OPEN ACCESS
Journal of Algebra and Its Applications
(2025) 2530002 (36 pages)
© The Author(s)
DOI: 10.1142/S0219498825300028

World Scientific
www.worldscientific.com

# A review of mathematical and computational aspects of CSIDH algorithms

Luciano Maino
*University of Bristol, UK*
*luciano.maino@bristol.ac.uk*

Marzio Mula*
*University of Trento, Italy*
*marzio.mula@unitn.it*

Federico Pintore
*University of Bari, Italy*
*federico.pintore@gmail.com*

Received 24 August 2022
Revised 16 May 2023
Accepted 7 September 2023
Published 3 November 2023

Communicated by E. Gorla

CSIDH is a post-quantum key-exchange scheme based on the action of ideal class groups on supersingular elliptic curves over prime fields. Its short keys and ciphertexts, together with its flexibility as a building block to construct complex cryptographic primitives, has motivated significant research on the efficiency of CSIDH and its resistance against side-channel attacks. In this work, some cutting-edge results from recent contributions are reviewed in a unified treatment, focusing on the mathematical ideas lying behind them rather than on cryptographic and low-level implementation techniques. In particular, we first describe ways to speed up the class-group-action evaluation, which range from the use of different models of elliptic curves to working with different ideal class groups. We then survey some constant-time variants of CSIDH, that make the time and memory consumption during the computation of a public/shared key independent of the secret key. Finally, we examine the computation of the ideal class action when the structure of the ideal class group is known, which is the case for a specific set of CSIDH parameters.

*Keywords*: Post-quantum Cryptography; CSIDH; Elliptic curve; Isogeny

Mathematics Subject Classification 2020: 94A60

*Corresponding author.

## 1. Introduction

The security of public-key primitives relies on the assumption that some mathematical problems cannot be solved in polynomial time. Namely, many of the current public-key cryptosystems base their security either on the hardness of factoring integers or computing discrete logarithms over finite cyclic groups. Both these problems were considered beyond the computational power of any calculator, until Shor [57], in 1994, proposed Las Vegas polynomial-time algorithms for factoring integers and computing discrete logarithms on quantum computers.

Within the near future, quantum computers will probably be powerful enough to run Shor's algorithm and, consequently, undermine the security of many currently-deployed public-key schemes. In order to tackle this threat, a new line of cryptographic research, named *post-quantum cryptography*, has started looking for mathematical problems, and therefore cryptosystems, that can resist even the attacks of quantum computers.

The (supposedly) quantum-resistant cryptosystems that have been proposed so far can be merged into five large families: lattice-based, code-based, hash-based, multivariate and isogeny-based schemes. In this work, we will restrict our attention to the latter family, which is derived from classical number-theoretical results and, despite being the most recent, is particularly appealing as it enjoys short keys and ciphertexts.

The first isogeny-based cryptosystem was a non-interactive key-exchange protocol detailed by Couveignes in a talk given in 1997 [23]. This work had only spread privately until 2006, and the system was rediscovered independently by Stolbunov and Rostovtsev [56]. For a given imaginary quadratic order $\mathcal{O}$, the scheme exploits the free and transitive action of the ideal class group $\mathcal{C}\ell(\mathcal{O})$ on the $\mathbb{F}_{p^m}$-isomorphism classes of ordinary elliptic curves over $\mathbb{F}_{p^m}$ having endomorphism ring isomorphic to $\mathcal{O}$. Despite its theoretical interest and the introduction of remarkable mathematically driven speed-ups by De Feo *et al.* [27], this scheme turned out to be too inefficient to cope with a subexponential quantum attack devised by Childs *et al.* [20] exploiting the commutativity of $\mathcal{C}\ell(\mathcal{O})$.

