Deterministic algorithms for class group actions

Marc Houben

Inria Bordeaux, Institut de Mathématiques de Bordeaux, France
marc.houben@math.u-bordeaux.fr

Abstract. We present an algorithm for the CSIDH protocol that is fully deterministic and strictly constant time. It does not require dummy operations and can be implemented without conditional branches. Our proof-of-concept C implementation shows that a key exchange can be performed in a constant (i.e. fixed) number of finite field operations, independent of the secret keys. The algorithm relies on a technique reminiscent of the standard Montgomery ladder, and applies to the computation of isogenies that divide an endomorphism of smooth degree represented by its kernel. We describe our method in the general context of class group actions on oriented elliptic curves, giving rise to a large family of non-interactive key exchanges different from CSIDH.

1 Introduction

CSIDH [20] is a key exchange protocol based on isogenies between elliptic curves. Its main advantages compared to other post-quantum candidates are its relatively small keys and that it is *non-interactive*. Its main disadvantages are that it is relatively slow, and that it admits subexponential quantum attacks [41,42,57]. Though the exact post-quantum security of CSIDH is still debated [9,10,55,23], this has led most recent work to focus on more conservative (i.e. larger) parameter choices [11,12], severely impacting estimates of its practical performance. Despite this, CSIDH public keys are still more compact than their lattice-based counterparts, even in the high-parameter range [11, Sec. 6]. Perhaps more importantly, the best lattice-based candidates do not provide non-interactive protocols; the first lattice-based non-interactive key exchange that is deemed practical is Swoosh [37], but involves keys several orders of magnitude greater than CSIDH's.

Because of its compact keys and non-interactive nature, CSIDH has enjoyed a considerable amount of attention in the years following its inception, with most research targeting practical performance improvements and resistance against side-channel attacks. The majority of research in the latter category focuses on constant-time implementations, with the first techniques [49] already appearing soon after the initial CSIDH paper (in fact, the original paper [20, Remark. 15] already alludes to an approach based on dummy operations). The current literature on constant-time CSIDH is vast [9,48,52,39,22,38,50,29,23,28,4,11,12]. Approaches may be roughly divided into two categories: those based on *dummy operations*, i.e. useless operations whose results may be discarded, but whose effect is to mask algorithms with a variable running time, and those which are *dummy free*, i.e. without employing dummy operations. Those in the first category, e.g. CTIDH [4] and dCTIDH [12], are typically more efficient, but may be susceptible to *fault injections*: by actively corrupting the algorithm and checking if the output remains unchanged, we can detect which operations are dummy, thus revealing information about the secret key. We note that fault injections typically require a relatively strong attacker model and form its own subject of ongoing research [13,43,45,6,44]. Attempts at constant-time CSIDH in the dummy-free category are typically computationally more expensive; examples are dummy-free OAYT [52] and dCSIDH [11].

The main reason that achieving constant-time CSIDH is a non-trivial problem, is that existing algorithms for evaluating the protocol are inherently *probabilistic*: they require to sample points on elliptic curves of prescribed order;[1] a process that fails with non-zero probability. In fact, sampling points of prescribed order is not only expensive (as it requires multiplying curve points by large integers), but it also makes the protocol fundamentally *non-deterministic*. As a consequence, all existing constant-time implementations have a strictly *variable* running time, but employ various tricks to ensure that the running time does not depend on secret data. The current state-of-the-art constant-time implementations, dCSIDH [11] and dCTIDH [12], attempt to avoid the issue of sampling points entirely, by restricting to a small key space for which the required points of prescribed order may be precomputed. Although the running time of both algorithms does not depend on secret information, they do not fully manage to avoid the issue, since they still require to resample points on the public key curve.

Our Contributions.

- We present an algorithm for evaluating the CSIDH group action that avoids point sampling entirely. It is fully deterministic, strictly constant time, dummy free, and can be implemented without conditional branches. Moreover, it applies to the full CSIDH key space. Our approach does not require advanced techniques and can be seen as a minor modification of the original CSIDH algorithm [20, Alg. 2];
- We present our results in the general setting of class group actions on oriented elliptic curves, giving rise to a larger family of non-interactive key exchange protocols of which CSIDH can be seen as a special case;
- We provide an unoptimized C implementation which executes a full CSIDH-512 key exchange in a constant number of finite field operations, independent of the secret key;
- We provide a proof-of-concept SageMath implementation of a non-interactive key exchange using supersingular elliptic curves with general orientations.

[1] There are other approaches, such as using radical isogenies [17,16,28,36], but these suffer other drawbacks, such as scaling poorly with larger parameters, being expensive to make constant-time (let alone dummy free) [28], and, for larger degrees, not being provably correct [36].

Main idea. We will now describe the main idea behind our approach, which is most easily explained by considering the CSIDH variant CSURF [14]. Let

$$p = 8 \cdot \prod_{i=1}^{n} \ell_i - 1$$

be a prime number, where the ℓ_i denote small odd primes. Consider the supersingular elliptic curve $E/\mathbf{F}_p : y^2 = x^3 - x$. The ring of $\mathbf{F}_p$-rational endomorphisms of E is $\mathcal{O} := \mathrm{End}_{\mathbf{F}_p}(E) = \mathbf{Z}\left[\frac{\pi-1}{2}\right]$, where $\pi : E \to E$ denotes the (p-)Frobenius endomorphism. CSURF works by acting on E with elements of the ideal class group $\mathrm{Cl}(\mathcal{O})$ of the order $\mathcal{O}$. This is done by computing successive ℓ_i-isogenies for various ℓ_i, corresponding to the $\mathcal{O}$-ideals above ℓ_i given by

$$\mathfrak{l}_i = \left(\ell_i, \frac{\pi - 1}{2}\right) \quad \text{and} \quad \overline{\mathfrak{l}_i} = \left(\ell_i, \frac{\pi + 1}{2}\right).$$

Each ℓ_i always gives rise to two ℓ_i-isogenies: the isogeny with kernel $E[\mathfrak{l}_i] := E[\ell_i] \cap E[(\pi - 1)/2]$, which is generated by an ℓ_i-torsion point that is $\mathbf{F}_p$-rational, and the one with kernel $E[\overline{\mathfrak{l}_i}] := E[\ell_i] \cap E[(\pi + 1)/2]$, which is generated by an ℓ_i-torsion point with $\mathbf{F}_p$-rational x-coordinate but (for odd ℓ_i) y-coordinate in $\mathbf{F}_{p^2} \setminus \mathbf{F}_p$. The codomain of these isogenies are $[\mathfrak{l}_i] * E$ and $[\overline{\mathfrak{l}_i}] * E = [\mathfrak{l}_i]^{-1} * E$ respectively. Special about CSURF (compared to CSIDH) is that we can also act with 2-isogenies via the ideals above $\ell_0 := 2$. In the resulting key exchange protocol, the secret key is a vector of exponents $(a_0, \ldots, a_n) \in \mathbf{Z}^n$, and the public key is the curve $E_A := \prod_{i=0}^{n} [\mathfrak{l}_i]^{a_i} E$. It is computed as a chain of ℓ_i-isogenies, by computing a point that generates the kernel of the correct isogeny at each step, and then applying, e.g., Vélu's formulae. To obtain (sufficiently many) such kernel generators is where random point sampling is required. We can slightly cheat by precomputing points $P \in E[(\pi - 1)/2]$ and $Q \in E[(\pi + 1)/2]$ of full order (i.e. order $\prod_{i=0}^{n} \ell_i$) on the base curve, and adding these to the public parameters of the scheme. This allows to perform the action by any ideal class with exponents in $\{-1, 0, 1\}$, as long as we push both points through each of the ℓ_i-isogenies. Since every ℓ_i-isogeny annihilates part of the available ℓ_i-torsion, it seems that eventually (if $|a_i| \geq 2$ for at least one i), we are forced to sample a new point. This is where our trick comes in.

A remarkable property of the CSURF group action, first noticed in [53] and [21, Lem. 8], is that the ideal class corresponding to the exponent vector $(1, \ldots, 1)$ is trivial; in particular, we have $([\mathfrak{l}_2] \cdot \ldots \cdot [\mathfrak{l}_n]) * E = E$. This is because

$$\prod_{i=0}^{n} \mathfrak{l}_i = \left(\frac{\pi - 1}{2}\right) \mathcal{O}$$

as $\mathcal{O}$-ideals.[2] This implies that we may evaluate the action by an ideal class $(a_0, \ldots, a_n)$ with exponents $a_i \in \{0, 1\}$ in two different ways: either as $E' =$

[2] This can be explained by the fact that the algebraic norm of $(\pi - 1)/2$ is $(p+1)/4 = \prod_{i=0}^{n} \ell_i$.

$\prod_{i=0}^{n}[\mathfrak{l}_i]^{a_i} * E$ or as $E' = \prod_{i=0}^{n}[\overline{\mathfrak{l}_i}]^{1-a_i} * E$, where the first isogeny, say $\varphi^+ : E \to E'$, has degree $\prod_{i=0}^{n} \ell_i^{a_i}$, and the second isogeny, say $\varphi^- : E \to E'$, has degree $\prod_{i=0}^{n} \ell_i^{1-a_i}$. Given the points $P \in E[(\pi - 1)/2]$ and $Q \in E[(\pi + 1)/2]$, we can compute φ^+, whose kernel is contained in $E[(\pi - 1)/2]$, from P. Similarly, we can compute φ^-, whose kernel is contained in $E[(\pi + 1)/2]$, from Q. More importantly, though, we have

$$E\left[\frac{\pi - 1}{2}\right] \cap E\left[\frac{\pi + 1}{2}\right] = \{0\},$$

which implies that the image $Q' := \varphi^+(Q) \in E'[(\pi+1)/2]$ still has order $\prod_{i=0}^{n} \ell_i$, and similarly for $P' := \varphi^-(P) \in E'[(\pi - 1)/2]$. Thus, by computing both φ^+ and φ^- and pushing through the corresponding points, we refrain from losing torsion information. Iterating the procedure allows to compute the class group action for any $(a_0, \ldots, a_n) \in \mathbf{Z}^n$. See Figure 1 for an example by picture.

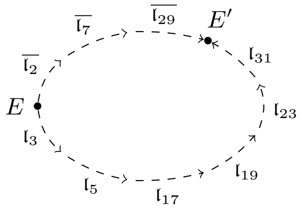

Fig. 1: Acting by the ideal class $(0, 1, 1, 0, 1, 1, 1, 0, 1)$ in "CSURF-32". Here we took $p = 8 \cdot 3 \cdot 5 \cdot 7 \cdot 17 \cdot 19 \cdot 23 \cdot 29 \cdot 31 - 1 \approx 2^{32}$, and $\mathfrak{l}_\ell$ and $\overline{\mathfrak{l}_\ell}$ denote the ideals $(\ell, \frac{\pi-1}{2})$ and $(\ell, \frac{\pi+1}{2})$ respectively. The bottom isogeny chain φ^+ corresponds to the original ideal class (going in the "positive" direction), whereas the top isogeny chain φ^- corresponds to $(-1, 0, 0, -1, 0, 0, 0, -1, 0)$ (going in the "negative" direction).

Note that, since the degrees of φ^+ and φ^- are complementary, this algorithm always computes the action by an ideal class with binary exponents (i.e. in $\{0, 1\}$) at the cost of *one* ℓ_i-isogeny for every ℓ_i. On a high level, and as we will more precisely see in Algorithm 1, this leads naturally to a Montgomery-ladder-style approach for computing the class group action; where the Montgomery ladder is based on the square-and-multiply (or double-and-add) algorithm, we obtain a method which can be described as isogeny-and-multiply.

In Section 3, we will present our algorithms in the general setting of class group actions on elliptic curves primitively oriented by an imaginary quadratic order $\mathcal{O} = \mathbf{Z}[\sigma]$. We will require that the generator σ, viewed as an endomorphism, has smooth degree and is represented either (i) by its kernel, or (ii) as the composition of two isogenies, both given by their kernels. We also need to assume that the kernel of σ and its dual intersect trivially. This applies to CSURF by taking $\sigma = \frac{\pi-1}{2}$, and (with a minor adjustment) to CSIDH by taking $\sigma = \pi - 1$.

Outline. In Section 2 we introduce the necessary background on orientations and their associated class group actions. Section 3 contains the main results and described the algorithms in a general setting. Section 4 specializes to the case of CSIDH, and in Section 5 we consider parameters for other orientations. We point out directions for future research in Section 6.

Acknowledgements We thank Damien Robert for several helpful discussions. We also thank Krijn Reijnders and the anonymous reviewers for their feedback on an earlier version of the paper. This work was financially supported by the France 2030 program, managed by the French National Research Agency under grant agreement No. ANR-22-PETQ-0008 PQ-TLS.

2 Preliminaries

Throughout, we denote by k a perfect field. Our main references are [30,51].

