\iota(P) = \frac{1+\alpha}{2}(P + \iota(P)) + \frac{\alpha-1}{2}(-P + \iota(P)) = \frac{1+\alpha}{2}Q + \frac{\alpha-1}{2}\iota(Q).$$

Note that the sum of $\frac{1+\alpha}{2}$ and $\frac{\alpha-1}{2}$ is odd which implies that one of the fractions is even while the other one is odd. If $\frac{\alpha-1}{2}$ is even, then it is clear that the curve is contained in either $S_{P,1}$ or $S_{P,2}$. Now we may assume that $\frac{1+\alpha}{2}$ is even. In this case, $E/\langle\frac{1+\alpha}{2}Q + \frac{\alpha-1}{2}\iota(Q)\rangle$ is isomorphic to $E/\langle\frac{1-\alpha}{2}Q + \frac{\alpha+1}{2}\iota(Q)\rangle$ (because their kernels are related by ι) and we have again shown that this curve is contained in either $S_{P,1}$ or $S_{P,2}$. □

In this subsection, we have partitioned I into three subsets corresponding to curves in S_P, $S_{P,1}$ and $S_{P,2}$, restricted to which f is injective. Moreover, for each of these sets we have given an abelian subgroup of $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$ that acts freely and transitively on it. Thus, we solve Task 4.3 as long as one determines which of the three sets contains E_A.

Remark 4.17. The reason for the even α case being simpler than the odd α case and using different groups is the following. Let G be the group defined in Proposition 4.12. Then one can show (for details see Appendix A) that the action of G has three orbits, one with $\frac{N_1}{2}$ and two with $\frac{N_1}{4}$ elements. The larger orbit corresponds to an even α and the two smaller ones correspond to odd αs. In order to have a free and transitive group action, the cardinality of the acting group has to be the same as the one of its orbit. This is why we need a smaller group acting on the orbits corresponding to odd αs. In particular, the action of G is no longer transitive on curves corresponding to odd αs as it has two orbits.

4.3 Lifting $\theta \in \mathbb{Z}[i]$ to an element of norm N_2 or eN_2

In the previous subsection, we described how to choose suitable abelian subgroups of $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$ in order to solve Task 4.3 after identifying whether $E_A = E_0/K$ is a curve in S_P, $S_{P,1}$ or $S_{P,2}$. Note that with the chosen groups, every acting group element can be trivially lifted to $\mathbb{Z}[i] = \mathbb{Q}[\iota] \cap \mathrm{End}(E_0)$. In this section we will describe how to lift these representatives to another endomorphism of E_0 with N_2-divisible degree which has the same action on I.

Let $q = 4$ denote the discriminant of $\mathbb{Z}[i]$. In this subsection, we solve the following task, which is a variant of Task 4.4, efficiently.

Task 4.18. *Let N_1, N_2 be integers such that $N_2 > |q|p^2N_1^4$ and let e denote the smallest positive quadratic non-residue modulo N_1. Given an endomorphism $\theta \in G$ of degree coprime to N_1 and an integer N_2 corpime to N_1, compute an endomorphism θ' of degree N_2 or eN_2 such that $\theta(K) = \theta'(K)$ for all $K \in I$.*

This is a relaxation of Task 4.4 in two ways. First, we require N_2 to be sufficiently large and unbalanced compared to N_1. Second, we allow θ' to be either of degree N_2 or eN_2 for some small integer e.

Note that e only depends on the fixed integer N_1 and not on the endomorphism θ. Assuming the conjecture due to Bach and Huelsbergen [2] we can bound e as follows.

Conjecture 4.19. ([2]) Let e denote the smallest integer that is a quadratic non-residue modulo an integer N_1. We have

$$e \leq \log(2)^{-1} \log N_1 \log\log N_1.$$

We now describe an algorithm to solve Task 4.18. By Lemma 4.11 it suffices to solve the following, which is similar to the problem solved at the core of the KLPT algorithm [19].

Task 4.20. *Given $\theta = a_0 + b_0 i + (c_0 + d_0 i)j$, find $\theta' = a' + b'i + (c' + d'i)j$ of degree eN_2 with coefficients $(a', b', c', d') \equiv \lambda(a_0, b_0, c_0, d_0) \pmod{N_1}$ for some scalar $\lambda \in (\mathbb{Z}/N_1\mathbb{Z})^*$.*

Let $\theta' = \lambda a_0 + N_1 a_1 + i(\lambda b_0 + N_1 b_1) + (\lambda c_0 + N_1 c_1 + i(\lambda d_0 + N_1 d_1))j$. Then its norm equals

$$\mathrm{Norm}(\theta') = h(\lambda a_0 + N_1 a_1, \lambda b_0 + N_1 b_1) + ph(\lambda c_0 + N_1 c_1, \lambda d_0 + N_1 d_1), \quad (3)$$

where $h(x, y) = \mathrm{Norm}(x + yi)$ is a principal quadratic form of discriminant q. Since $\theta \in \mathbb{Z}[i]$ implies $c_0 = d_0 = 0$, Equation (3) simplifies to

$$\mathrm{Norm}(\theta') = h(\lambda a_0 + N_1 a_1, \lambda b_0 + N_1 b_1) + pN_1^2 h(c_1, d_1). \quad (4)$$

Let e be 1 if $N_2/h(a_0, b_0)$ is a quadratic residue modulo N_1, and the smallest positive non-quadratic residue modulo N_1 otherwise. Compute θ' such that $\mathrm{Norm}(\theta') = eN_2$. Considering Equation (4) modulo N_1, we obtain

$$eN_2 \equiv \lambda^2 h(a_0, b_0) \pmod{N_1}. \quad (5)$$

Since both eN_2 and $h(a_0, b_0) = \deg(\theta)$ are coprime to N_1, the choice of e implies that there exists a solution for λ. Compute any such solution, and lift it to the integers in $[1, N_1 - 1]$ in a natural way. Note that this is without loss of generality as any other lift of λ corresponds to a change in a_1, b_1 instead.