To overcome the efficiency issues, in 2018, Castryck *et al.* [15] reshaped the seminal idea of Couveignes, Stolbunov and Rostovtsev using supersingular, rather than ordinary, elliptic curves. In fact, given a prime $p$ and a supersingular elliptic curve $E$ over $\mathbb{F}_p$, the ring $\mathsf{End}_{\mathbb{F}_p}(E)$, whose elements are the $\mathbb{F}_p$-endomorphisms of $E$, is an order $\mathcal{O}$ in a quadratic field (specifically, $\mathbb{Q}(\sqrt{-p})$). Furthermore, the ideal class group $\mathcal{C}\ell(\mathcal{O})$ acts freely and transitively on the set of ($\mathbb{F}_p$-isomorphism classes of) supersingular elliptic curves $E$ over $\mathbb{F}_p$ for which $\mathsf{End}_{\mathbb{F}_p}(E)$ is isomorphic to $\mathcal{O}$. The resulting protocol is named CSIDH, which stands for Commutative Supersingular Isogeny Diffie–Hellman and is pronounced "sea-side". CSIDH enjoys a faster key exchange than the original scheme, since the structure of rational groups of supersingular elliptic curves over $\mathbb{F}_p$ allows the efficient computation of the action of some class group elements.

To be more precise, computing the action of an arbitrary element $[\mathfrak{a}] \in \mathcal{C}\ell(\mathcal{O})$ has subexponential complexity [11, §9.5]. However, the set of rational points $E(\mathbb{F}_p)$ of a supersingular elliptic curve $E$ over $\mathbb{F}_p$ has cardinality equal to $p+1$ and, considering a prime $p = 4\ell_1\ell_2\cdots\ell_n - 1$ with $\ell_1, \ell_2, \ldots, \ell_n$ small odd primes, a special element $[\mathfrak{I}_i] \in \mathcal{C}\ell(\mathcal{O})$ can be associated to each prime $\ell_i$. The action of such special elements (respectively, their inverses) can be computed somewhat efficiently, since it is determined by an isogeny whose kernel is the unique subgroup of $E(\mathbb{F}_p)$ (respectively, $E^t(\mathbb{F}_p)$, where $E^t$ denotes the quadratic twist of $E$) having order $\ell_i$. As a consequence, the action of the elements in $\mathcal{C}\ell(\mathcal{O})$ having the form $\prod_{i=1}^n [\mathfrak{I}_i]^{e_i}$ — where the integral exponents $e_i$ are chosen from some small interval $[-B, B]$ — can be computed "ideal-by-ideal" working in the base field $\mathbb{F}_p$.

The CSIDH scheme deals only with the actions of such elements. In fact, the private key of a user is a vector $(e_1, \ldots, e_n) \in [-B, B]^n$, and a key exchange requires computing the action of $[\mathfrak{a}] = \prod_{i=1}^n [\mathfrak{I}_i]^{e_i}$. This computation consists in calculating a chain of "atomic" isogenies (one for each occurrence of the factors $[\mathfrak{I}_i]$), and it is the most expensive step of the protocol. Recently, several strategies have been proposed to speed up this computation [5, 12, 47, 51, 54] and to make its running time independent of the private key in order to prevent certain kinds of side-channel attacks [3, 7, 17, 19, 35, 46, 53].

### 1.1. *Our contribution*

Isogeny-based cryptography is growing fast. Keeping track of all its developments has become more difficult, especially for those who approach the subject for the first time. The need for a unified dissertation is particularly evident for CSIDH, since its promising features have already motivated many contributions regarding both its efficiency and its security. Our goal is to gather all those results which center on efficiency in a self-contained work, focusing on the mathematical ideas lying behind them rather than on cryptographic and low-level implementation techniques. This is motivated by the fact that CSIDH itself, as well as almost all cryptosystems deduced from it, is not as efficient as some other post-quantum counterparts. Our belief is that a major computational speed-up is likely to come either from some computational number-theoretical results or a high-level implementation optimization, rather than cryptographic and low-level implementation techniques. This work is then meant as an invitation to the broad number-theoretic community, with the hope of triggering further research on the topic.[a]

### 1.2. *Overview*

Three main themes will be considered throughout this survey: speed-up of the class-group-action evaluation, constant-time class-group-action evaluation and computations in a class group with known structure. Below we provide a short summary of

[a] This paper is the extended version of the long abstract [45].

the results we discuss for each theme. As its aim is that of providing a preview of the papers surveyed in this work, it refers to some mathematical objects which will be introduced later in the paper.

### 1.2.1. *Speed-up of the class-group-action evaluation*

In the original CSIDH proof-of-concept implementation, computing the action of some ideal class $[\mathfrak{I}_i]$ requires an $\mathbb{F}_p$-rational point $P$ lying on the domain curve and having order divisible by $\ell_i$. Usually, the order of $P$ is also divisible by some of the other odd primes $\ell_j \neq \ell_i$ dividing $p+1$. As a consequence, the image of $P$ via the atomic isogeny corresponding to $[\mathfrak{I}_i]$, say $\varphi$, can still be used to compute some of the subsequent atomic isogenies. Using $\varphi(P)$ is cheaper than determining a new $\mathbb{F}_p$-rational point, and therefore improving its computation is of great interest.