2.1 Isogenies

An isogeny is a non-constant morphism of elliptic curves. Given an elliptic curve E/k, isogenies with domain E are uniquely determined up to post-composition by an isomorphism by their scheme-theoretic kernel, denoted $E[\varphi]$. For separable isogenies, this is the usual (group-theoretic) kernel, consisting of all points of $E(\overline{k})$ that map to the identity element on the codomain. We will often identify separable isogenies with their kernel, even though this only well-defines the underlying map up to post-composition with an isomorphism. Given separable isogenies $\varphi : E \to E_1$ and $\psi : E \to E_2$, the *pushforward* of ψ by φ is the isogeny $\varphi_*(\psi) : E_1 \to E_3$ with kernel $E_1[\varphi_*(\psi)] := \varphi(E[\psi])$. We have $\deg \varphi_*(\psi) = \deg \psi$ if and only if $E[\varphi] \cap E[\psi] = \{0\}$.

2.2 Orientations

Let E/k be an elliptic curve and let K be an imaginary quadratic number field. A *K-orientation* is a (necessarily injective) ring homomorphism

$$\iota : K \to \mathrm{End}^0(E) := \mathrm{End}(E) \otimes_{\mathbf{Z}} \mathbf{Q},$$

where $\mathrm{End}(E) = \mathrm{End}_{\overline{k}}(E)$ denotes the full endomorphism ring of E. The pair (E, ι) is called a *K-oriented elliptic curve*. Given an order $\mathcal{O} \subseteq K$, we call ι an *$\mathcal{O}$-orientation* if $\iota(\mathcal{O}) \subseteq \mathrm{End}(E)$. For any element $\sigma \in \mathcal{O}$, we have that $\deg \iota(\sigma) = N(\sigma)$, where $N : K \to \mathbf{Q}$ denotes the algebraic norm.

An $\mathcal{O}$-orientation is called *primitive* if it does not extend to a strictly larger order. That is, for any strict superorder $\mathcal{O}' \supsetneq \mathcal{O}$ in K, we have that $\iota(\mathcal{O}') \not\subseteq \mathrm{End}(E)$. A K-orientation is a primitive $\mathcal{O}$-orientation for a unique order, called the *primitive order*, given by $\mathcal{O}_{\mathrm{pr}} := \iota^{-1}(\mathrm{End}(E))$.

If (E, ι) is a K-oriented elliptic curve, and $\varphi : E \to E'$ is an isogeny, then we obtain an induced K-orientation $\varphi_*(\iota)$ on E' given by

$$\varphi_*(\iota)(\alpha) := \varphi \circ \iota(\alpha) \circ \hat{\varphi} \otimes \frac{1}{\deg(\varphi)}. \tag{1}$$

Given two K-oriented elliptic curves (E, ι) and (E', ι'), a *K-oriented isogeny* is an isogeny $\varphi : E \to E'$ for which $\varphi_*(\iota) = \iota'$. If $\varphi : (E, \iota) \to (E', \iota')$ is a K-oriented isogeny of prime degree ℓ, then exactly one of the following is true:

(i) $\mathcal{O}_{\mathrm{pr}} = \mathcal{O}'_{\mathrm{pr}}$, in which case φ is called *horizontal*;
(ii) $\mathcal{O}_{\mathrm{pr}} \subsetneq \mathcal{O}'_{\mathrm{pr}}$ and $[\mathcal{O}'_{\mathrm{pr}} : \mathcal{O}_{\mathrm{pr}}] = \ell$, in which case φ is called *ascending*;
(iii) $\mathcal{O}'_{\mathrm{pr}} \subsetneq \mathcal{O}_{\mathrm{pr}}$ and $[\mathcal{O}_{\mathrm{pr}} : \mathcal{O}'_{\mathrm{pr}}] = \ell$, in which case φ is called *descending.*

In general, a K-oriented isogeny of composite degree can be any composition of these three types of isogenies. In case the primitive orders of the domain and codomain are *comparable*, i.e. one is contained in the other, we will use the same terminologies (horizontal, ascending, descending) to describe the corresponding isogeny.

2.3 Class group actions

Let $\mathcal{O} \subseteq K$ be an imaginary quadratic order, and let (E, ι) be an $\mathcal{O}$-oriented elliptic curve. If $\mathfrak{a} \subseteq \mathcal{O}$ is an ideal, we define its kernel by

$$E[\mathfrak{a}] := \bigcap_{\alpha \in \mathfrak{a}} E[\iota(\alpha)],$$

and denote by $\varphi_{\mathfrak{a}} : E \to E'$ an isogeny with kernel $E[\mathfrak{a}]$. The curve $E'/\overline{k}$, unique up to isomorphism, is also denoted $\mathfrak{a} * E$. The induced K-oriented isogeny

$$(E, \iota) \to (\mathfrak{a} * E, (\varphi_{\mathfrak{a}})_*(\iota))$$

is horizontal if and only if $\mathfrak{a}$ is invertible (if $\mathfrak{a}$ is not invertible, then it is ascending). Moreover, if $\mathfrak{b}$ is an equivalent $\mathcal{O}$-ideal, then

$$(\mathfrak{a} * E, (\varphi_{\mathfrak{a}})_*(\iota)) \cong (\mathfrak{b} * E, (\varphi_{\mathfrak{b}})_*(\iota))$$

as K-oriented elliptic curves. This induces an action

$$\mathrm{Cl}(\mathcal{O}) \circlearrowright \{(E, \iota) \mid E/\overline{k} \text{ ell. curve, } \iota \text{ an } \mathcal{O}\text{-orientation}\}/\cong$$

by the ideal class group $\mathrm{Cl}(\mathcal{O})$ of $\mathcal{O}$ on the set of $\mathcal{O}$-oriented elliptic curves over $\overline{k}$ up to K-oriented isomorphism, as soon as the latter set is nonempty. If we further restrict to primitive $\mathcal{O}$-orientations, then this action is free.

By a slight abuse of notation, given an $\mathcal{O}$-oriented curve (E, ι) and an element $\sigma \in \mathcal{O}$, we will denote by $E[\sigma] := E[\iota(\sigma)] = E[(\sigma)]$ the kernel of $\iota(\sigma)$. Similarly, if (N, σ) is an $\mathcal{O}$-ideal, where $N \in \mathbf{Z}_{>0}$, we denote its kernel by $E[N, \sigma] := E[(N, \sigma)]$.

Lemma 2.1. *Let E/k be an elliptic curve, primitively oriented by an imaginary quadratic order $\mathcal{O} = \mathbf{Z}[\sigma]$. Let ℓ be a prime number, different from the characteristic of k, such that $\ell \mid N(\sigma)$. Then*

(i) $E[\ell, \sigma] \cong \mathbf{Z}/\ell\mathbf{Z}$;
(ii) $E[\ell, \sigma] = E[\ell, \hat{\sigma}] \iff \ell \mid \operatorname{tr}(\sigma)$.

Proof. Since $\ell \mid N(\sigma)$, we have that $E[\ell, \sigma] \neq \{0\}$. If $E[\ell, \sigma] = E[\ell]$, then σ factors through $[\ell]$, contradicting the assumption that the orientation is primitive, hence $E[\ell, \sigma]$ is cyclic of order ℓ, proving (i). For (ii), note that $\sigma + \hat{\sigma} = [\operatorname{tr}(\sigma)]$, so $E[\sigma] \cap E[\hat{\sigma}] \subseteq E[\operatorname{tr}(\sigma)]$. It follows that $E[\ell, \sigma] \cap E[\ell, \hat{\sigma}] \neq \{0\}$ if and only if $\ell \mid \operatorname{tr}(\sigma)$. □

Corollary 2.2. *Let E/k be primitively oriented by $\mathbf{Z}[\sigma]$, and suppose that $N(\sigma)$ is coprime to the characteristic of k. Then $E[\sigma]$ is cyclic, and $E[\sigma] \cap E[\hat{\sigma}] = \{0\} \iff \gcd(N(\sigma), \operatorname{tr}(\sigma)) = 1$.*

Proposition 2.3. *Let $\mathcal{O} = \mathbf{Z}[\sigma]$ be an imaginary quadratic order of conductor f. Let $I = (N, \sigma)$ be an $\mathcal{O}$-ideal, where $N \in \mathbf{Z}_{>0}$ and $N \mid N(\sigma)$. Then I is invertible if and only if $\gcd(N, N(\sigma)/N, f) = 1$.*

Proof. Let K be the fraction field of $\mathcal{O}$. Then I is invertible if and only if $\mathcal{O}_L(I) := \{\alpha \in K \mid \alpha I \subseteq I\} = \mathcal{O}$ [31, Cor. 4.4]. Let $\alpha = x + y\sigma \in K$, where $x, y \in \mathbf{Q}$. Then

$$(x + y\sigma)\sigma = y\sigma^2 + x\sigma = -yN(\sigma) + (x + y\operatorname{tr}(\sigma))\sigma.$$

This is an element of I if and only if $x + y\operatorname{tr}(\sigma) \in \mathbf{Z}$ and $N \mid yN(\sigma)$. The latter condition is equivalent to $y \in \frac{1}{N(\sigma)/N}\mathbf{Z}$. Similarly,

$$(x + y\sigma)N = xN + yN\sigma,$$

which is in I if and only if $x \in \mathbf{Z}$ and $y \in \frac{1}{N}\mathbf{Z}$. Combining the conditions, we find that

$$x + y\sigma \in \mathcal{O}_L(I) \iff x \in \mathbf{Z} \text{ and } y \in \frac{1}{N}\mathbf{Z} \cap \frac{1}{N(\sigma)/N}\mathbf{Z} \cap \frac{1}{\operatorname{tr}\sigma}\mathbf{Z}.$$

Therefore I is invertible if and only if $\gcd(N, N(\sigma)/N, \operatorname{tr}(\sigma)) = 1$. Now, let ℓ be a prime number such that $\ell \mid \gcd(N, N(\sigma)/N)$. The desired result will follow if we can show that $\ell \mid \operatorname{tr}(\sigma) \iff \ell \mid f$. Note that, since $\ell \mid \gcd(N, N(\sigma)/N)$, we must have $\ell^2 \mid N(\sigma)$. Thus, if $\ell \mid \operatorname{tr}(\sigma)$, then σ/ℓ is an algebraic integer (indeed, its norm and trace are both integral), hence $\ell \mid f$. Now suppose that $\ell \mid f$. Then $\ell^2 \mid \operatorname{Disc}(\sigma) = \operatorname{tr}(\sigma)^2 - 4N(\sigma)$. Since $\ell^2 \mid N(\sigma)$, it follows that $\ell \mid \operatorname{tr}(\sigma)$, completing the proof. □

Corollary 2.4. *Let E/k be primitively oriented by an imaginary quadratic order $\mathcal{O} = \mathbf{Z}[\sigma]$ of conductor f. Let $K \subseteq E[\sigma]$ be a subgroup of order d. Then the d-isogeny $E \to E/K$ is horizontal if and only if $\gcd(d, N(\sigma)/d, f) = 1$.*

Proof. By Corollary 2.2, we have that $E[\sigma]$ is cyclic, so $K = E[d, \sigma] \subseteq E[\sigma]$ is the unique subgroup of order d. The result follows by applying the proposition to the ideal $I = (d, \sigma)$. □

Example 2.5 (CSIDH). Let $p \equiv 3 \pmod 8$. Let $E/\mathbf{F}_p$ be the supersingular elliptic curve $E : y^2 = x^3 + x$, and denote by $\pi \in \mathrm{End}(E)$ the Frobenius endomorphism. Then E is primitively oriented by $\mathcal{O} = \mathbf{Z}[\pi]$. The minimal polynomial of π is $X^2 + p = 0$, and the maximal order of $\mathbf{Q}(\pi)$ is $\mathbf{Z}\left[\frac{\pi-1}{2}\right]$. The conductor of $\mathcal{O}$ is $f = 2$. Let $\sigma := \pi - 1$. Then $E[\sigma] = E(\mathbf{F}_p)$. The ideal $I = (4, \pi - 1)$ corresponds to the unique subgroup of $E(\mathbf{F}_p)$ of order 4, which is generated by the point $P = (-1, \sqrt{-2})$. We have $N(\sigma) = p + 1 \equiv 4 \pmod 8$, so $\gcd(4, N(\sigma)/4, f) = \gcd(4, (p + 1)/4, 2) = 1$, hence I is invertible and the isogeny $E \to E/\langle P\rangle$ is horizontal. Since $\gcd(2, (p + 1)/2, 2) = 2$, the isogeny $E \to E/\langle 2P\rangle = E/\langle(0, 0)\rangle$ corresponding to the ideal $(2, \pi - 1)$ is not horizontal. Indeed, it is ascending; its codomain $E' : y^2 = x^3 - x$ is oriented by the maximal order. Figure 2 depicts the situation using the (in this case disconnected) 2-*isogeny volcano* [61].

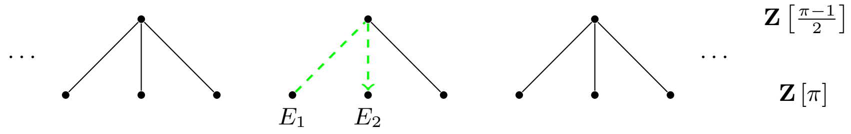

Fig. 2: The supersingular $\mathbf{F}_p$-rational 2-isogeny graph, also known as the 2-isogeny volcano, if $p \equiv 3 \pmod 8$. The curves at the *surface* (i.e. at the top) are primitively oriented by the maximal order, whereas the curves at the *floor* (i.e. at the bottom) are primitively oriented by the order of conductor 2. The dashed 4-isogeny $E_1 \to E_2$ is given by the invertible $\mathbf{Z}[\pi]$-ideal $(4, \pi - 1)$.