Then consider the equation modulo N_1^2. This gives an affine relation between a_1 and b_1 modulo N_1, i.e.,

$$2\lambda(a_0 a_1 + b_0 b_1) \equiv \frac{\mathrm{Norm}(\theta') - \lambda^2 h(a_0, b_0)}{N_1} \pmod{N_1}.$$

Take the affine relation between a_1 and b_1 modulo N_1, say $e_b b_1 = e_a a_1 + e_c + mN_1$ for some fixed integers e_a, e_b, e_c and a variable integer m. Assume $e_b \not\equiv 0 \pmod{p}$ as lifting would be trivial otherwise, and substitute b_1 in Equation (4) modulo the prime p, i.e.,

$$eN_2 \equiv h(\lambda a_0 + N_1 a_1 \ , \ \lambda b_0 + N_1 e_b^{-1}(e_a a_1 + e_c + mN_1)) \pmod{p}.$$

Note that fixing any value for m leaves a quadratic equation in a_1 modulo p. Fix $m = 0$ and complete the square in the equation to solve it, if there exists a solution. Otherwise, increase m by one and repeat. Heuristically, one expects this degree-2 polynomial modulo p to be split with probability $1/2$ and hence we expect to iterate twice before finding a solution.

Once a solution for a_1 is obtained modulo p, lift it to the integers. One is left with the problem of representing an integer as the norm of an element in $\mathbb{Z}[i]$, i.e., finding c_1 and d_1 such that

$$h(c_1, d_1) = r := \frac{\mathrm{Norm}(\theta') - h(\lambda a_0 + N_1 a_1, \lambda b_0 + N_1 b_1)}{pN_1^2}$$

if they exist. Clearly, r can only be a norm if it is positive. This happens when the parameters N_1 and N_2 are overstretched, and more precisely if $\mathrm{Norm}(\theta') > |q| p^2 N_1^4$, where q denotes the discriminant of h.

If the prime decomposition of r is known, Cornacchia's algorithm [8] can efficiently answer the question whether r can be decomposed that way and compute a solution if one exists. Assuming the values of $h(\lambda a_0 + N_1 a_1, \lambda b_0 + N_1 b_1)$ behave as random values around $|q|p^2 N_1^4$, one expects to choose $\log(|q|p^2 N_1^4)$ different values for m with a solution to the quadratic equation modulo p before finding a solution with Cornacchia's algorithm.

Remark 4.21. Cornacchia's algorithm requires the factorization of r. This could be done in subexponential time on a classical computer or in quantum polynomial time. To avoid such computations, we apply Cornacchia's algorithm only when r is a prime and keep sampling further values for m otherwise.

Since we do not apply Cornacchia's algorithm until r is prime, we expect to sample roughly $\log(|q|p^2 N_1^4)$ values for m until r is prime.

It is easy to see that a solution for (a_1, b_1, c_1, d_1) as computed with the routine described above satisfies Equation (4). The full algorithm is summarized in Algorithm 3.

Since Algorithm 3 provides θ' of norm N_2 or eN_2, where e denotes the smallest quadratic non-residue modulo N_1, Conjecture 4.19 facilitates the derivation of the following lemma.

Lemma 4.22. *Assuming Conjecture 4.19, a solution of Algorithm 3 to Task 4.18 is of norm at most eN_2, where $e \leq \log^{-1}(2) \log N_1 \log\log N_1$.*

An examination of algorithm 3 shows that it aborts after a fixed number of trials for m. Recalling the discussion at the beginning of this section, we can then state the following.

Lemma 4.23. *Algorithm 3 always terminates and is correct if it returns a solution.*

We conclude this section by investigating the heuristic probability of the lifting algorithm being successful despite aborting, as well as its complexity.

The success probability is based on the following heuristic assumptions:

1. The discriminant of $h(\lambda a_0 + N_1 a_1\ ,\ \lambda b_0 + N_1 b_1)$ in Line 10 of Algorithm 3 is uniformly distributed modulo p.
2. r in Line 13 of Algorithm 3 behaves like a random value around $|q|p^2 N_1^4$.