In [47], Meyer and Reith propose a trick to reduce the number of field operations for computing $\varphi(P)$. Furthermore, they highlight how the birational equivalence between Montgomery curves and Twisted Edwards curves can improve the calculation of the image curve of $\varphi$.

Costello and Hisil [21] build on [47] to derive more efficient formulas for evaluating isogenies between Montgomery curves. Following the same research direction, two independent works of Bernstein *et al.* [5] and Kodera *et al.* [39] construct new efficient formulas for computing both $\varphi(P)$ and the image curve.

In [54], Onuki and Takagi show that $\prod_{i=1}^{n}[\mathfrak{I}_i]^3 = [(1)]$ (the same was observed independently in [16] by Castryck *et al.*) and derive an efficient procedure to compute the action of $\prod_{i=1}^{n}[\mathfrak{I}_i]$. This can be exploited to speed up the computation of the actions of ideals $\prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i}$ corresponding to vectors $(e_1, \ldots, e_n)$ of Hamming weight $n$.

Nakagawa *et al.* [51] propose using an $L_1$-norm ball as a secret-key space, thus achieving a secure variant of CSIDH whose average execution cost is minimal among a specific family of secret-key spaces.

Castryck and Decru [12] use a different approach to speed up the class-group-action evaluation. In particular, they transpose the CSIDH scheme to a set of supersingular elliptic curves different from that used in CSIDH. The resulting scheme for a bespoke prime $p$ is 5.68% faster than the CSIDH protocol for a prime of similar size.

Finally, Castryck *et al.* [14] describe a deterministic procedure to obtain a chain of $\ell$-isogenies starting from an elliptic curve with a given $\ell$-torsion point. In order to fully take advantage of the procedure, they use a different secret-key space than the original CSIDH parameters.

### 1.2.2. *Constant-time class-group-action evaluation*

In the CSIDH proof-of-concept implementation proposed in [15], the running time to compute the action of $\prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i}$ heavily depends on the private key $(e_1, \ldots, e_n)$.

Consequently, an attacker having access to timing data could retrieve information about the private key.

To tackle this issue, Bernstein *et al.* [7] present a constant-time implementation that, for each private key, executes the same number of field operations, with a negligible failure probability. Building on some of the techniques introduced in [7], Meyer *et al.* [46] devise a no-failure implementation whose number of field operations is independent of the private key but may vary due to the randomness used. In particular, in their implementation, a fixed number of isogenies is computed for each of the small odd primes $\ell_1, \dots, \ell_n$, and a key space lying in $(\mathbb{Z}_{\geq 0})^n$ is considered.

The implementation proposed by Onuki *et al.* in [53] retains some of the techniques from [46] while allowing secret keys to have negative entries. In particular, for the computation of each atomic isogeny, two points — one on the domain curve and one on its twist — are used.

Hutchinson *et al.* [35] and Chi-Domínguez and Rodríguez-Henríquez [19] independently extend to the CSIDH setting the optimal strategies introduced in [36], gaining a further speed-up for the constant-time evaluation.

Finally, in [3], Banegas *et al.* design CTIDH, which is a new constant-time evaluation of the CSIDH group action based on a suitably tailored key space.

### 1.2.3. *Computations in an ideal class group with known structure*

The similarities of CSIDH with the standard Diffie–Hellman protocol have led researchers to use it as a building block for new isogeny-based cryptosystems and, in particular, digital signatures.

Stolbunov [59] was the first to propose an isogeny-based digital signature scheme (working with ordinary elliptic curves), but the security of his protocol requires the ideal class group $\mathcal{C}\ell(\mathcal{O})$ to have a suitable structure. The same assumption on $\mathcal{C}\ell(\mathcal{O})$ is also needed when translating the scheme to the CSIDH setting. Unfortunately, for primes $p$ of cryptographic size, computing the structure of $\mathcal{C}\ell\ell(\mathcal{O})$ is prohibitive. Therefore, alternative solutions to make the signature scheme safe have been proposed [26], but none of them seem efficient enough to be used in practice.