Example 2.6 (CSURF). Let $p \equiv 7 \pmod 8$. The supersingular elliptic curves over $\mathbf{F}_p$ (i.e. the supersingular curves over $\overline{\mathbf{F}_p}$ orientable by $\mathbf{Z}[\pi]$) again partition into two subsets, depending on whether the primitive order is $\mathbf{Z}[\pi]$ or $\mathcal{O}_K = \mathbf{Z}\left[\frac{\pi-1}{2}\right]$. In this situation, however, the prime 2 is split (instead of inert) in $\mathcal{O}_K$, which implies that the curves oriented by $\mathcal{O}_K$ admit horizontal 2-isogenies. This leads to an isogeny volcano of the shape depicted in Figure 3.

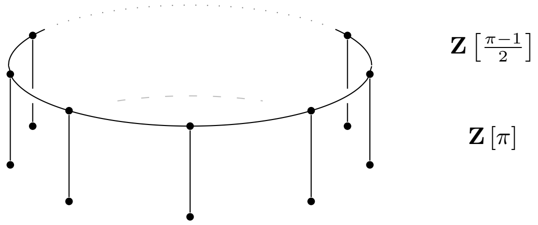

Fig. 3: The supersingular $\mathbf{F}_p$-rational 2-isogeny graph if $p \equiv 7 \pmod 8$.

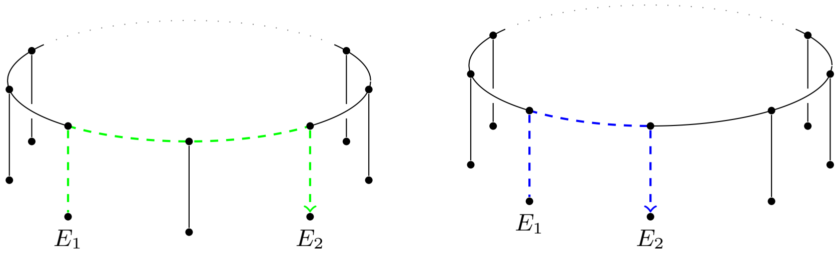

Fig. 4: Let $p \equiv 15 \pmod{32}$ (i.e. $p+1$ is divisible by 16 but not by 32). The dashed isogeny on the left corresponds to the ideal $(16, \pi - 1)$, whereas the one on the right corresponds to the ideal $(8, \pi - 5)$. Note that both ideals are indeed invertible by Proposition 2.3; we have $16 \mid\mid (p+1) = N(\pi - 1)$ and $8 \mid\mid (p+25) = N(\pi - 5)$.

2.4 Representing orientations

Let $E/\mathbf{F}_q$ be an elliptic curve and let $\mathcal{O}$ be an imaginary quadratic order. We say an *effective representation* of an orientation $\iota : \mathcal{O} \hookrightarrow \mathrm{End}(E)$ consists of the following data:

(i) The minimal polynomial $f \in \mathbf{Z}[X]$ of a generator σ of $\mathcal{O}$ (or, equivalently, the norm and trace of σ);
(ii) An algorithm to evaluate $\iota(\sigma)$ on any point $P \in E(\mathbf{F}_{q^r})$ for any $r \in \mathbf{Z}_{>0}$.

There are several ways in which an orientation may be effectively represented. In particular, when it comes to condition (ii), a number of methods exist in the literature. In the case of CSIDH and CRS, we have that $\mathcal{O} = \mathbf{Z}[\pi]$ is generated by the Frobenius endomorphism, which may be represented as an explicit map $E \to E, (x, y) \mapsto (x^p, y^p)$. A generalization to this are (d, ϵ)-structures [26], where an additional d-isogeny from the curve to its Frobenius conjugate is provided. In SCALLOP [35] and PEARL-SCALLOP [2], the orientation is given by the kernels of two (smooth degree) isogenies $\varphi_1 : E \to E_1, \varphi_2 : E \to E_2$ such that $\iota(\sigma) = \hat{\varphi}_2\varphi_1$. SCALLOP-HD [24] uses a two-dimensional representation. More precisely, we are given the images of $\iota(\sigma)$ on $E[2^e]$, together with an element $\tau = a + b\sigma \in \mathcal{O}$ such that $N(\sigma) + N(\tau) = 2^e$. This allows to represent $\iota(\sigma)$ by the $(2^e, 2^e)$-isogeny $E^2 \to E^2$ given by $\begin{pmatrix} \widehat{\iota(\sigma)} & \widehat{\iota(\tau)} \\ -\iota(\tau) & \iota(\sigma) \end{pmatrix}$ with kernel $\{(\iota(\sigma)(R), \iota(\tau)(R)) \mid R \in E[2^e]\}$. In OSIDH [30], on the other hand, public keys do not contain an effective representation of the orientation. In fact, it was shown that knowledge of such a representation renders the protocol insecure [51], and that this leads to a practically feasible (though exponential time) attack [34].

In this work, we consider two (closely related) types of effective representations. In both cases, we assume that the generator σ of $\mathcal{O}$ has smooth norm. Firstly, we study the situation in which we are given the kernel of $\iota(\sigma)$ as a subgroup of E. This applies to CSIDH by taking $\sigma = \pi - 1$. Secondly, we study the case in which $\iota(\sigma)$ is written as the composition of two isogenies of smooth degree, again given by their kernel (or the kernel of the dual). This is the situation of (PEARL-)SCALLOP.

3 Deterministic class group actions

Let k be a perfect field. All elliptic curves in this section are assumed to be defined over k. For practical purposes, the field k will always be a finite field $\mathbf{F}_q$, and for the described algorithms to be practical, all points and isogenies should be defined over $\mathbf{F}_q$ or over a small field extension of this field. What is relevant, however, is that we can efficiently perform field arithmetic, elliptic curve arithmetic (in particular, that we can multiply curve points by scalars), and that we can *compute* cyclic isogenies given a point that generates the kernel, in the sense that we can compute the codomain and evaluate the isogeny at points. In practice this could be realized by Vélu [62] or $\sqrt{\text{é}}$lu [7] formulae.

Let E/k be an elliptic curve, primitively oriented by an imaginary quadratic order $\mathcal{O} = \mathbf{Z}[\sigma]$ of conductor f. Assume that $\gcd(N(\sigma), \operatorname{char} k) = 1$; in particular, the endomorphism $\iota(\sigma)$ is separable. Let $c := \gcd(N(\sigma), f)$ and write $N(\sigma) = c \cdot \prod_{i=1}^{n} \ell_i^{e_i}$ for primes $\ell_i \nmid f$ and $e_i \in \mathbf{Z}_{>0}$. This implies a factorization of the principal $\mathcal{O}$-ideal generated by σ as

$$(\sigma) = (c, \sigma) \cdot \prod_{i=1}^{n} (\ell_i, \sigma)^{e_i} =: \mathfrak{c} \cdot \prod_{i=1}^{n} \mathfrak{l}_i^{e_i}. \tag{2}$$

Note that Proposition 2.3 implies that the ideal $\mathfrak{c} = (c, \sigma)$ is invertible; this ideal will become relevant once we consider the CSIDH protocol in Section 4. For the remainder of this section, however, we will assume that $c = \gcd(N(\sigma), f) = 1$, so that (2) simplifies as

$$(\sigma) = \prod_{i=1}^{n} (\ell_i, \sigma)^{e_i} = \prod_{i=1}^{n} \mathfrak{l}_i^{e_i}. \tag{3}$$

The ideal classes $[\mathfrak{l}_i]$ span a subgroup of the ideal class group $\mathrm{Cl}(\mathcal{O})$ of $\mathcal{O}$. Denote by $\varpi : \mathbf{Z}^n \to \mathrm{Cl}(\mathcal{O})$ the group homomorphism $(s_i)_{i=1}^{n} \mapsto \prod_{i=1}^{n} [\mathfrak{l}_i]^{s_i}$. Note that

$(e_i)_{i=1}^n \in \ker\varpi$; in fact, under a mild condition on the trace of σ, it is "minimal" as an element of $\ker\varpi$ in the following sense.

Proposition 3.1. *Suppose that* $\operatorname{tr}(\sigma)^2 < 4(N(\sigma) - \sqrt{N(\sigma)})$. *Let* $0 \leq s_i \leq e_i$ *be integers. If* $(s_1, \dots, s_n) \in \ker\varpi$, *then either* $s_i = 0$ *for all* i, *or* $s_i = e_i$ *for all* i.

Proof. Suppose to the contrary that $(s_1, \dots, s_n) \in \ker\varpi$ where $s_i \neq 0$ and $s_j \neq e_j$ for some (not necessarily distinct) $1 \leq i, j \leq n$. Then $\prod_{i=1}^n \mathfrak{l}_i^{s_i} = (\tau)$ for some $\tau \in \mathbf{Z}[\sigma]$. Since σ generates $\mathcal{O}$, we have that $\tau \notin \mathbf{Z}$. Possibly replacing $(s_1, \dots, s_n)$ by $(e_1 - s_1, \dots, e_n - s_n)$, we may assume that $N(\tau) \leq \sqrt{N(\sigma)}$. Now

$$4\sqrt{N(\sigma)} < 4N(\sigma) - \operatorname{tr}(\sigma^2) = |\operatorname{Disc}(\sigma)| \leq |\operatorname{Disc}(\tau)| \leq 4N(\tau) \leq 4\sqrt{N(\sigma)},$$

a contradiction. □

Suppose henceforth that $\gcd(N(\sigma), \operatorname{tr}(\sigma)) = 1$, or, equivalently (since $\operatorname{Disc}(\mathcal{O}) = \operatorname{Disc}(\sigma) = \operatorname{tr}(\sigma)^2 - 4N(\sigma)$), that

$$\gcd(N(\sigma), \operatorname{Disc}(\mathcal{O})) = 1. \tag{4}$$

Since $\operatorname{Disc}(\mathcal{O}) = f^2 \operatorname{Disc}(\mathcal{O}_K)$, this is a stronger assumption than $\gcd(N(\sigma), f) = 1$. In particular, it implies that all primes in the factorization (3) are split. By Corollary 2.2, it follows that $E[\sigma]$ and $E[\hat{\sigma}]$ are cyclic and that

$$E[\sigma] \cap E[\hat{\sigma}] = \{0\}. \tag{5}$$

Our main focus will be on the following computational problem.

Problem 3.2. Given an effective representation of the orientation by $\mathbf{Z}[\sigma]$ on E, together with a vector of integers $(s_1, \dots, s_n) \in \mathbf{Z}^n$, compute the elliptic curve $E' := \prod_{i=1}^n \mathfrak{l}_i^{s_i} * E$, together with an effective representation of the (induced) orientation by $\mathbf{Z}[\sigma]$ on E'.

Of course, the nature of this problem depends heavily on *how* the orientation is effectively represented. We consider two main cases. In the first case, we are given generators $P \in E[\sigma]$ and $Q \in E[\hat{\sigma}]$, say, over (a small field extension of) the base field k. In the second, the endomorphism $\iota(\sigma)$ has square degree M^2, and is given as the composition of two isogenies of degree M. Here, we assume that we are given kernels of both isogenies as well as of their duals.

3.1 Given two kernel points

Suppose we are given generators $P \in E[\sigma]$ and $Q \in E[\hat{\sigma}]$. By (5), we have that P and Q form a basis for $E[N]$, where $N := N(\sigma) = \prod_{i=1}^n \ell_i^{e_i}$.

Remark 3.3. One may note that the data of P and Q *overdetermines* the information of a $\mathbf{Z}[\sigma]$-orientation. Indeed, we can compute Q from P using a probabilistic algorithm: if $R \in E[N]$ is a random point, then $\sigma(R) \in E[\hat{\sigma}]$. We can compute the order of $\sigma(R)$ (for example using the Weil pairing with P), and keep sampling points until the lcm of their orders is N. We then obtain Q as a sum of suitable multiples of the $\sigma(R)$. However, our goal is to give a *deterministic* algorithm for Problem 3.2, and our eventual method will rely on the knowledge of both P and Q.

Let $s_1, \ldots, s_n \in \mathbf{Z}$, and suppose that we want to compute the elliptic curve

$$E_A := \left[\prod_{i=1}^{n} \mathfrak{l}_i^{s_i}\right] * E = [\mathfrak{a}] * E,$$

together with a basis for $E_A[N]$ corresponding to a $\mathbf{Z}[\sigma]$-orientation on E_A. Recall that $\varpi : \mathbf{Z}^n \to \mathrm{Cl}(\mathcal{O})$ denotes the map $(s_1, \ldots, s_n) \mapsto [\prod \mathfrak{l}_i^{s_i}]$. By adding a suitable multiple of $(e_1, \ldots, e_n) \in \ker(\varpi)$, we may assume that $s_i \in \mathbf{Z}_{\geq 0}$ for all i. If $0 \leq s_i \leq e_i$ for all i, then computing E_A is straightforward; since $P \in E[\sigma]$ has order $\prod \ell_i^{e_i}$, a suitable multiple K of P has order $\prod \ell_i^{s_i}$. The desired elliptic curve E_A can then be computed as the codomain of an isogeny $\varphi_{\mathfrak{a}} : E \to E/\langle K \rangle \cong E_A$ with kernel $\langle K \rangle$ (for example using Vélu's formulas) as a chain of ℓ_i-isogenies. What is less straightforward, is how to find generators for $E_A[\sigma]$ and $E_A[\hat{\sigma}]$, or how to compute the group action if $s_i > e_i$ for some i. Note that if we can solve the first problem, then we can solve the second, by applying the group action iteratively, so we will focus on the first. The crucial ingredient is the following.