Lemma 4.24. *Let $\varepsilon > 0$ and let $B := \log(\varepsilon) \log(|q|p^2 N_1^4) / \log(1 - \log^{-1}(|q|p^2 N_1^4))$ be the limit for the number of values of m over which we iterate in Algorithm 3 (Step 9). Under the heuristic assumptions mentioned in the preceding paragraph, the algorithm returns a lift with probability $1 - \varepsilon$ and an error $\perp$ otherwise.*

Proof. Based on the heuristic that the discriminant of $h(\lambda a_0 + N_1 a_1\ ,\ \lambda b_0 + N_1 b_1)$ in Step 10 of Algorithm 3 is uniformly distributed modulo p, we expect to find a solution for $a_1 \pmod p$ for half of the chosen m. Moreover, if r (Line 13, Algorithm 3) behaves like a random value around $|q|p^2 N_1^4$, we expect it to be prime with probability roughly $1/\log(|q|p^2 N_1^4)$ and Cornacchia's algorithm to

Algorithm 3: Lift element from $\mathbb{Z}[i]$ to quaternion of norm N_2 or eN_2

Input: $\theta = a_0 + b_0 i \in \mathrm{End}(E_0)$, $q := \mathrm{Disc}(\mathbb{Z}[i])$ and parameters p, ε, N_1, $N_2 > p^2 N_1^4$

Output: $\theta' = \lambda a_0 + N_1 a_1 + (\lambda b_0 + N_1 b_1)i + N_1 c_1 j + N_1 d_1 k$ and $\mathrm{Norm}(\theta') = N_2$ or eN_2 with probability $1 - \varepsilon$ and $\perp$ otherwise

Let $h(x, y) := \mathrm{Norm}(x + yi)$;

if λ *in* $N_2 = \lambda^2 h(a_0, b_0) \pmod{N_1}$ *has solution for* λ **then**

  Compute λ;

else

  $e \leftarrow$ smallest quadratic non-residue $\pmod{N_1}$;

  Compute λ in $eN_2 = \lambda^2 h(a_0, b_0) \pmod{N_1}$;

Compute linear relation between a_1 and b_1 mod N_1, say $e_b b_1 = e_a a_1 + e_c \pmod{N_1}$ for some integers e_a, e_b, e_c, using

$$2\lambda(a_0 a_1 + b_0 b_1) = \frac{eN_2 - h(\lambda a_0, \lambda b_0)}{N_1} \pmod{N_1};$$

$B \leftarrow 2\log(\varepsilon)\log(|q|p^2 N_1^4)/\log(1 - \log^{-1}(|q|p^2 N_1^4))$;

for $m = 0, 1, \ldots, B$ **do**

  Substitute b_1 using expression $e_b b_1 = e_a a_1 + e_c + mN_1$ in

$$eN_2 = h(\ \lambda a_0 + N_1 a_1\ ,\ \lambda b_0 + N_1 b_1\ ) \pmod{p};$$

  if *solution for* $a_1 \pmod{p}$ *exists* **then**

    Compute a_1 and b_1 modulo p and lift them to integers in $[-p/2, p/2]$;

    $r \leftarrow \frac{eN_2 - h(\lambda a_0 + N_1 a_1, \lambda b_0 + N_1 b_1)}{pN_1^2}$;

    if r *is prime* **then**

      Use Cornacchia's algorithm to find solutions for c_1, d_1 in $h(c_1, d_1) = r$ or determine that no solution exists;

    if *solution is found* **then**

      return $\theta' = \lambda a_0 + N_1 a_1 + (\lambda b_0 + N_1 b_1)i + N_1 c_1 j + N_1 d_1 k$;

return $\perp$

provide a solution with probability roughly $1/(\log(|q|p^2 N_1^4))$ due to Landau [23] and Ramanujan [27]. Iterating over B values of m, we therefore expect our algorithm to return $\perp$ with probability

$$\Big(1 - \frac{1}{\log(|q|p^2 N_1^4)}\Big)^{B/2(\log(|q|p^2 N_1^4)}.$$

In particular, iterating over $B \geq 2\log(\varepsilon)\log(|q|p^2 N_1^4)/\log(1 - \log^{-1}(|q|p^2 N_1^4))$ as in Algorithm 3, we fail to find a solution with probability less than ε heuristically. □

Lemma 4.25. *Algorithm 3 runs in time polynomial in* $\log p$, $\log N_1$ *and* $\log^{-1}(\varepsilon)$ *for every* $\varepsilon > 0$.

Proof. For any $\varepsilon > 0$, the worst-case runtime of the algorithm stems from the iteration over up to $2\log(\varepsilon)\log(|q|p^2N_1^4)/\log(1-\log^{-1}(p^2N_1^4))$ values of m. In each iteration one needs to solve at most one quadratic equation modulo p, and apply Cornacchia's algorithm to a prime of size polynomial in p and N_1. □

We have implemented the lifting algorithm in MAGMA and the experiments are consistent with the theoretical results. The main drawback of our lifting algorithm is the necessity of having the unbalancedness $N_2 > p^2N_1^4$. While the algorithm presented in this section might be improved, the following remark discusses why we can a priori not expect to find a lifting algorithm for balanced parameters.

Remark 4.26. The bound of $N_2 > p^2N_1^4$ is partly caused by inefficiencies in the lifting algorithm. This raises the question of what is the best bound that could be achieved with unbounded computational power in the lifting algorithm.