Finally, in 2019, Beullens *et al.* [8] made a record class group computation and explicitly found the structure of $\mathcal{C}\ell(\mathcal{O})$ for one set of CSIDH parameters, named CSIDH-512. Starting from this result, they were able to design the first practical isogeny-based digital signature, named CSI-FiSh.

For CSIDH-512, $\mathcal{C}\ell(\mathcal{O})$ is a cyclic group for which a generator $\mathfrak{g}$ is known. Therefore, it is isomorphic to $\mathbb{Z}/N\mathbb{Z}$, where $N = |\mathcal{C}\ell(\mathcal{O})|$. Thus, each vector $(e_1, \dots, e_n)$ can be represented by an integer in $\{0, \dots, N-1\}$. Such unique representation guarantees that no information is leaked when producing a CSI-FiSh signature. However, the verification algorithm requires computing the action of an element $[\mathfrak{g}]^a$, with $a \in \mathbb{Z}/N\mathbb{Z}$. In order to make this computation feasible, it is necessary to find a vector $(f_1, \dots, f_n)$ with *small* integral coordinates such that $[\mathfrak{g}]^a = \prod_{i=1}^n [\mathfrak{I}_i]^{f_i}$. In [8], a lattice-based strategy to find a vector of this kind is proposed.

### 1.3. *Roadmap*

The remainder of the paper is organized as follows. In Section 2, some preliminary results on elliptic curves and isogenies are reviewed. Section 3 describes the CSIDH key-exchange protocol and details its original proof-of-concept implementation. In Section 4, we discuss the techniques that have been proposed so far to speed up the execution of CSIDH. In Section 5, some constant-time CSIDH implementations are described. Section 6 focuses on the CSIDH-512 set of parameters. Finally, Section 7 draws some conclusions.

## 2. Preliminaries

In this section, we recall some basic notions and results about elliptic curves over finite fields. In doing this, at the beginning we will denote by $K$ a perfect field such that $\mathrm{char}(K) \notin \{2, 3\}$, and then we will specialize to the case $K = \mathbb{F}_p$, where $\mathbb{F}_p$ is the finite field with $p$ elements and $p > 3$ is a prime. For more details about elliptic curves over finite fields, we refer the interested reader to [24, 30, 48, 58]. Later on, we focus on the class group action which the CSIDH key exchange is based on. As evaluating this action is the heaviest computational task within an execution of the cryptosystem, we recall a few models of elliptic curves that have been used to speed it up (see Section 3).

### 2.1. *Elliptic curves*

An *elliptic curve* over $K$ is a projective algebraic curve whose affine equation, up to $\overline{K}$-birational equivalence, is a nonsingular affine Weierstrass equation of the form

$$E : \underbrace{y^2 + a_1xy + a_3y - (x^3 + a_2x^2 + a_4x + a_6)}_{F(x,y)} = 0 \tag{1}$$

with $a_1, a_3, a_2, a_4, a_6 \in K$. By *nonsingular* we mean that

$$\left(\frac{\partial F}{\partial x}(P), \frac{\partial F}{\partial y}(P)\right) \neq (0, 0)$$

for all pairs $P \in \overline{K} \times \overline{K}$ satisfying Eq. (1), where $\overline{K}$ denotes the algebraic closure of $K$. Given a second elliptic curve $E'$ over $K$,

$$E' : y^2 + a_1'xy + a_3'y = x^3 + a_2'x^2 + a_4'x + a_6',$$

we say that $E$ and $E'$ are *isomorphic over* $K$ (or $K$-isomorphic) if $E'$ can be obtained from $E$ by a change of variables of the form

$$\begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} u^2 & 0 \\ u^2s & u^3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} r \\ t \end{pmatrix} \quad \text{with } u \in K^*, r, s, t \in K$$

and dividing the resulting equation by $u^6$. Every elliptic curve $E$ is given by a *short Weierstrass equation*

$$E : y^2 = x^3 + Ax + B \tag{2}$$

up to $\overline{K}$-birational equivalence.