Proposition 3.4. *Let $\mathcal{O} = \mathbf{Z}[\sigma]$ be an imaginary quadratic order, let (E, ι) be an $\mathcal{O}$-oriented elliptic curve, and let $\varphi : E \to E'$ be an isogeny. Assume that both φ and $\iota(\sigma)$ are separable, and that $E[\sigma] \cap E[\varphi] = \{0\}$. Assume also that $\varphi_*(\iota)(\sigma) \in \mathrm{End}(E')$, where $\varphi_*(\iota)$ denotes the induced orientation on E' as defined by (1). Then, up to post-composition by an automorphism of E', we have $\varphi_*(\iota)(\sigma) = \varphi_*(\iota(\sigma))$.*

Proof. We show that both isogenies have the same kernel. By definition of the pushforward, $E'[\varphi_*(\iota(\sigma))] = \varphi(E[\iota(\sigma)])$. Now note that

$$\#\varphi(E[\iota(\sigma)]) = \#E[\iota(\sigma)] = N(\sigma) = \#E'[\varphi_*(\iota)(\sigma)].$$

It thus suffices to prove that $\varphi(E[\iota(\sigma)]) \subseteq E'[\varphi_*(\iota)(\sigma)]$. So let $P \in \varphi(E[\iota(\sigma)])$. Then $P = \varphi(Q)$ for some $Q \in E[\iota(\sigma)]$. We find that

$$\varphi_*(\iota)(\sigma)(P) = \frac{1}{\deg(\varphi)} \varphi \circ \iota(\sigma) \circ \hat{\varphi} \circ \varphi(Q) = \varphi\left(\iota(\sigma)(Q)\right) = \varphi(0) = 0,$$

completing the proof. □

Now, let $\mathfrak{a} = \prod_{i=1}^{n} \mathfrak{l}_i^{s_i}$ be an $\mathcal{O}$-ideal, where $0 \leq s_i \leq e_i$ for all i. The kernel K of the (necessarily horizontal) isogeny $\varphi_{\mathfrak{a}} : E \to E_A$ is contained in $E[\sigma]$, hence intersects $E[\hat{\sigma}]$ trivially. Applying the lemma, it follows that $\varphi_{\mathfrak{a}}(Q)$ generates $E_A[\hat{\sigma}]$. Similarly, the equivalent $\mathcal{O}$-ideal $\mathfrak{b} := \prod_i (\ell_i, \hat{\sigma})^{e_i - s_i}$ corresponds to an isogeny $\varphi_{\mathfrak{b}} : E \to E_A$ whose kernel is contained in $E[\hat{\sigma}]$, hence $\varphi_{\mathfrak{b}}(P)$ generates $E_A[\sigma]$. This gives rise to the following high-level description of an algorithm for evaluating the class group action.

(i) From $P \in E[\sigma]$, compute $K^+ := \left[\prod_{i=1}^n \ell_i^{e_i - s_i}\right] P$ of order $\prod_i \ell_i^{s_i}$.
(ii) From $Q \in E[\hat{\sigma}]$, compute $K^- := [\prod_{i=1}^n \ell_i^{s_i}]\, Q$ of order $\prod_i \ell_i^{e_i - s_i}$.
(iii) Compute $\varphi^+ : E \to E'$ with kernel K^+, and set $Q' := \varphi^+(Q)$.
(iv) Compute $\varphi^- : E \to E'$ with kernel K^-, and set $P' := \varphi^-(P)$.
(v) Return E', P', Q'.

In essence, we have factored the endomorphism $\iota(\sigma)$ as a product of two isogenies $\iota(\sigma) = \widehat{\varphi^-} \circ \varphi^+$. The pushforward of $\iota(\sigma)$ by φ^- defines an orientation by $\mathbf{Z}[\sigma]$ on the codomain, and likewise for the pushforward of $\iota(\hat{\sigma})$ by φ^+ (cf. Figure 5). In practice, the isogenies in steps (iii) and (iv) of the algorithm would each be computed as a chain of ℓ_i-isogenies. Since the total number of ℓ_i-isogenies is always e_i, this can naturally be done in constant time (assuming that the cost of an ℓ_i-isogeny only depends on ℓ_i). The same holds for the point multiplications in steps (i) and (ii), given that they are executed as successive ℓ_i-multiplications (and assuming that the cost of an ℓ_i-multiplication only depends on ℓ_i). In fact, this leads quite naturally to a “Montgomery-ladder”-style algorithm, where the classical square-and-multiply (or double-and-add) is replaced by isogeny-and-multiply. See Algorithm 1 for a complete description of the procedure. Here, the subroutine Eval-ℓ_i-Isogeny denotes an algorithm that takes as input an elliptic curve E, a point $K \in E$ (of order ℓ_i), and two points $P, Q \in E$, and returns the codomain E' of a cyclic ℓ_i-isogeny $\varphi : E \to E' \cong E/\langle K \rangle$ with kernel $\langle K \rangle$, as well as the evaluations $\varphi(P), \varphi(Q)$ of φ at P and Q.

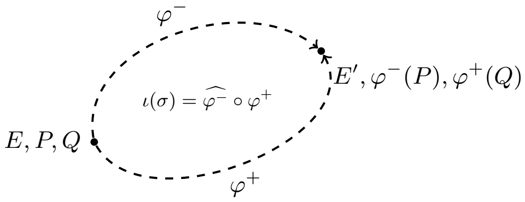
φ^-
$E', \varphi^-(P), \varphi^+(Q)$
$\iota(\sigma) = \widehat{\varphi^-} \circ \varphi^+$
E, P, Q
φ^+

Fig. 5: Evaluating a class group action by factoring an endomorphism.

Remark 3.5. As in the traditional Montgomery ladder, the conditional branch in the inner loop of Algorithm 1 can be replaced by a constant-time conditional swap (cswap) of two memory registers R_0 and R_1, in this case storing (E^+, P^+, Q') and (E^-, P^-, P') respectively, where the cswap-bit is $b \leftarrow j \leq s_i$.

Remark 3.6. The main idea of factoring an endomorphism as in Figure 5 essentially already appears in SCALLOP [35, Prop. 9], but it was not remarked that this leads to a constant-time evaluation of the ideal action.

Algorithm 1 Evaluating a class group action using two kernel points

```
Input: An elliptic curve $E/k$, generators $P \in E[\sigma], Q \in E[\hat{\sigma}]$, a vector of integers $(s_1, \dots, s_n) \in [0, e_i]^n$.
Output: The curve $E' := [\prod_i \mathfrak{l}_i^{s_i}] * E$, generators $P' \in E'[\sigma], Q' \in E'[\hat{\sigma}]$.
  $(E^+, P^+, Q') \leftarrow (E, P, Q)$;                ▷ $P^+ \in E^+[\sigma]$ and $Q' \in E^+[\hat{\sigma}]$.
  $(E^-, P^-, P') \leftarrow (E, Q, P)$;                ▷ $P^- \in E^-[\hat{\sigma}]$ and $P' \in E^-[\sigma]$.
  $m \leftarrow \prod_i \ell_i^{e_i}$;
  for $i = 1, \dots, n$ do
    for $j = 1, \dots, e_i$ do
      if $j \leq s_i$ then
        $m \leftarrow m/\ell_i$; $K \leftarrow [m]P^+$;                ▷ $K$ has order $\ell_i$.
        $(E^+, P^+, Q') \leftarrow$ Eval-$\ell_i$-Isogeny$(E^+, K, P^+, Q')$;   ▷ “Isogeny”
        $P^- \leftarrow [\ell_i]P^-$;                                  ▷ “Multiply”
      else                                ▷ Same, but with $E^+$ and $E^-$ swapped.
        $m \leftarrow m/\ell_i$; $K \leftarrow [m]P^-$;
        $(E^-, P^-, P') \leftarrow$ Eval-$\ell_i$-Isogeny$(E^-, K, P^-, P')$;
        $P^+ \leftarrow [\ell_i]P^+$;
      end if
    end for
  end for
  assert $E^+ = E^-$;
  return $(E^+, P', Q')$;
```

3.2 Given four kernel points

An endomorphism of an elliptic curve can sometimes be more efficiently represented as a composition of two isogenies. This is an idea that is also used in SCALLOP [35] and PEARL-SCALLOP [2], where it is called an “effective representation”. We will consider the case where we are given points $P \in E[\sigma]$ and $Q \in E[\hat{\sigma}]$ such that $\iota(\sigma) = \widehat{\varphi_Q} \circ \varphi_P$, where φ_P, φ_Q denote isogenies with kernels $\langle P \rangle, \langle Q \rangle$ respectively.

Example 3.7. To illustrate the advantage of such a representation over simply giving a generator of $E[\sigma]$, suppose that $E/\mathbf{F}_q$, where $q = p^2$, is a supersingular elliptic curve with $(p+1)^2$ points, so that $E/\mathbf{F}_q \cong \mathbf{Z}/(p+1)\mathbf{Z} \times \mathbf{Z}/(p+1)\mathbf{Z}$. Let $m \in \mathbf{Z}_{>0}$ such that $m \mid (p+1)$, and suppose that $\omega \in \mathrm{End}(E)$ is a cyclic endomorphism of degree m^2. Let P, Q be generators of $E[\omega, m], E[\hat{\omega}, m]$ respectively, so that $\omega = \widehat{\varphi_Q} \circ \varphi_P$. Since $E[m] \subseteq E(\mathbf{F}_q)$, the points P, Q are necessarily defined over $\mathbf{F}_q$, whereas a generator of $E[\omega]$, being of order m^2, may only be defined over a field extension (possibly of large degree).

As we will describe now, Algorithm 1 adapts naturally to this setting. Suppose again that E/k is primitively oriented an imaginary quadratic order $\mathcal{O} = \mathbf{Z}[\sigma]$ of conductor f and that $\gcd(f \operatorname{char} k, N(\sigma)) = 1$, but now, suppose that $N(\sigma) = M^2$ is a perfect square, and let $N(\sigma) = \prod_{i=1}^n \ell_i^{2e_i}$ be its prime factorization. As before, we obtain a factorization of the principal $\mathcal{O}$-ideal generated by σ into prime ideals as

$$(\sigma) = \prod_{i=1}^{n} (\ell_i, \sigma)^{2e_i} = \prod_{i=1}^{n} \mathfrak{l}_i^{2e_i}.$$

Let $P \in E[\sigma, M]$ and $Q \in E[\hat{\sigma}, M]$ be generators. Then $E/\langle P \rangle \cong E/\langle Q \rangle$. We will call this common elliptic curve the *companion curve* (of E), and we will denote it by E_c. Denote by $\varphi^+, \varphi^- : E \to E_c$ the isogenies with kernels $\langle P \rangle, \langle Q \rangle$ respectively; under the map $\varpi : \mathbf{Z}^n \to \mathrm{Cl}(\mathcal{O})$ as before, they correspond to the ideal class $\varpi(e_1, \ldots, e_n) = \varpi(-e_1, \ldots, -e_n)$. The endomorphism $\iota(\sigma)$ is given by $\widehat{\varphi^-} \circ \varphi^+$, and the images of P, Q under φ^-, φ^+ are generators for $E_c[\sigma, M]$ and $E_c[\hat{\sigma}, M]$ respectively, which we will denote by $P_c := \varphi^-(P)$ and $Q_c := \varphi^+(Q)$. A straightforward adaptation of Algorithm 1 is described by Algorithm 2.