Since we cannot expect a non-homogeneous quadratic equation in two variables to have a solution, we cannot expect to find a lift of norm eN_2 in $\mathbb{Z}[i]$. Therefore, pN_1^2 is a lower bound for the degree of the lifted endomorphisms. However, using a heuristic argument and the pigeonhole principle we see that pN_1^3 could be feasible. More precisely, after fixing λ it remains to find $a_1 < A$ and $b_1 < B$ for two bounds $A, B \in \mathbb{Z}$, such that

$$\frac{h(\lambda a_0 + N_1a_1, \lambda b_0 + N_1b_1) - eN_2}{N_1} \equiv 0 \pmod{pN_1}.$$

Hence, if $A \approx B \approx \sqrt{pN_1}$ and we rely on the heuristic that different a_1, b_1 will lead to distinct values in the above equation modulo pN_1, we expect the existence of a solution by the pigeonhole principle. By the definition of h, we have $h(\lambda a_0 + N_1a_1, \lambda b_0 + N_1b_1) \approx \max\{(AN_1)^2, (BN_1)^2\}$. Therefore, a solution with $A \approx B \approx \sqrt{pN_1}$ gives roughly

$$h(\lambda a_0 + N_1a_1, \lambda b_0 + N_1b_1) \approx eN_2 \approx pN_1^3.$$

We conclude this section with some final remarks on the lifting algorithm.

Remark 4.27. If N_2 is even larger than $p^2N_1^4$ and a factor of e divides N_2, our algorithm can start by setting $\mathrm{Norm}(\theta')$ to be a divisor of N_2, reducing the degree of the lifted endomorphism.

Remark 4.28. The lifting algorithm requires only constant memory.

Remark 4.29. In Algorithm 2, which solves the problem underlying overstretched SIDH, the lifting algorithm is used for every element of the acting group G. In our case this group is of size N_1. Since the lifting fails with probability ε in every single run and the functions in Algorithm 2 are only exact shifts of each other

when it does not fail a single time, we need to choose ε sufficiently small. Assuming independence between the different executions of the lifting algorithm, we expect to find two functions satisfying the promise of a hidden shift with probability $(1-\varepsilon)^{N_1} \approx 1 - \varepsilon N_1$ by first order Taylor approximation. Thus, choosing $\varepsilon < \frac{1}{2N_1}$ we would expect our lifting to work with probability roughly $\frac{1}{2}$ on all endomorphisms of G. Note that the lifting remains polynomial in $\log N_1$ and $\log p$ for such an ε. Clearly, one could choose ε even smaller to achieve an even larger heuristic success probability of the algorithm on all elements of G.

4.4 Algorithm summary

We start the summary of our attack by proving that a solution to Task 4.4 allows us to construct a malleability oracle for f.

Proposition 4.30. *Let $f_{|I'} : I' \to O$ be the function defined in (1) restricted to a domain I' so it is injective, let G be an abelian subgroup of $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$ acting freely and transitively on I' and let $\varphi : E_0 \to E_0/K$, where $K \in I'$ is chosen uniformly at random and unknown. Suppose the public parameters allow the solution of Task 4.4 for endomorphisms in G efficiently. Given $\varphi|_{E_0[N_2]}$, we have a malleability oracle for G at $f_{|I'}(K)$.*

Proof. We need to show that there exists an efficient algorithm that, on input $f(K) = f_{|I'}(K) = j(E_0/K) =: j(E_A)$ and $\theta \in G$, computes $f(\theta(K))$. Let φ be the isogeny corresponding to the cyclic subgroup $K \subset E_0$ of order N_1.

The endomorphism θ has degree coprime to N_1 and using the efficient solution to Task 4.4, we can compute some θ' of degree N_2 such that it has the same action on the N_1-torsion as θ. Therefore, $f(\theta(K)) = E_0/\theta(K) = E_0/\theta'(K)$ up to isomorphism. By Lemma 4.5, this equals $E_A/\varphi(\ker\theta')$. Since $\ker\theta'$ lies in $E_0[N_2]$, we can compute its image under φ and therefore we can calculate $f(\theta(K)) = E_A/\varphi(\ker\theta')$ efficiently. □

Proposition 4.30 calls for solutions to the Tasks 4.3 and 4.4. In Sections 4.2 and 4.3 we presented solutions to slight *variants* of these tasks. We use the remainder of this section to summarize the impact of these variations on the success of our approach.

Clearly, restricting the function $f : I \to O$ to a subset I' such that $f_{|I'}$ is injective and that it contains the secret kernel one aspires to recover requires information on the secret we do not posses. However, we showed in Section 4.2 that one can find three subsets of I such that their union contains all of I and f restricted to any of the subsets is injective, i.e., one of the three will yield the desired result. These three subsets corresponded to the sets of curves S_P, $S_{P,1}$ and $S_{P,2}$ at distance N_1 from E_0. Moreover, we provided abelian subgroups of $\mathbb{Q}[i] \cap (\mathcal{O}_0/N_1\mathcal{O}_0)^*$ acting freely and transitively on their respective subset of I.

Then, we gave an algorithm to solve Task 4.18, a variant of Task 4.4 when N_1 and N_2 are sufficiently unbalanced, lifting endomorphisms contained in the abelian subgroups of $(\mathcal{O}_0/N_1\mathcal{O}_0)^*$ to one with the same action on I of degree N_2

or eN_2. Here, e is a small integer bounded by Conjecture 4.19. As a consequence, to use the torsion point information of $E_0[eN_2]$ under the secret isogeny given the image of $E_0[N_2]$, we need to guess the action on $E_0[e]$ which takes $O(e^3)$ trials.