A binary operation, for which the additive notation is usually adopted, can be defined over the set containing the affine points of an elliptic curve $E$ and a formal point $\infty$, making it an abelian group. Given two elliptic curves

$$E\colon y^2 = x^3 + Ax + B \quad \text{and} \quad E'\colon y^2 = x^3 + A'x + B'$$

over $K$, an *isogeny* $\varphi : E \to E'$ is a non-constant rational map which is furthermore, a group homomorphism.[b] We say that $\varphi$ is *separable* if $\overline{K}(E)$ is a separable field extension of $\varphi^*(\overline{K}(E'))$, where

- $\overline{K}(E)$ is the fraction field of $\overline{K}[x, y]/(y^2 - x^3 - Ax - B)$ and $\overline{K}(E')$ is defined analogously.
- $\varphi^*$ is the pullback map sending $f \in \overline{K}(E')$ into $\varphi^*(f) = f \circ \varphi \in \overline{K}(E)$.

Furthermore, the *degree* of $\varphi$ is defined as the dimension of $\overline{K}(E)$ over $\varphi^*(\overline{K}(E'))$, and we denote it by $\deg(\varphi)$. For a separable isogeny $\varphi$, the relation $\deg(\varphi) = |\mathrm{Ker}(\varphi)|$ holds [64, Theorem 12.8].

### 2.2. *Supersingular elliptic curves*

An *endomorphism* of an elliptic curve $E$ is an isogeny from $E$ to itself. We denote by $\mathsf{End}(E)$ the set containing all the endomorphisms of $E$ together with the zero map. This set is a ring with respect to pointwise addition and composition. An elliptic curve $E$ is called *supersingular* if $\mathsf{End}(E)$ is non-commutative, otherwise it is called *ordinary*. In particular, when $E$ is a supersingular elliptic curve, $\mathsf{End}(E)$ is an order in a quaternion algebra [58, Theorem V.3.1].

Although some of the results discussed below hold for general fields, for the sake of simplicity we will assume $K = \mathbb{F}_p$, where $p > 3$ is a prime, until the end of the section.

An elliptic curve $E : y^2 = x^3 + Ax + B$ over $\mathbb{F}_p$ is supersingular if and only if the subgroup of its rational points, i.e.

$$E(\mathbb{F}_p) = \{(x_0, y_0) \in \mathbb{F}_p \times \mathbb{F}_p \,|\, y_0^2 = x_0^3 + Ax_0 + B\} \cup \{\infty\},$$

has cardinality $p+1$ [65, Theorem 4.1]. Furthermore, given a supersingular elliptic curve $E$ over $\mathbb{F}_p$, an elliptic curve $E'$ over $\mathbb{F}_p$ is isogenous to $E$ (i.e. there exists an isogeny $E' \to E$) if and only if $E'$ is supersingular too [62, §3].

By definition, each coordinate of an isogeny $\varphi : E \to E'$ is the fraction of two polynomials in $\overline{\mathbb{F}}_p[x, y]$; if the coefficients of such polynomials lie in $\mathbb{F}_p$, then $\varphi$ is said to be *defined over* $\mathbb{F}_p$. We denote by $\mathsf{End}_{\mathbb{F}_p}(E)$ the subring of $\mathsf{End}(E)$ containing the zero map and all endomorphisms of $E$ which are defined over $\mathbb{F}_p$.

[b] More generally, if $\mathcal{E}$ and $\mathcal{E}'$ are curves, and $\alpha\colon E \to \mathcal{E}$ and $\beta\colon \mathcal{E}' \to E'$ are $\overline{K}$-birational equivalences, we say that $\Phi\colon \mathcal{E} \to \mathcal{E}'$ is an isogeny (between curves) if $\beta \circ \Phi \circ \alpha$ is an isogeny.

### 2.3. *Class-group action*

From now on, we assume that $E$ is a supersingular elliptic curve over $\mathbb{F}_p$, with $p > 3$.

The ring $\mathsf{End}_{\mathbb{F}_p}(E)$ is isomorphic to an order in the imaginary quadratic field $\mathbb{Q}(\sqrt{-p})$ [28, Theorem 2.1]. In particular, we have either $\mathsf{End}_{\mathbb{F}_p}(E) \simeq \mathbb{Z}[\sqrt{-p}]$ or $\mathsf{End}_{\mathbb{F}_p}(E) \simeq \mathcal{O}_{\mathbb{Q}(\sqrt{-p})}$. In the former case we say that $E$ lies *on the floor*, otherwise we say that $E$ lies *on the surface*.[c] We note that floor and surface coincide when $p$ equals 1 modulo 4. CSIDH deals only with supersingular elliptic curves on the floor, and works with primes $p$ of the form $p = 4\ell_1 \cdot \ldots \cdot \ell_n - 1$, where $\ell_1, \ldots, \ell_n$ are distinct odd primes. In this case, the group $E(\mathbb{F}_p)$ has two possible structures, namely:

$$\mathbb{Z}_4 \times \mathbb{Z}_{\ell_1} \times \cdots \times \mathbb{Z}_{\ell_n} \quad \text{or} \quad \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_{\ell_1} \times \cdots \times \mathbb{Z}_{\ell_n}.$$