Algorithm 2 Evaluating a class group action using four kernel points

```
Input: An elliptic curve $E/\mathbf{F}_q$, generators $P \in E[\sigma, M], Q \in E[\hat{\sigma}, M], P_c \in E_c[\sigma, M], Q_c \in E_c[\hat{\sigma}, M]$, a vector of integers $(s_1, \ldots, s_n) \in [0, e_i]^n$.
Output: The curve $E' := \left[\prod_i \mathfrak{l}_i^{s_i}\right] * E$ and generators $P' \in E'[\sigma, M], Q' \in E'[\hat{\sigma}, M]$.
  $(E^+, P^+, Q') \leftarrow (E, P, Q)$;            ▷ $E^+[\sigma] = \langle P^+ \rangle$ and $E^+[\hat{\sigma}] = \langle Q' \rangle$.
  $(E^-, P^-, P') \leftarrow (E_c, Q_c, P_c)$;      ▷ $E^-[\hat{\sigma}] = \langle P^- \rangle$ and $E^-[\sigma] = \langle P' \rangle$.
  $m \leftarrow \prod_i \ell_i^{e_i}$;
  for $i = 1, \ldots, n$ do
      for $j = 1, \ldots, e_i$ do
          if $j \le s_i$ then
              $m \leftarrow m/\ell_i$; $K \leftarrow [m]P^+$;
              $(E^+, P^+, Q') \leftarrow$ Eval-$\ell_i$-Isogeny$(E^+, K, P^+, Q')$;
              $P^- \leftarrow [\ell_i]P^-$;
          else
              $m \leftarrow m/\ell_i$; $K \leftarrow [m]P^-$;
              $(E^-, P^-, P') \leftarrow$ Eval-$\ell_i$-Isogeny$(E^-, K, P^-, P')$;
              $P^+ \leftarrow [\ell_i]P^+$;
          end if
      end for
  end for
  assert $E^+ = E^-$;
  return $(E^+, P', Q')$;
```

Comparison to Algorithm 1. In contrast with Algorithm 1, the output of Algorithm 2 is *not* of the same form as its input: what is missing is the companion curve E'_c of E', together with generators $P'_c \in E'_c[\sigma, M]$ and $Q'_c \in E'_c[\hat{\sigma}, M]$. However, this data can be computed by running the algorithm again, with the roles of E_c and E swapped. This can be easily seen on the level of ideal classes. Indeed, we have $E' = \varpi(s_1, \ldots, s_n) * E$, so

$$\varpi(s_1, \ldots, s_n)E_c = \varpi(s_1 + e_1, \ldots, s_n + e_n)E = \varpi(e_1, \ldots, e_n)E' = E'_c. \tag{6}$$

See Figure 6 for a graphical representation of the situation.

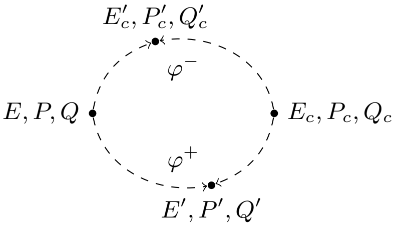

Fig. 6: Factoring the endomorphism $\sigma = \widehat{\varphi^-} \circ \varphi^+$ through its companion curve.

Recovering the full four-point representation $P' \in E'[\sigma, M], Q' \in E'[\hat{\sigma}, M], P'_c \in E'_c[\sigma, M], Q'_c \in E'_c[\hat{\sigma}, M]$ (which is required to apply the group action iteratively) thus costs two evaluations of Algorithm 2. From a high-level point of view, the cost of *one* execution of Algorithm 2 is equivalent to that of computing one isogeny of degree M as a chain of ℓ_i-isogenies (together with some evaluations of these isogenies at points and some elliptic curve arithmetic). More precisely, it consists of a factorization of the M-isogeny φ^+. In this sense, the two executions of Algorithm 2 have a combined cost equivalent to that of *one* evaluation of the (degree M^2) endomorphism $\iota(\sigma)$, which would be the same as the cost of one application of Algorithm 1 if we were granted access to full order generators (i.e. of order M^2) of $E[\sigma]$ and $E[\hat{\sigma}]$ over the base field.

Although the computational cost of evaluating an ideal class corresponding to an endomorphism of a fixed degree is equivalent between the two algorithms, the exponent range in Algorithm 2 is of half the size. This may seem to put Algorithm 2 at a strict computational disadvantage. However, it could be argued that from the point of view of post-quantum security, not the key space, but the size of $\mathrm{Cl}(\mathbf{Z}[\sigma])$ is the most relevant parameter. Algorithm 2 allows to double the value of $N(\sigma)$ (which, in case $\mathrm{tr}(\sigma)$ is chosen small, dictates the size of $|\operatorname{Disc}(\sigma)|$, and hence $\#\,\mathrm{Cl}(\mathbf{Z}[\sigma])$), without increasing the size of the base field or the degrees of the isogeny computations required. In contrast, increasing $N(\sigma)$ in Algorithm 1 forces the available torsion, and with it the size of the base field, to grow.

4 Application: fully deterministic CSIDH

Let p be a prime number of the form $p = 4 \cdot \prod_{i=1}^{n} \ell_i - 1$, where ℓ_i are (small) primes. For example, CSIDH-512 as described in [20] uses the prime number

$$p := 4 \cdot \underbrace{(3 \cdot 5 \cdot \ldots \cdot 373)}_{73 \text{ consecutive primes}} \cdot 587 - 1 \approx 2^{511}.$$

Let E be the supersingular elliptic curve given by $E : y^2 = x^3 + x$. Denote by $\pi \in \mathrm{End}(E_0)$ the p-Frobenius endomorphism. As an algebraic integer, it satisfies the equation $\pi^2 + p = 0$. The curve E is primitively oriented by the order $\mathcal{O} := \mathbf{Z}[\pi]$ of conductor $f = 2$. Consider the generator $\sigma := \pi - 1$ of $\mathcal{O}$. Then $N(\sigma) = p + 1 = 4 \cdot \prod_{i=1}^{n} \ell_i$, corresponding to a factorization (cf. Equation (2))

$$(\sigma) = \mathfrak{c} \cdot \prod_{i=1}^{n} \mathfrak{l}_i, \tag{7}$$

where $\mathfrak{c} := (4, \sigma)$ and $\mathfrak{l}_i := (\ell_i, \sigma)$. The subgroup $E[\sigma]$ is the group $E(\mathbf{F}_p)$ of $\mathbf{F}_p$-rational points. We have $E[\mathfrak{l}_i] = E[\ell_i](\mathbf{F}_p)$ and $E[\mathfrak{c}] = E[4](\mathbf{F}_p)$. Likewise, we denote $\overline{\mathfrak{l}_i} := (\ell_i, \hat{\sigma}) = (\ell_i, \pi + 1)$ and $\overline{\mathfrak{c}} = (4, \hat{\sigma}) = (4, \pi + 1)$ for the conjugate ideals, so that

$$(\hat{\sigma}) = \overline{\mathfrak{c}} \cdot \prod_{i=1}^{n} \overline{\mathfrak{l}_i}.$$

Note that, in this situation, we have $E[\sigma] \cap E[\hat{\sigma}] \neq \{0\}$, so that Algorithm 1 does not immediately apply. This slight inconvenience is a consequence of the fact that CSIDH works with curves at the *bottom* of their 2-isogeny volcano, cf. Example 2.5. Of course, this is not really an issue, since we could instead use a base curve on the surface of the volcano, e.g. the curve given by $y^2 = x^3 - x$, and instead do CSURF [14]. Curves on the surface are oriented by the maximal order $\mathcal{O}_K = \mathbf{Z}[\frac{\pi-1}{2}]$ and the factorization $\left(\frac{\pi-1}{2}\right) = \prod_i (\ell_i, \frac{\pi-1}{2})$ contains only prime ideals of odd norm. In any case, acting on curves on the surface or at the floor of the volcano produces group actions with equivalent security, cf. [14, Rem. 2]. We choose, however, to stick to the original description of the CSIDH protocol, and use an alternative approach; computing the isogeny given by $\mathfrak{c}$ using an explicit formula, rather than from its kernel.

4.1 Adjusting Algorithm 1

Let $0 \leq s_i \leq 1$ be integers, and consider the ideals

$$\mathfrak{a} = \prod_{i=1}^{n} \mathfrak{l}_i^{s_i} \qquad \text{and} \qquad \mathfrak{b} = \prod_{i=1}^{n} \overline{\mathfrak{l}_i}^{1-s_i}.$$

The elliptic curves $E' = [\mathfrak{a}] * E$ and $E'' = [\mathfrak{b}] * E$ are not isomorphic, but they are connected by the ideal $\mathfrak{c}$. Indeed, by Equation (7), we have that $[\mathfrak{c}][\mathfrak{a}][\mathfrak{b}]^{-1} =$

$[(\sigma)] = 1$. In fact, given both E' and E'' in Montgomery form, there is an explicit equation for the 4-isogeny $\varphi_4 : E' \to E''$ given by $\mathfrak{c}$, see e.g. [40, Eq. (21)]. We denote by Eval-4-Isogeny an algorithm that executes this isogeny. More precisely, it takes as input a (Montgomery) curve E and a point $P \in E$, and outputs $\mathfrak{c} * E$ and $\varphi_{\mathfrak{c}}(P)$. This gives rise to Algorithm 3 for evaluating the CSIDH group action. Here, slightly abusing notation, we denote by $E^t(\mathbf{F}_p)$ the group $E[\pi + 1] = E[\hat{\sigma}]$; the x-coordinates of the latter coincide with those of the rational points on a quadratic twist E^t of E.

Algorithm 3 Deterministic CSIDH

Input: An elliptic curve $E/\mathbf{F}_p$, points $P \in E(\mathbf{F}_p), Q \in E^t(\mathbf{F}_p)$ of full odd order, an integer $B \in \mathbf{Z}_{>0}$, a vector of integers $(s_1, \dots, s_n) \in [0, B]^n$.
Output: The curve $E' := [\prod_i \mathfrak{l}_i^{s_i}] * E$, points $P' \in E'(\mathbf{F}_p), Q' \in E'^t(\mathbf{F}_p)$ of full odd order.

```
$(P', Q') \leftarrow (P, Q)$;
for $j = 1, \dots, B$ do
    $(P^+, P^-) \leftarrow (P', Q')$;
    $m \leftarrow \prod_i \ell_i$;
    for $i = 1, \dots, n$ do
        if $j \leq s_i$ then
            $m \leftarrow m/\ell_i$; $K \leftarrow [m]P^+$;
            $(E^+, P^+, Q') \leftarrow$ Eval-$\ell_i$-Isogeny$(E^+, K, P^+, Q')$;
            $P^- \leftarrow [\ell_i]P^-$;
        else
            $m \leftarrow m/\ell_i$; $K \leftarrow [m]P^-$;
            $(E^-, P^-, P') \leftarrow$ Eval-$\ell_i$-Isogeny$(E^-, K, P^-, P')$;
            $P^+ \leftarrow [\ell_i]P^+$;
        end if
    end for
    $(E^-, P') \leftarrow$ Eval-4-Isogeny$(E^-, P')$;
    assert $E^+ = E^-$;
end for
return $(E^+, P', Q')$;
```

Replacing the conditional branch in the inner loop of Algorithm 3 by a cswap, cf. Remark 3.5, we obtain a branch-free, deterministic, dummy-free, constant-time algorithm for evaluating the CSIDH group action. At the expense of adding the points P', Q' to the public key, this gives a *fully* deterministic variant (i.e. without sampling points) of the associated key exchange protocol.

Remark 4.1. Note that even though P', Q' are images of a secret isogeny, the degree of this isogeny is not known (in fact, the degree *is* the secret), hence higher-dimensional isogeny interpolation methods [15,47,59] do not apply to recover the secret isogeny. At the expense of making the protocol slightly non-deterministic, a more cautious approach would be to scale P', Q' by random scalars. An even more cautious approach would be to discard the points P' and

Q' entirely, as they can be resampled anyway. In fact, we can combine this with an Elligator-style [8] sampling as in [11, Sec. 4.2] to compress public keys (i.e., find the smallest Elligator seed $u \in \mathbf{F}_p$ for which the corresponding $\mathbf{F}_p$-rational point has full order, and publish u as part of the public key). In fact, given the points P' and Q', sampling new points of full order is faster, since we can check their order using the Weil pairing [58].

Remark 4.2. Similar to the first proposals for constant-time CSIDH [48], Algorithm 3 restricts to ideal classes with *non-negative* exponents. Assuming the size of the targeted key space is the same, this introduces a computational overhead by a factor of two compared to its variable-time counterpart. For large parameter CSIDH this may not be an issue in practice, since relatively small classical key spaces suffice [11]. Rather, the opposite issue arises: we are forced to compute an ℓ_i-isogeny for every odd prime $\ell_i \mid (p+1)$, which may be expensive for large p.

A proof-of-concept unoptimized implementation in C of the CSIDH protocol as described in Algorithm 3 can be found in the GitHub repository [1]. Its main feature is that it executes a key exchange in a constant (i.e. fixed) number of finite field operations, independent of the secret key. Our code follows the prescription of Algorithm 3 almost exactly, without additional optimizations (such as strategies). Since our implementation is unoptimized, its running time is about twice that of the worst case of the variable-time CSIDH algorithm it is based on [54]. We leave optimizing the computational performance of the algorithm to future work. Nevertheless, in Section 4.2 we give a qualitative comparison to the state of the art on constant-time CSIDH.

4.2 Comparison to existing methods for CSIDH

We will give a comparison of Algorithm 3 to existing constant-time implementations of CSIDH. The literature on this topic is vast, with a significant amount of research dedicated to making the protocol more computationally efficient. Tricks include, but are not limited to, using different curve models [49], radical isogenies [17,16,28,36], (batching) strategies [48,52,38,29,25], Matryoshka isogenies [9,4,12], optimizing addition chains [22], and choosing primes with faster finite field arithmetic [11,12].