For each combination of guesses of $E_0[eN_2]$ and whether the secret is in the subset of I corresponding to $S_P, S_{P,1}, S_{P,2}$, we can test whether the hidden shift property for the corresponding functions F_K and F_J as defined in Algorithm 2 is satisfied. This can be done with the algorithm due to Friedl et al. [13] mentioned in Section 2.5. Once the premise is satisfied, we recover the solution to an abelian hidden shift problem using a subexponential abelian hidden shift algorithm such as Kuperberg's [21] and thus we recover the secret isogeny as described in Section 4. Therefore, we can summarize our result as follows.

Theorem 4.31. *Let $N_2 > |q|p^2N_1^4$. Then the SIDH problem can be solved in quantum subexponential time via a reduction to the injective abelian hidden shift problem.*

During this section, we have made some restrictions to simplify the presentation of our attack. We described the attack with the starting curve E_0 being a supersingular curve with j-invariant 1728. However, the attack also applies to other curves with known endomorphism rings that are close to E_0. In Section 4.2, we described the required group action on I under the assumption that N_1 is a power of 2, which can be generalized to powers of small primes. A sketch for powers of 3 can be found in Appendix B. Finally, we assumed that $N_1^2 < \frac{p+1}{4}$ in Lemma 4.8. However, to run our attack we can slightly ease this restriction. Namely, if $N_1^2 > \frac{p+1}{4}$, then we choose a divisor N_1' of N_1 such that $N_1'^2 < \frac{p+1}{4}$ and run the attack with N_1' instead. This will reveal the N_1' part of the isogeny and then we can guess the remaining part. For sufficiently small $\frac{N_1}{N_1'}$, this is only a minor inefficiency.

4.5 Hybrid attacks on overstretched SIDH

In this section, we examine to what extent partial knowledge of the secret, i.e., knowledge of the most or least significant bits, renders the attack more efficient. Moreover, we describe how the attack can be adapted to some further parameters that are not quite sufficiently unbalanced.

We start with the case where the most significant bits of the secret are leaked. These bits correspond to the last steps of the secret isogeny in the isogeny graph. Assume N_1 is a power of a prime ℓ. If the most significant k digits of the secret with respect to their representation in base ℓ are leaked or guessed correctly, the remaining isogeny we need to recover is of degree N_1/ℓ^k and we can run our attack as soon as N_1/ℓ^k fulfills the unbalancedness criterion $N_2 > p^2(N_1/\ell^k)^4$.

The case where the least significant digits are known or guessed requires a little more work. For simplicity of our exposition we assume again that N_1 is a power of 2 as in Section 4.2, but the results generalize to powers of small primes.

Lemma 4.32. *Let G be the group of Proposition 4.12, and let $G' \subset G$ be the subset of the form $\{a + b\iota \mid a \text{ odd}, b \text{ divisible by } 2^k\}$ where we identify two endomorphisms with each other if they differ by multiplication by an odd scalar modulo N_1. Then G' is an abelian subgroup of G.*

Proof. Since G is abelian, it suffices to show that G' is a subgroup. Consider $(a+b\iota)(a'+b'\iota) = (aa' - bb') + (ab' + a'b)\iota$. It is easy to see that $aa' - bb'$ is odd and $ab' + a'b$ is divisible by 2^k if $a + b\iota$ and $a' + b'\iota$ are in G'. □

Assume the least significant k bits of the secret, or equivalently the first k steps of the secret isogeny, are known. Kernels of isogenies of degree $N_1 > 2^k$ that share the same first k steps lie in the same 2^k-torsion subgroup and are therefore congruent modulo 2^k.

Let $S_P, S_{P,1}, S_{P,2}$ be the three sets introduced in Section 4.2.

Proposition 4.33. *Let S' be any of the subsets of S_P, as defined in (2), containing curves associated to isogeny-kernels with α's congruent modulo 2^k. The group G' of Lemma 4.32 acts freely and transitively on any S'.*

Proof. First, we need to show that $G' \times S' \to S'$ is well-defined. Let $(1 + b\iota)$ be a representative of some element in G' and let $P + k\iota(P)$, for some $k \in \mathbb{Z}$, be the kernel of an isogeny leading to a curve in S'. We have

$$(1 + b\iota)(P + k\iota(P)) = P + k\iota(P) + b(\iota(P) - kP) \equiv P + k\iota(P) \pmod{b}$$

and as b is divisible by 2^k, $P + k\iota(P) \in S'_i$ implies $(1 + b\iota)(P + k\iota(P)) \in S'$. That the action is free and transitive follows from Proposition 4.12 and a counting argument as $|G|/|G'| = 2^{k-1} = |S_P|/|S'|$. □

Similarly, we can take subsets of $S_{P,1}$ and $S_{P,2}$ and restrict the group acting on these sets of curves.

This gives rise to an attack strategy when N_2 is not large enough. Guessing k bits of the secret before applying the attack on the remaining part allows an attacker to reduce the requirements on the parameters to $N_2 > p^2(N_1/2^k)^4$. This is the same as when guessing the last bits of the secret.

Given such a partial isogeny, one computes the correct set S' from the kernel of the known part of the isogeny. Moreover, one needs to compute the lift of elements of G' to endomorphisms of norm N_2 or eN_2. Computing the action of G' on the set S' allows to test for the hidden shift property. Once it is satisfied, the secret can be recovered by solving an injective abelian hidden shift problem. Otherwise, one can make another guess on the k bits of the secret.