From the point of efficiency, the current state of the art are CTIDH [4], dCSIDH [11], and the most recent dCTIDH [12]. Out of these protocols, dCSIDH seems strongest in terms of security: it is constant-time, dummy free, and *almost* deterministic (it still requires random point sampling, but only on the public key curve). dCTIDH is *deterministic* in the same sense as dCSIDH, but uses dummy operations. CTIDH is constant-time, but probabilistic and also uses dummy operations. Although dCTIDH seems superior to CTIDH in any metric (it is even slightly faster), one disadvantage of dCTIDH compared to CTIDH, is that its keys have small ℓ^∞-norm, which gives rise to more memory-efficient attacks [27]. However, since dCTIDH is the current state of the art regarding efficiency, and dCSIDH seems strongest in terms of security, these will be the two main methods that we compare against.

Disadvantages compared to the state of the art. The main disadvantages of Algorithm 3 compared to dCTIDH and dCSIDH are as follows.

(i) Algorithm 3 seems incompatible with the approach of dCTIDH of exploiting the Matryoshka structure [9, Sec. 5.3] of Vélu's formulae; this is a powerful combinatorial trick that creates a larger key space at the cost of fewer isogenies (it is also the crucial ingredient for CTIDH's [4] efficiency).

(ii) To evaluate the group action using Algorithm 3 requires to evaluate an ℓ_i-isogeny of every degree ℓ_i at least once. On the other hand, dCSIDH [11, Sec. 3.2] is free to employ any subset of the ℓ_i in the group action computation. The latter approach gives a possible advantage when the characteristic of the base field is large (i.e. when targeting high levels of quantum security).

Advantages. The main advantages of Algorithm 3 compared to dCTIDH and dCSIDH are as follows.

(i) It requires no random point sampling. Both dCSIDH and dCTIDH require to sample torsion points on the codomain curve after application of the group action. This is not only expensive, but also makes the algorithm, strictly speaking, non-deterministic and non-constant time.[3] Moreover, since random point sampling fails with higher probability for small order torsion points, the dCSIDH protocol avoids using isogenies of the smallest degrees, even though these are the cheapest.

(ii) It is fully dummy-free. The dCTIDH algorithm uses Matryoshka isogenies [12, Sec. 2.4], which is a method that relies inherently on dummy operations. Such operations may be susceptible to attacks based on fault injections.

(iii) Public key validation comes practically for free; see Section 4.3.

(iv) It allows to evaluate the full (restricted effective) class group action. Both dCSIDH and dCTIDH restrict to a key space which is a subset of $\{-1, 0, 1\}^n$, which admits time-memory trade-offs for classical attacks [27]. Moreover, such a small key space means that sufficient classical security can only be guaranteed for larger parameters, thus targeting mainly a pessimistic post-quantum scenario for CSIDH.

(v) It does not depend on the specific structure of Vélu's formulae. CTIDH and dCTIDH's Matryoshka isogenies rely strongly on the structure of Vélu isogeny formulas. Although Vélu and $\sqrt{\text{élu}}$ are the currently most widely

[3] Of course, since point sampling only happens on a publicly known elliptic curve (the public key curve), its running time does not depend on secret information, and hence does not affect the security of the protocol.

employed methods for computing an isogeny from a kernel point, the subroutine Eval-ℓ_i-Isogeny in Algorithm 3 may be replaced by any other such algorithm.

(vi) It does not require constant-time elliptic curve arithmetic. CTIDH and dCTIDH mask the degree of an isogeny computation among a "batch" of different possible degrees (called WOMBats in [12]). Since the degree of the isogeny being computed influences the elliptic curve multiplication that needs to be performed (e.g. the cofactor multiplication to obtain a kernel point of the correct degree), this requires a form of constant-time elliptic curve arithmetic, where multiplications are performed in constant-time independent of the degree in a batch. This is a non-trivial problem that requires precomputing differential addition chains and padding those that perform too well with dummy operations. In Algorithm 3, the point multiplications are independent of the secret key. This allows to precompute and employ optimal differential additions chains for all of the multiplications.

Fault injections. The main difference between dCSIDH and dCTIDH is that dCTIDH uses dummy operations. As noted in advantage (ii), such operations may be susceptible to fault injection attacks. In the CSIDH literature, several papers study fault injections targeting dummy operations [13,43,45,44]. However, this is not the only situation in which fault attacks may apply; another example is disorientation faults [6], which specifically target the random point sampling process. None of these these attacks directly apply to Algorithm 3, since it does not use dummy operations nor random point sampling. We argue, however, that more research is required to measure against different fault attacker models; a good starting point may be to study which fault attacks and countermeasures for the classical Montgomery ladder translate to the setting of Algorithm 3.

4.3 Public key validation

To prevent active attacks against the CSIDH protocol, we are required to *validate* public keys. In case the public key consists only of an elliptic curve, this involves a supersingularity test; see e.g. [5] for a survey of methods specific to CSIDH. If we are additionally given (the x-coordinates of) points P and Q as in Algorithm 3, then we also need to verify that:

(a) we have $P \in E(\mathbf{F}_p)$ and $Q \in E^t(\mathbf{F}_p)$;
(b) both P and Q have order $M = \prod_{i=1}^{n} \ell_i$.

Note that the combination of both claims already implies that E is supersingular. In fact, the curves E and E^t have the same trace of Frobenius up to sign, so if $S \in E(\mathbf{F}_p) \cup E^t(\mathbf{F}_p)$ is a point of order $M \mid (p+1)$, then $M \mid \operatorname{tr}(\pi)$. Since $M = p + 1 > 2\sqrt{p}$ (we certainly assume $p > 2$), it follows that $\operatorname{tr}(\pi) = 0$ by the Hasse–Weil bound.

We argue that with a minor adjustment, Algorithm 3 validates public keys almost automatically. Consider the first iteration of the outer loop of the algorithm, i.e. when $j = 1$. After each iteration of the inner loop, the order of both

P^+ and P^- decreases by a factor of ℓ_i. Hence, if we verify that $P^+ \neq 0 \neq P^-$ at the start of each iteration, and that $P^+ = 0 = P^-$ after the last iteration, then it follows that the input points P and Q were indeed both of order M. Moreover, if the assertion $E^+ = E^-$ succeeds, then P and Q necessarily lie in distinct subgroups by Proposition 3.1.

Remark 4.3. An attentive reader may notice that this does not determine which of P and Q is in $E(\mathbf{F}_p)$ and which is in $E^t(\mathbf{F}_p)$. Although this could be decided by one Legendre symbol computation, it does not appear strictly necessary from the standpoint of active security. Indeed, if we were to apply an ideal class with exponent vector $(s_1, \ldots, s_n)$ to a curve with the points P and Q swapped, then this amounts to replacing s_i by $B - s_i$, which yields an equally valid secret key.

5 General orientations

5.1 Finding practical parameters for a non-interactive key exchange

To set up a class group action by a general imaginary quadratic order for which Algorithms 1 or 2 apply, we first need to find an elliptic curve, assume over a finite field, together with an effective orientation. Ordinary elliptic curves only admit orientations by $\mathbf{Q}(\pi)$, where π is the Frobenius endomorphism, which gives rise to the CRS scheme [33,60]. The problem of finding ordinary elliptic curves for which this orientation is "good" is a different problem outside of the scope of this paper, so we will restrict to the supersingular case. In general, the problem of finding an orientation by a given imaginary quadratic order on a supersingular elliptic curve, even if we know that an orientation exists, is difficult if the endomorphism ring of the curve is unknown; c.f. [3]. If the endomorphism ring is known, then embedding a given imaginary quadratic order essentially comes down to solving a quaternion norm equation, which can be done with Cornacchia's algorithm (modulo local obstructions); we will thus restrict to this case. Let $p \equiv 3 \pmod 4$ be a prime number such that $M := p + 1$ is smooth, say $M = 4 \cdot \prod_{i=1}^{n} \ell_i$. Let $E/\mathbf{F}_p$ be the supersingular elliptic curve given by $E : y^2 = x^3 + x$. Write $\mathbf{F}_q = \mathbf{F}_p(\sqrt{-1})$ for the finite field of $q = p^2$ elements. Denote by $\pi : E \to E, (x, y) \mapsto (x^p, y^p)$ the Frobenius endomorphism, and by $i : E \to E$ the automorphism $(x, y) \mapsto (-x, \sqrt{-1}y)$. Then $\mathrm{End}(E) \cong \mathbf{Z}[1, i, \frac{i+\pi}{2}, \frac{1+i\pi}{2}]$.

Orienting the curve. Say we want to orient E by an imaginary quadratic order $\mathcal{O} = \mathbf{Z}[\sigma]$ of discriminant $\mathrm{Disc}(\mathcal{O})$, or equivalently, we want to find an element of $\mathrm{End}(E)$ with prescribed norm N and trace t (where $t^2 - 4N = \mathrm{Disc}(\mathcal{O})$). For simplicity of the exposition, let us restrict to the case of embedding $\mathcal{O}$ into the suborder $\mathbf{Z}[1, i, \pi, i\pi]$ of $\mathrm{End}(E)$, i.e. we want to find an element $\sigma := a + bi + c\pi + di\pi$ such that $\mathrm{tr}(\sigma) = t$ and $N(\sigma) = N$, where tr and N denote the (reduced) quaternion trace and norm respectively. The trace t of σ dictates the value of a (hence must be necessarily be even). The norm equation now reads

$$N(\sigma) = a^2 + b^2 + p(c^2 + d^2) = N.$$

Regarding this equation modulo p, we find either zero, one, or two solutions for $b \pmod{p}$. Suppose we are in the latter case, and let $b \in \mathbf{Z}_{>0}$ be a lift of such a solution. The final step is to solve $m := \frac{N(\sigma)-(a^2+b^2)}{p} = c^2 + d^2$ for c and d, for which Cornacchia's algorithm may be employed; a solution exists if and only if m is positive and if every odd prime $p \equiv 3 \pmod{4}$ divides m an even number of times (i.e. the exponent of p in the prime factorization of m is even). In practice, we will only consider the situation where m which is prime and $\equiv 1 \pmod{4}$, since this can be verified efficiently. Moreover, as we explain below, we will search for σ of *odd* trace, so that we actually must consider the full quaternion order $\mathbf{Z}[1, i, \frac{i+\pi}{2}, \frac{1+i\pi}{2}]$.

Conditions on the orientation. Assume that the order $\mathcal{O} = \mathbf{Z}[\sigma]$ we orient by is maximal (i.e. of conductor $f = 1$); choosing an order of small conductor > 1 makes the class group larger, but does not benefit the security of the scheme, since we can always reduce to case of the maximal order by computing an ascending isogeny (see e.g. [35, Sec. 7] or [30, Sec. 5.1]). Moreover, if $\mathcal{O}$ is maximal then every $\mathcal{O}$-orientation is automatically primitive.

The size of the class group $\mathrm{Cl}(\mathcal{O}) \approx 0.46|\operatorname{Disc}(\mathcal{O})|^{1/2}$ dictates the (quantum) security of the group action; we thus want it to be large. Note that $\operatorname{Disc}(\mathcal{O}) = \operatorname{tr}(\sigma)^2 - 4N(\sigma)$, so we want to choose $N(\sigma)$ as large as possible and $\operatorname{tr}(\sigma)$ as small as possible. For the first condition, the best we can do is $N(\sigma) = M^2$, assuming that we want to represent σ using elements of $E(\mathbf{F}_q) = E[M]$ (as a composition of two isogenies given by their kernels). Once we have chosen $N(\sigma) = M^2$, we may try successive small traces t and attempt to solve the norm equation using the procedure described above. Note that

$$|\operatorname{Disc}(\mathcal{O})| = |\operatorname{tr}(\sigma)^2 - 4N(\sigma)| = (2M - t)(2M + t). \tag{8}$$

In order for $\mathcal{O}$ to be maximal, we need the discriminant to be square-free, at least away from 2. Moreover, in certain cases, small prime divisors of the discriminant may introduce weaknesses through attacks using self-pairings [19,18], so we prefer to avoid those entirely. Most optimal would thus be the situation where $2M - t$ and $2M + t$ are both prime; if the trace t is small, then both factors are of approximate size $2M$. Summarizing, we obtain the following procedure to find $\sigma = a + bi + c\frac{i+\pi}{2} + d\frac{1+i\pi}{2} \in \operatorname{End}(E)$ with the desired properties.

(i) Let t be the smallest odd prime number different from all ℓ_i;

(ii) Check if $2M - t$ and $2M + t$ are both prime; if not, try the next smallest value of t.

(iii) Solve the norm equation with fixed trace $t = 2a + d$ modulo p, and lift the solution to $\mathbf{Z}$; if there is no solution, start over with the next value of t.

(iv) Rewrite the remaining equation as $c^2 + d^2 = m$ for some integer m. If m is not a prime $\equiv 1 \pmod{4}$, start over with the next value of t.

(v) Solve for c and d using Cornacchia's algorithm, and return the values of $a, b, c, d \in \mathbf{Z}$.

Example 5.1. Let

$$p = 4 \cdot \underbrace{(3 \cdot 5 \cdot \ldots \cdot 373)}_{\text{73 consecutive primes}} \cdot 587 - 1 \approx 2^{511}$$

be the CSIDH-512 prime. Write $M = p+1$. Then $E/\mathbf{F}_p : y^2 = x^3 + x$ admits a (necessarily) primitive orientation by the (maximal) imaginary quadratic order $\mathcal{O}$ of discriminant $\operatorname{Disc}(\mathcal{O}) = (2M+t)(2M-t)$, where $t = 2748817$ and $2M+t$ and $2M-t$ are both prime. Concretely, the orientation sends a generator of $\mathcal{O}$ of trace t (and norm M^2) to the endomorphism $a + bi + c\frac{i+\pi}{2} + d\frac{1+i\pi}{2}$ of E, where[4]

$$\begin{aligned} a &= -47759066608122666336249209644838096351043958589505534723031842577505927695955;\\ b &= 13330115377726222420483962441734730714600237606311941633063749684971410307471\\ &\quad 086777423661614616351159775178954748483073397493132549795640957231967517224295;\\ c &= 104158061203732314501189828191505558995707185611287822116128484535427590155074 0;\\ d &= 95518133216245332672498419289676192702087917179011069446063685155011858140727. \end{aligned}$$

Computing a kernel representation. Once $\sigma \in \operatorname{End}(E)$ has been found, finding a representation as a composition of two isogenies given by their kernel is quite straightforward. Concretely, we want to find generators $P \in E[M, \sigma] \subseteq E(\mathbf{F}_q)$ and $Q \in E[M, \hat{\sigma}] \subseteq \mathbf{F}_q$. To do so, sample random points $R \in E(\mathbf{F}_q) = E[M]$. If $\sigma(R) \neq 0$, then it is a non-trivial element of $E[M, \hat{\sigma}]$. By sampling sufficiently many points and applying the endomorphisms σ and $\hat{\sigma}$ by their explicit description as a map (in terms of the endomorphisms i and π), we obtain generators P and Q as a sum of multiples of the sampled (image) points.

5.2 Non-interactive key exchange

With the effective orientation in place, we are ready to initiate a non-interactive key exchange protocol as follows.

Setup. Firstly, we compute the companion curve E_c, P_c, Q_c from E, P, Q. Recall that, denoting $\varphi^+ : E \to E_c$ the isogeny with kernel $\langle P \rangle$ and by $\varphi^- : E \to E_c$ the isogeny with kernel $\langle Q \rangle$, we have that $P_c = \varphi^-(P)$ and $Q_c = \varphi^+(Q)$. We add E_c, P_c, Q_c to the public parameters of the protocol. Given E, P, Q, E_c, P_c, Q_c, we can evaluate the action on E by any ideal class of the form $\prod_{i=1}^{n} (\ell, \sigma)^{s_i}$, where $0 \leq s_i \leq 1$ for all i, using Algorithm 2 (note that, by assumption on the factorization of M, we are in the case where $e_i = 1$ for all i). This returns $E' = \prod_{i=1}^{n} (\ell, \sigma)^{s_i} E$ together with a representation of the orientation on E' given by $P' \in E[\sigma]$ and $Q' \in E[\hat{\sigma}]$. Recall from Equation (6) that by acting by the same ideal class on E_c, P_c, Q_c, we obtain the companion curve E'_c to E'. We can thus compute $E', P', Q', E'_c.P'_c, Q'_c$ using two applications of Algorithm 2, at a cost roughly equivalent to one evaluation of σ.

[4] It may seem remarkable that the size of b is about twice the size of the other constants. This is explained by the fact that it comes from the solution to a quadratic equation modulo p, specifically step (iii) of the algorithm, hence we expect its order of magnitude to be about p. The other constants are dominated by the solution to the Cornacchia equation $c^2 + d^2 = m$, where m has size about p, so they have size about $\sqrt{p}$.

The protocol. Let $B \in \mathbf{Z}_{>0}$ denote a bound on the exponent range of the ideals used; the private key space will be of size $(B+1)^n$.

1. Alice's private key is a sequence of integers $(a_1, \dots, a_n) \in \{0, \dots, B\}^n$. She computes $E^A = \prod_{i=1}^n [(\ell_i, \sigma)]^{a_i} E$ together with $P^A \in E^A[\sigma]$ and $Q^A \in E^A[\hat{\sigma}]$ as well as $E_c^A = \prod_{i=1}^n [(\ell_i, \sigma)]^{a_i} E_c$ together with $P_c^A \in E_c^A[\sigma]$ and $Q_c^A \in E_c^A[\hat{\sigma}]$ using $2B$ applications of Algorithm 2. She sends E^A, P^A, Q^A as well as E_c^A, P_c^A, Q_c^A to Bob.
2. Bob's private key is a sequence of integers $(b_1, \dots, b_n) \in \{0, \dots, B\}^n$. He computes $E^B = \prod_{i=1}^n [(\ell_i, \sigma)]^{b_i} E$ together with $P^B \in E^B[\sigma]$ and $Q^B \in E^B[\hat{\sigma}]$ as well as $E_c^B = \prod_{i=1}^n [(\ell_i, \sigma)]^{b_i} E_c$ together with $P_c^B \in E_c^B[\sigma]$ and $Q_c^B \in E_c^B[\hat{\sigma}]$ using $2B$ applications of Algorithm 2. He sends E^B, P^B, Q^B as well as E_c^B, P_c^B, Q_c^B to Alice.
3. Alice computes $E^{AB} = \prod_{i=1}^n [(\ell_i, \sigma)]^{a_i} E^B$, using $2B$ applications of Algorithm 2. Her shared secret is the j-invariant $j(E^{AB})$ of E^{AB},
4. Bob computes $E^{BA} = \prod_{i=1}^n [(\ell_i, \sigma)]^{b_i} E^A$ using $2B$ applications of Algorithm 2. His shared secret is $j(E^{BA}) = j(E^{AB})$.

A few remarks about the described protocol are in place.

(i) Although Bob requires E_c^A, P_c^A, Q_c^A to execute Algorithm 2, they may be recomputed from E^A, P^A, Q^A (at the cost of exactly two applications of Algorithm 2). At the expense of slightly increasing the cost of the shared key derivation step, this data may thus be removed from the public key.

(ii) The data E^A, P^A, Q^A is essentially an SIDH public key, which may be further compressed using well-known techniques such as pairings [32].

(iii) Compared to CSIDH, the protocol works over $\mathbf{F}_{p^2}$ instead of $\mathbf{F}_p$, but only requires a prime p of *half* the size for an equally sized class group. This makes the size of the base field equivalent to CSIDH's in terms of post-quantum security.

(iv) Compared to CSIDH, the classical key space is smaller in comparison to the size of the base field; it only depends on the characteristic p (i.e. the number of ℓ_i in the factorization of $M = p + 1$). If post-quantum security (i.e. resistance to Kuperberg's algorithm) is the bottleneck, this may not be a practical issue; cf. [11, Sec. 3.1].

(v) Since the shared secret only depends on E^{AB} and not on E_c^{AB}, one of the applications of Algorithm 2 in the shared key derivation step may be skipped (the very last one), making the actual cost equal to $2B - 1$ applications of Algorithm 2.

(vi) If Alice and Bob both use Vélu's formulae, they can ensure that their secret isogenies commute as maps (not just up to isomorphism), by [46, Thm. 3.1][5]. This guarantees that the points $P^{AB}, P^{BA} \in E^{AB}[\sigma]$ are equal, and similarly for $Q^{AB}, Q^{BA} \in E^{AB}[\hat{\sigma}]$. These points, which represent a $\mathbf{Z}[\sigma]$-orientation on E^{AB}, may thus be considered part the shared secret.

[5] The reference only concerns the case where the degrees of the isogenies are coprime, but the proof does not seem to require this assumption.

We have implemented the parameter search described in Section 5.1 and an unoptimized proof-of-concept of the key exchange protocol in SageMath, using the library [56] for Kummer line arithmetic. The associated code can be found in the GitHub repository [1].

6 Future work

We highlight the following possible directions for future work.

(a) Re-analyze the quantum oracle cost for CSIDH. The cost of evaluating the CSIDH group action on a quantum computer is difficult to estimate and has been subject of debate since CSIDH's origin. The main difficulty is that evaluating the class group action in quantum superposition requires a constant-time, fully deterministic algorithm without conditional branches. In fact, the main bottleneck in the comprehensive analysis given by [9] is that random point sampling fails with non-zero probability, and with different probabilities for every degree ℓ_i. This induces a large overhead on the quantum circuit in order to guarantee a quantum algorithm with sufficiently low overall failure rate, see [9, Sec. 6 & 7]. Algorithm 3 simplifies the situation considerably, since it naturally satisfies all properties required for evaluation on a quantum computer.[6] In fact, it gives a quantum algorithm with a failure rate of zero. It would be interesting to study the precise gain, together with an updated analysis of more recent tricks such as strategies and $\sqrt{\text{élu}}$, in the quantum oracle cost.[7]

(b) Study the practical performance of Algorithm 3. In view of Remark 4.2, we expect that Algorithm 3 performs best for primes p for which the number of ℓ_i matches the (bit)size of the classical key space.

(c) Study the practicality of *higher-order factorizations* of endomorphisms. We have considered (Section 3.1 and 3.2) the cases where $\iota(\sigma)$ is the composition of either one or two isogenies explicitly given by their kernel. A generalization would be to consider an endomorphism of degree M^r for some $r \in \mathbf{Z}_{>0}$, given as a composition of r isogenies of degree M. The advantage of such an approach could be similar to the one described in Example 3.7; to increase the size of the orientation without increasing the size of the base field. The size of the public key (which now requires at least r curves together with a kernel point), as well as the computational cost of the protocol, is linear in r. The size of the key space remains constant, but the size of the class group (and hence the quantum security of the group action) also grows linearly with r. At larger parameters, this may be more efficient than the current approach of increasing the size of the base field.

[6] Note that dCSIDH and dCTIDH cannot be readily implemented as a quantum oracle, since they still require random point sampling to evaluate exponent vectors outside of the range $\{-1, 0, 1\}^n$.

[7] A rough estimate building on [9] that includes $\sqrt{\text{élu}}$ is given in [23, Sec. 3.6].

References

1. GitHub repository associated to this paper, https://github.com/houbenmr/DeterministicClassGroupActions
2. Allombert, B., Biasse, J.F., Eriksen, J.K., Kutas, P., Leonardi, C., Page, A., Scheidler, R., Bagi, M.T.: PEARL-SCALLOP: Parameter Extension Applicable in Real-Life SCALLOP. Cryptology ePrint Archive, Report 2024/1744 (2024), https://eprint.iacr.org/2024/1744
3. Arpin, S., Clements, J., Dartois, P., Eriksen, J.K., Kutas, P., Wesolowski, B.: Finding orientations of supersingular elliptic curves and quaternion orders. DCC **92**(11), 3447–3493 (2024). https://doi.org/10.1007/s10623-024-01435-5
4. Banegas, G., Bernstein, D.J., Campos, F., Chou, T., Lange, T., Meyer, M., Smith, B., Sotáková, J.: CTIDH: faster constant-time CSIDH. IACR TCHES **2021**(4), 351–387 (2021). https://doi.org/10.46586/tches.v2021.i4.351-387, https://tches.iacr.org/index.php/TCHES/article/view/9069
5. Banegas, G., Gilchrist, V., Smith, B.: Efficient supersingularity testing over $\mathbb{F}_p$ and CSIDH key validation. Cryptology ePrint Archive, Report 2022/880 (2022), https://eprint.iacr.org/2022/880
6. Banegas, G., Krämer, J., Lange, T., Meyer, M., Panny, L., Reijnders, K., Sotáková, J., Trimoska, M.: Disorientation faults in CSIDH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 310–342. Springer, Cham (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_11
7. Bernstein, D.J., De Feo, L., Leroux, A., Smith, B.: Faster computation of isogenies of large prime degree. Cryptology ePrint Archive, Report 2020/341 (2020), https://eprint.iacr.org/2020/341
8. Bernstein, D.J., Hamburg, M., Krasnova, A., Lange, T.: Elligator: elliptic-curve points indistinguishable from uniform random strings. In: Sadeghi, A.R., Gligor, V.D., Yung, M. (eds.) ACM CCS 2013. pp. 967–980. ACM Press (Nov 2013). https://doi.org/10.1145/2508859.2516734
9. Bernstein, D.J., Lange, T., Martindale, C., Panny, L.: Quantum circuits for the CSIDH: Optimizing quantum evaluation of isogenies. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part II. LNCS, vol. 11477, pp. 409–441. Springer, Cham (May 2019). https://doi.org/10.1007/978-3-030-17656-3_15
10. Bonnetain, X., Schrottenloher, A.: Quantum security analysis of CSIDH. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 493–522. Springer, Cham (May 2020). https://doi.org/10.1007/978-3-030-45724-2_17
11. Campos, F., Chávez-Saab, J., Chi-Domínguez, J.J., Meyer, M., Reijnders, K., Rodríguez-Henríquez, F., Schwabe, P., Wiggers, T.: Optimizations and practicality of high-security CSIDH. CiC **1**(1), 5 (2024). https://doi.org/10.62056/anjbksdja
12. Campos, F., Hellenbrand, A., Meyer, M., Reijnders, K.: dCTIDH: Fast & Deterministic CTIDH. Cryptology ePrint Archive, Report 2025/107 (2025), https://eprint.iacr.org/2025/107
13. Campos, F., Kannwischer, M.J., Meyer, M., Onuki, H., Stöttinger, M.: Trouble at the CSIDH: Protecting CSIDH with dummy-operations against fault injection attacks. Cryptology ePrint Archive, Report 2020/1005 (2020), https://eprint.iacr.org/2020/1005
14. Castryck, W., Decru, T.: CSIDH on the surface. In: Ding, J., Tillich, J.P. (eds.) Post-Quantum Cryptography - 11th International Conference,