Apart from reducing the requirements on the unbalancedness, guessing part of the isogeny reduces the number of elements one needs to lift and the size of the hidden shift instance.

5 Childs-Jao-Soukharev's attack on HHS

We begin by providing more detail on how the algorithm proposed by Childs, Jao and Soukharev [7] succeeds to construct an isogeny between two given ordinary elliptic curves in quantum subexponential time. The provided strategy can further be applied to attack CSIDH [5].

Recall the free and transitive group action from Section 2.3 of the class group on the set of isogenous ordinary curves with the same endomorphism ring. The hard problem is to find an isogeny between two isogenous ordinary elliptic curves with the same endomorphism ring, i.e. reversing this group action. Childs-Jao-Soukharev provide an algorithm that constructs such an isogeny in quantum subexponential time [7] using a reduction to the hidden shift problem.

We summarize the core idea as another instance of our framework using malleability oracles. Let $I := \mathrm{Cl}(\mathcal{O})$ and $O := \mathrm{Ell}_{q,n}(\mathcal{O})$. We can look at the group action defined above as a one-way function

$$f : I \to O \ , \ [x] \mapsto [x] \cdot j(E_0).$$

Note the class group $\mathrm{Cl}(\mathcal{O})$ acts on itself and therefore f has a malleability oracle with respect to the class group readily available everywhere on the image, i.e., f is malleable with respect to this group action.

Finding an isogeny φ is now equivalent to finding the ideal $[\mathfrak{b}]$ in $\mathcal{O}$ corresponding to the kernel of φ, i.e., we would like to compute the preimage of f at $j(E_1) = [\mathfrak{b}] \cdot j(E_0)$.

Childs-Jao-Soukharev observed that the functions $F_0, F_1 : \mathrm{Cl}(\mathcal{O}) \to \mathrm{Ell}_{q,n}(\mathcal{O})$ defined by $F_i([x]) := [x] \cdot j(E_i)$ for $i = 0, 1$, i.e., $F_0([x]) = f([x])$ and $F_1([x]) = f([x][\mathfrak{b}])$, are shifts of each other. Moreover, they are injective functions since the action of the class group on $\mathrm{Ell}_{q,n}(\mathcal{O})$ is free and transitive. The injective abelian hidden shift problem can be solved in quantum subexponential time, which allows one to recover $[\mathfrak{b}]$ and therefore an isogeny $\varphi : E_0 \to E_1$.

Analogously to the case for ordinary curves, the group action in CSIDH utilizing supersingular curves can be attacked this way. Recall that CSIDH uses the $\mathbb{F}_p$-rational endomorphism ring of the fixed starting curve E_0, $\mathcal{O}$. In the Diffie-Hellman-type key exchange, recovering a party's secret key constitutes of computing their secret ideal class $[\mathfrak{b}] \in \mathrm{Cl}(\mathcal{O})$ which satisfies $[\mathfrak{b}] \cdot E_0 = E_B$ for the party's public curve E_B. Through defining functions $F_0, F_1 : \mathrm{Cl}(\mathcal{O}) \to \mathrm{Ell}_p(\mathcal{O})$ by $F_0([x]) = [x] \cdot E_0$ and $F_1([x]) = [x] \cdot E_B$, it is again possible to reduce finding Bob's secret key $[\mathfrak{b}]$ to an instance of the injective hidden shift problem: We have $F_1([x]) = F_0([x] \cdot [\mathfrak{b}])$ for any ideal class $[x] \in \mathrm{Cl}(\mathcal{O})$, where the functions are both injective due to the group action being free and transitive.

6 Conclusion and further work

In this paper, we constructed an abelian group action on the key space of the inherently non-commutative SIDH. Having this group action in place allows us

to construct a malleability oracle using the torsion point information provided in SIDH when overstretched and sufficiently unbalanced parameters are being used. This contradicts the widespread folklore that no such group action exists in the branch of isogeny-based cryptography where one considers the full isogeny graph of a supersingular elliptic curves. We embedded our attack in a more general framework that also captures other quantum attacks on schemes in isogeny-based cryptography.

The attack does *not* apply to balanced parameters as specified in the original SIDH proposal [18] or the NIST post-quantum candidate SIKE [17]. The unbalancedness condition between N_1 and N_2 is stronger than required by the attack from [22]. However, the obstruction to attack SIDH with balanced parameters in our case is not directly related to the hindrances in other attacks on unbalanced SIDH exploiting torsion point information [4, 22, 26] but to limitations of the KLPT algorithm [19] and the ones described in Remark 4.26 instead. Improvements to the lifting subroutine included in the KLPT algorithm would not only partially decrease the required unbalancedness of SIDH parameters in this work, but also improve various isogeny-based schemes such as Galbraith-Petit-Silva's signatures [15] and SQISign [11].

Further improvements to decrease the unbalancedness required by our lifting algorithm are possible by lifting elements of the ring $j\mathbb{Z}[i]$ instead of $\mathbb{Z}[i]$. This was done in the original KLPT algorithm [19]. Combining this approach with the improvements presented by Petit-Smith at Mathcrypt 2018 lowers the required unbalancedness to $N_2 > pN_1^3$ which is reaching the heuristic bound of this approach outlined in Remark 4.26. The idea is to lift the endomorphism $j\theta$ instead of θ. Recovering the isogeny between the curves E_0 and $E'_A = E_0/j(A)$ instead of $E_A := E_0/A$ and applying the Frobenius endomorphism afterwards, gives the required isogeny from E_0 to E_A. The details of this approach will be added to the full version of this paper.