PQCrypto 2020. pp. 111–129. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-44223-1_7

15. Castryck, W., Decru, T.: An efficient key recovery attack on SIDH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 423–447. Springer, Cham (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_15
16. Castryck, W., Decru, T., Houben, M., Vercauteren, F.: Horizontal racewalking using radical isogenies. In: Agrawal, S., Lin, D. (eds.) ASIACRYPT 2022, Part II. LNCS, vol. 13792, pp. 67–96. Springer, Cham (Dec 2022). https://doi.org/10.1007/978-3-031-22966-4_3
17. Castryck, W., Decru, T., Vercauteren, F.: Radical isogenies. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part II. LNCS, vol. 12492, pp. 493–519. Springer, Cham (Dec 2020). https://doi.org/10.1007/978-3-030-64834-3_17
18. Castryck, W., Houben, M., Merz, S.P., Mula, M., van Buuren, S., Vercauteren, F.: Weak instances of class group action based cryptography via self-pairings. In: Handschuh, H., Lysyanskaya, A. (eds.) CRYPTO 2023, Part III. LNCS, vol. 14083, pp. 762–792. Springer, Cham (Aug 2023). https://doi.org/10.1007/978-3-031-38548-3_25
19. Castryck, W., Houben, M., Vercauteren, F., Wesolowski, B.: On the decisional Diffie-Hellman problem for class group actions on oriented elliptic curves. Cryptology ePrint Archive, Report 2022/345 (2022), https://eprint.iacr.org/2022/345
20. Castryck, W., Lange, T., Martindale, C., Panny, L., Renes, J.: CSIDH: An efficient post-quantum commutative group action. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part III. LNCS, vol. 11274, pp. 395–427. Springer, Cham (Dec 2018). https://doi.org/10.1007/978-3-030-03332-3_15
21. Castryck, W., Panny, L., Vercauteren, F.: Rational isogenies from irrational endomorphisms. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 523–548. Springer, Cham (May 2020). https://doi.org/10.1007/978-3-030-45724-2_18
22. Cervantes-Vázquez, D., Chenu, M., Chi-Domínguez, J.J., De Feo, L., Rodríguez-Henríquez, F., Smith, B.: Stronger and faster side-channel protections for CSIDH. In: Schwabe, P., Thériault, N. (eds.) LATINCRYPT 2019. LNCS, vol. 11774, pp. 173–193. Springer, Cham (Oct 2019). https://doi.org/10.1007/978-3-030-30530-7_9
23. Chávez-Saab, J., Chi-Domínguez, J.J., Jaques, S., Rodríguez-Henríquez, F.: The SQALE of CSIDH: sublinear Vélu quantum-resistant isogeny action with low exponents. Journal of Cryptographic Engineering **12**(3), 349–368 (Sep 2022). https://doi.org/10.1007/s13389-021-00271-w
24. Chen, M., Leroux, A., Panny, L.: SCALLOP-HD: Group action from 2-dimensional isogenies. In: Tang, Q., Teague, V. (eds.) PKC 2024, Part II. LNCS, vol. 14603, pp. 190–216. Springer, Cham (Apr 2024). https://doi.org/10.1007/978-3-031-57725-3_7
25. Cheng, H., Fotiadis, G., Großschädl, J., Ryan, P.Y.A., Rønne, P.B.: Batching CSIDH group actions using AVX-512. IACR TCHES **2021**(4), 618–649 (2021). https://doi.org/10.46586/tches.v2021.i4.618-649, https://tches.iacr.org/index.php/TCHES/article/view/9077
26. Chenu, M., Smith, B.: Higher-degree supersingular group actions. Cryptology ePrint Archive, Report 2021/955 (2021), https://eprint.iacr.org/2021/955
27. Chi-Domínguez, J.J., Esser, A., Kunzweiler, S., May, A.: Low memory attacks on small key CSIDH. In: Tibouchi, M., Wang, X. (eds.) ACNS 23International Conference on Applied Cryptography and Network Security, Part II. LNCS,

vol. 13906, pp. 276–304. Springer, Cham (Jun 2023). https://doi.org/10.1007/978-3-031-33491-7_11

28. Chi-Domínguez, J.J., Reijnders, K.: Fully projective radical isogenies in constant-time. In: Galbraith, S.D. (ed.) CT-RSA 2022. LNCS, vol. 13161, pp. 73–95. Springer, Cham (Mar 2022). https://doi.org/10.1007/978-3-030-95312-6_4
29. Chi-Domínguez, J.J., Rodríguez-Henríquez, F.: Optimal strategies for CSIDH. Cryptology ePrint Archive, Report 2020/417 (2020), https://eprint.iacr.org/2020/417
30. Colò, L., Kohel, D.: Orienting supersingular isogeny graphs. Journal of Mathematical Cryptology, vol. 14, pp. 414–437 (2020). https://doi.org/10.1515/jmc-2019-0034
31. Conrad, K.: The conductor ideal of an order, https://kconrad.math.uconn.edu/blurbs/gradnumthy/conductor.pdf
32. Costello, C., Jao, D., Longa, P., Naehrig, M., Renes, J., Urbanik, D.: Efficient compression of SIDH public keys. In: Coron, J.S., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part I. LNCS, vol. 10210, pp. 679–706. Springer, Cham (Apr / May 2017). https://doi.org/10.1007/978-3-319-56620-7_24
33. Couveignes, J.M.: Hard homogeneous spaces. Cryptology ePrint Archive, Report 2006/291 (2006), https://eprint.iacr.org/2006/291
34. Dartois, P., De Feo, L.: On the security of OSIDH. In: Hanaoka, G., Shikata, J., Watanabe, Y. (eds.) PKC 2022, Part I. LNCS, vol. 13177, pp. 52–81. Springer, Cham (Mar 2022). https://doi.org/10.1007/978-3-030-97121-2_3
35. De Feo, L., Fouotsa, T.B., Kutas, P., Leroux, A., Merz, S.P., Panny, L., Wesolowski, B.: SCALLOP: Scaling the CSI-FiSh. In: Boldyreva, A., Kolesnikov, V. (eds.) PKC 2023, Part I. LNCS, vol. 13940, pp. 345–375. Springer, Cham (May 2023). https://doi.org/10.1007/978-3-031-31368-4_13
36. Decru, T.: Radical $\sqrt[N]{\text{élu}}$ isogeny formulae. In: Reyzin, L., Stebila, D. (eds.) CRYPTO 2024, Part V. LNCS, vol. 14924, pp. 107–128. Springer, Cham (Aug 2024). https://doi.org/10.1007/978-3-031-68388-6_5
37. Gajland, P., de Kock, B., Quaresma, M., Malavolta, G., Schwabe, P.: SWOOSH: Efficient lattice-based non-interactive key exchange. In: Balzarotti, D., Xu, W. (eds.) USENIX Security 2024. USENIX Association (Aug 2024)
38. Hutchinson, A., LeGrow, J.T., Koziel, B., Azarderakhsh, R.: Further optimizations of CSIDH: A systematic approach to efficient strategies, permutations, and bound vectors. In: Conti, M., Zhou, J., Casalicchio, E., Spognardi, A. (eds.) ACNS 20International Conference on Applied Cryptography and Network Security, Part I. LNCS, vol. 12146, pp. 481–501. Springer, Cham (Oct 2020). https://doi.org/10.1007/978-3-030-57808-4_24
39. Jalali, A., Azarderakhsh, R., Kermani, M.M., Jao, D.: Towards optimized and constant-time CSIDH on embedded devices. In: Polian, I., Stöttinger, M. (eds.) COSADE 2019. LNCS, vol. 11421, pp. 215–231. Springer, Cham (Apr 2019). https://doi.org/10.1007/978-3-030-16350-1_12
40. Jao, D., De Feo, L.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Yang, B.Y. (ed.) Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011. pp. 19–34. Springer, Berlin, Heidelberg (Nov / Dec 2011). https://doi.org/10.1007/978-3-642-25405-5_2
41. Kuperberg, G.: A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. SIAM Journal on Computing **35**(1), 170–188 (2005)
42. Kuperberg, G.: Another subexponential-time quantum algorithm for the dihedral hidden subgroup problem. In: 8th Conference on the Theory of Quantum

Computation, Communication and Cryptography (TQC 2013). Leibniz International Proceedings in Informatics (LIPIcs), vol. 22, pp. 20–34 (2013). https://doi.org/10.4230/LIPIcs.TQC.2013.20

43. LeGrow, J., Hutchinson, A.: An analysis of fault attacks on CSIDH. Cryptology ePrint Archive, Report 2020/1006 (2020), https://eprint.iacr.org/2020/1006
44. LeGrow, J.T.: A faster method for fault attack resistance in static/ephemeral CSIDH. Journal of Cryptographic Engineering **13**(3), 283–294 (Sep 2023). https://doi.org/10.1007/s13389-023-00318-0
45. LeGrow, J.T., Hutchinson, A.: (Short paper) analysis of a strong fault attack on static/ephemeral CSIDH. In: Nakanishi, T., Nojima, R. (eds.) IWSEC 21. LNCS, vol. 12835, pp. 216–226. Springer, Cham (Sep 2021). https://doi.org/10.1007/978-3-030-85987-9_12
46. Leonardi, C.: A note on the ending elliptic curve in SIDH. Cryptology ePrint Archive, Report 2020/262 (2020), https://eprint.iacr.org/2020/262
47. Maino, L., Martindale, C., Panny, L., Pope, G., Wesolowski, B.: A direct key recovery attack on SIDH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 448–471. Springer, Cham (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_16
48. Meyer, M., Campos, F., Reith, S.: On lions and elligators: An efficient constant-time implementation of CSIDH. In: Ding, J., Steinwandt, R. (eds.) Post-Quantum Cryptography - 10th International Conference, PQCrypto 2019. pp. 307–325. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-25510-7_17
49. Meyer, M., Reith, S.: A faster way to the CSIDH. In: Chakraborty, D., Iwata, T. (eds.) INDOCRYPT 2018. LNCS, vol. 11356, pp. 137–152. Springer, Cham (Dec 2018). https://doi.org/10.1007/978-3-030-05378-9_8
50. Moriya, T., Onuki, H., Takagi, T.: How to construct CSIDH on Edwards curves. In: Jarecki, S. (ed.) CT-RSA 2020. LNCS, vol. 12006, pp. 512–537. Springer, Cham (Feb 2020). https://doi.org/10.1007/978-3-030-40186-3_22
51. Onuki, H.: On oriented supersingular elliptic curves. Finite Fields and Their Applications (2021), https://doi.org/10.1016/j.ffa.2020.101777
52. Onuki, H., Aikawa, Y., Yamazaki, T., Takagi, T.: (Short paper) A faster constant-time algorithm of CSIDH keeping two points. In: Attrapadung, N., Yagi, T. (eds.) IWSEC 19. LNCS, vol. 11689, pp. 23–33. Springer, Cham (Aug 2019). https://doi.org/10.1007/978-3-030-26834-3_2
53. Onuki, H., Takagi, T.: On collisions related to an ideal class of order 3 in CSIDH. In: Aoki, K., Kanaoka, A. (eds.) IWSEC 20. LNCS, vol. 12231, pp. 131–148. Springer, Cham (Sep 2020). https://doi.org/10.1007/978-3-030-58208-1_8
54. Panny, L.: Open-source implementation of CSIDH, https://yx7.cc/code/csidh/csidh-20180826.tar.xz
55. Peikert, C.: He gives C-sieves on the CSIDH. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 463–492. Springer, Cham (May 2020). https://doi.org/10.1007/978-3-030-45724-2_16
56. Pope, G.: An implementation of isogenies between Kummer Lines of Montgomery curves using x-only arithmetic, https://github.com/GiacomoPope/KummerIsogeny
57. Regev, O.: A subexponential time algorithm for the dihedral hidden subgroup problem with polynomial space (2004), https://arxiv.org/pdf/quant-ph/0406151
58. Reijnders, K.: Effective pairings in isogeny-based cryptography. In: Aly, A., Tibouchi, M. (eds.) LATINCRYPT 2023. LNCS, vol. 14168, pp. 109–128. Springer, Cham (Oct 2023). https://doi.org/10.1007/978-3-031-44469-2_6

59. Robert, D.: Breaking SIDH in polynomial time. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 472–503. Springer, Cham (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_17
60. Rostovtsev, A., Stolbunov, A.: Public-Key Cryptosystem Based On Isogenies. Cryptology ePrint Archive, Report 2006/145 (2006), https://eprint.iacr.org/2006/145
61. Sutherland, A.: Isogeny volcanoes. The Open Book Series **1**(1), 507–530 (Nov 2013). https://doi.org/10.2140/obs.2013.1.507
62. Vélu, J.: Isogénies entre courbes elliptiques. Comptes-Rendus de l'Académie des Sciences **273**, 238–241 (1971)