Future work will moreover extend the given quantum algorithm to more general group actions of quadratic orders that embed optimally into the (known) endomorphism ring of the starting curve. Hereby, the starting curve does not necessarily need to be of j-invariant 1728.

To improve the framework further and to give conditions on the malleability oracle that have to be fulfilled in order to invert one-way functions in quantum polynomial time, as well as providing applications beyond isogeny-based cryptography remain open questions.

Acknowledgements. We thank Lorenz Panny for helpful comments on a previous version of this paper. Work by the first and third authors was supported by an EPSRC New Investigator grant (EP/S01361X/1).

Bibliography

[1] Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, David Jao, Brian Koziel, Brian LaMacchia, Patrick

Longa, et al. Supersingular isogeny key encapsulation. *Updated parameters for round 2 of NIST Post-Quantum Standardization project*, 2019.

[2] Eric Bach and Lorenz Huelsbergen. Statistical evidence for small generating sets. *mathematics of computation*, 61(203):69–82, 1993.

[3] Xavier Bonnetain and André Schrottenloher. Quantum security analysis of CSIDH. In *Advances in Cryptology - EUROCRYPT 2020*, pages 493–522, 2020.

[4] Paul Bottinelli, Victoria de Quehen, Chris Leonardi, Anton Mosunov, Filip Pawlega, and Milap Sheth. The dark SIDH of isogenies. *IACR Cryptology ePrint Archive*, 2019:1333, 2019. `https://eprint.iacr.org/2019/1333`.

[5] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes. CSIDH: an efficient post-quantum commutative group action. In *Advances in Cryptology - ASIACRYPT 2018*, pages 395–427, 2018.

[6] Denis X Charles, Kristin E Lauter, and Eyal Z Goren. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113, 2009.

[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 dell'equazione $\sum_{h=0}^{n} c_h x^{n-h} y^h = p$. *Giornale di Matematiche di Battaglini*, 46:33–90, 1908.

[9] Jean-Marc Couveignes. Hard homogeneous spaces. *IACR Cryptology ePrint Archive*, 2006:291, 1999.

[10] Luca De Feo. Mathematics of isogeny based cryptography. *arXiv preprint:1711.04062*, 2017.

[11] Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: compact post-quantum signatures from quaternions and isogenies. *IACR Cryptology ePrint Archive*, 2020:1240, 2020.

[12] Luca De Feo and Michael Meyer. Threshold schemes from isogeny assumptions. Cryptology ePrint Archive, Report 2019/1288, 2019. `https://eprint.iacr.org/2019/1288`.

[13] Katalin Friedl, Miklos Santha, Frédéric Magniez, and Pranab Sen. Quantum testers for hidden group properties. *Fundamenta Informaticae*, 91:325–340, 2009.

[14] Steven D. Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. In *Advances in Cryptology - ASIACRYPT 2016*, pages 63–91, 2016.

[15] Steven D Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. *Journal of Cryptology*, 33(1):130–175, 2020.

[16] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. *IACR Cryptology ePrint Archive*, 2017:604, 2017.

[17] David Jao, Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, Brian Koziel, Brian LaMacchia, Patrick

Longa, Michael Naehrig, Joost Renes, Vladimir Soukharev, and David Urbanik. SIKE: Supersingular isogeny key encapsulation. `http://sike.org/`, 2017.

[18] 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.

[19] David Kohel, Kristin Lauter, Christophe Petit, and Jean-Pierre Tignol. On the quaternion ℓ-isogeny path problem. *LMS Journal of Computation and Mathematics*, 17(A):418–432, 2014.

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

[21] Greg Kuperberg. Another subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *arXiv preprint:1112.3333*, 2011.

[22] Péter Kutas, Chloe Martindale, Lorenz Panny, Christophe Petit, and Katherine E. Stange. Weak instances of SIDH variants under improved torsion-point attacks. *IACR Cryptology ePrint Archive*, 2020:633, 2020.

[23] Edmund Landau. *Über die Einteilung der positiven ganzen Zahlen in vier Klassen nach der Mindestzahl der zu ihrer additiven Zusammensetzung erforderlichen Quadrate.* 1909.

[24] Ashley Montanaro and Ronald de Wolf. *A survey of quantum property testing.* Number 7 in Graduate Surveys. Theory of Computing Library, 2016.

[25] Chris Peikert. He gives C-sieves on the CSIDH. In *Advances in Cryptology - EUROCRYPT 2020*, pages 463–492, 2020.

[26] Christophe Petit. Faster algorithms for isogeny problems using torsion point images. In *Advances in Cryptology - ASIACRYPT 2017*, pages 330–353, 2017.

[27] Srinivasa Ramanujan. First letter to G.H. Hardy. 1913.

[28] Oded Regev. A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space. *arXiv preprint:0406151*, 2004.

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

[30] René Schoof. Nonsingular plane cubic curves over finite fields. *J. Comb. Theory, Ser. A*, 46(2):183–211, 1987.

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

[32] Jacques Vélu. Isogénies entre courbes elliptiques. *CR Acad. Sci. Paris, Séries A*, 273:305–347, 1971.

[33] William C Waterhouse. Abelian varieties over finite fields. In *Annales scientifiques de l'École Normale Supérieure*, volume 2, pages 521–560, 1969.

A The orbits of the group action

Recall that in Section 4.2, we have defined a group action on the set of S_P which differs from the group action defined on the sets $S_{P,1}$ and $S_{P,2}$. The

reason for having multiple group actions is that we require them to be free and transitive. Let G be the group which consists of the endomorphisms $a+b\iota$ where $a, b \in \mathbb{Z}/N_1\mathbb{Z}$, b is even and two elements are considered equal if they differ by an odd scalar. Now G acts on all the kernels generated by points of the form $P + \alpha\iota(P)$. We will now study the orbits of this group action in more detail. As we have already seen in Proposition 4.12, the kernels where α is even form a single orbit. Now we show that there are two more orbits, occurring when α is odd. For simplicity we will refer to a kernel generated by $P + \alpha\iota(P)$ by $(1, \alpha)$.

Lemma A.1. *Let α be odd. Then $(1, \alpha)$ is either in the orbit of $(1, 1)$ or $(1, 3)$.*

Proof. First we suppose that $\alpha \equiv 1 \pmod 4$. In this case, $(1, \alpha)$ is in the orbit of $(1, 1)$, and to show this we must prove the existence of an odd λ and an even b such that the following system is satisfied: $\lambda(1 + b) = 1$ and $\lambda(1 - b) = \alpha$.

Solving the system, we find that $\lambda = \frac{1+\alpha}{2}$ and $b = (1 - \lambda)\lambda^{-1}$. These satisfy the required criteria since $1 + \alpha \equiv 2 \pmod 4$, hence λ is odd and $1 - \lambda$ is even.

Now suppose that $\alpha \equiv 3 \pmod 4$. In this case we show that $(1, \alpha)$ is in the orbit of $(1, 3)$. Again there must exist an odd λ and an even b such that both $\lambda(1 + 3b) = 1$ and $\lambda(3 - b) = \alpha$.

Solving gives that $\lambda = \frac{1+3\alpha}{10}$, which is an odd integer because α is congruent to 3 modulo 4 and so $1+3\alpha$ is congruent to 2 modulo 4. Now one has to calculate b which is equal to $\frac{1-\lambda}{3\lambda}$ which is even since λ is odd, proving the second case. □

By Lemma A.1 the group action defined above has three orbits on I. However, the action of G is no longer free on the orbits corresponding to an odd α.

B Generalizing to $N_1 = 3^k$

In this section we sketch a generalization of Section 4.2 to the case where N_1 is a power of 3.

Lemma 4.8 carries over to this case as 9^k can only be written as a sum of two squares in a trivial fashion. Let P be a point such that $\{P, \iota(P)\}$ is a basis of the N_1-torsion. We show that every curve at distance N_1 from E_0 can be reached by an isogeny with a kernel of the form $\langle P + \alpha\iota(P)\rangle$. Let $Q = \beta_1 P + \beta_2\iota(P)$ be a point of order N_1. If β_1 is coprime to 3, then we may multiply Q by an appropriate scalar such that the coordinate of P becomes 1. Suppose that β_1 is divisible by 3. Since Q has order N_1, β_2 is not divisible by 3. Observe that the points Q and $\iota(Q)$ generate isomorphic curves which implies that $\beta_1 P + \beta_2\iota(P)$ and $\iota(Q) = -\beta_2 P + \beta_1\iota(P)$ generate isomorphic curves. Multiplying $\iota(Q)$ with an appropriate scalar, we obtain a kernel generator of the form $P + \alpha\iota(P)$.

However, some curves of the form $E_0/\langle P + \alpha\iota(P)\rangle$ may be pairwise isomorphic. Namely let α be coprime to 3. Then the kernels generated by $P + \alpha\iota(P)$ and $P - \alpha^{-1}\iota(P)$ correspond to isomorphic curves. On the other hand, it is easy to see that α and $-\alpha^{-1}$ are not congruent modulo 3. In particular, all curves at distance N_1 from E_0 can be reached by isogenies with kernels of the form

$P + \alpha\iota(P)$ where α is congruent to 0 or 1 modulo 3. These curves are pairwise non-isomorphic which can be shown by a calculation similar to the one in Section 4.2.

The acting group can be defined in a similar fashion, namely as the endomorphisms of the form $a + b\iota$ where b is divisible by three and two endomorphisms are identified whenever they are the same modulo N_1 up to multiplication by a scalar coprime to N_1. For simplicity we refer to the point $P + \alpha\iota(P)$ as $(1, a)$. Similarly to Appendix A one can check that the action has two orbits:

1. The orbit of $(1, 0)$ consisting of points of the form $(1, x)$, where 3 divides x.
2. The orbit of $(1, 1)$ consisting of points of the form $(1, x)$, where $x \equiv 1 \bmod 3$.

The orbit of $(1, 2)$ contains points of the form $(1, x)$ where x is congruent to 2 modulo 3, but in terms of j-invariants it consists of exactly the same curves as the second orbit.

Since all these orbits have the same cardinality as the acting group, the group action is free and transitive, as required.