Theorem 2.7 from [28] implies that $E$ lies on the floor if and only if the former relation holds. Therefore, the number of 2-torsion points in $E(\mathbb{F}_p)$ determines whether $E$ lies on the floor or on the surface.

We denote by $\mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$ the set of all supersingular elliptic curves $E$ over $\mathbb{F}_p$ (up to $\mathbb{F}_p$-isomorphism) such that there exists an isomorphism between $\mathcal{O}$ and $\mathsf{End}_{\mathbb{F}_p}(E)$ mapping $\sqrt{-p}$ to the Frobenius endomorphism $\pi : (x, y) \mapsto (x^p, y^p)$. Under this isomorphism, in particular, an ideal $\mathfrak{a} \subset \mathcal{O}$ can be viewed as a set of endomorphisms.

The ideal class group $\mathcal{C}\ell(\mathcal{O})$ acts freely and transitively on $\mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$ [15, Theorem 7] as follows: given $[\mathfrak{a}] \in \mathcal{C}\ell(\mathcal{O})$, we define $[\mathfrak{a}] \star [E]$ — or $E/\mathfrak{a}$ or $\mathfrak{a} \star E$ for simplicity — to be the codomain of the unique (up to $\mathbb{F}_p$-isomorphism) isogeny $\varphi_{\mathfrak{a}} \colon E \to E/\mathfrak{a}$ with kernel $\cap_{\alpha \in \mathfrak{a}} \operatorname{Ker}(\alpha)$. One can check that this definition does not depend on the representative chosen for $[\mathfrak{a}]$. On the other hand, we remark that $\varphi_{\mathfrak{a}}$ *does* depend on such choice, and its degree equals the norm of $\mathfrak{a}$.

The general approach to compute the action of a random element $[\mathfrak{a}] \in \mathcal{C}\ell(\mathcal{O})$ is the following:

- Find an ideal $\mathfrak{b} \subset \mathcal{O}$ that can be written as a product of ideals $\prod_i \mathfrak{b}_i^{e_i}$ such that the norm of each $\mathfrak{b}_i$ is at most $M$ and $[\mathfrak{b}] = [\mathfrak{a}]$ (such $\mathfrak{b}$ always exists for $M = O(\log p)$ by [11, §9.5]).
- Compute the chain of isogenies

$$E_0 \xrightarrow{\varphi_{\mathfrak{b}_1}} \cdots \xrightarrow{\varphi_{\mathfrak{b}_n}} E_n$$

using Vélu's formulas (which will be presented later in this section).

With no further assumption on $\mathcal{O}$, however, finding a suitable $\mathfrak{b}$ takes subexponential time. The state-of-the-art algorithm, presented in [9], also ensures that the exponents $e_i$ are suitably small. Another possible problem is that the kernels of

[c] This terminology, first introduced by Kohel [40], is borrowed from ordinary isogeny graphs, which have a volcano structure (see e.g. [60]).

$\mathfrak{b}_1, \dots, \mathfrak{b}_n$, needed to apply Vélu's formulas, might lie in large extensions of $\mathbb{F}_p$. We will now see how CSIDH circumvents these issues by means of an appropriate choice of parameters.

Namely, the CSIDH scheme sets $\mathcal{O} = \mathbb{Z}[\sqrt{-p}]$ for $p = 4\ell_1 \cdot \ldots \cdot \ell_n - 1$, and exploits the action $\star$ of $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ on $\mathcal{E}\ell\ell_p(\mathbb{Z}[\sqrt{-p}], \pi)$ to construct a key exchange. By construction, each small odd prime $\ell_i$ dividing $p+1$ *splits* in $\mathcal{O}$, i.e. $\ell_i\mathcal{O}$ can be written as a product of two distinct prime ideals $\mathfrak{I}_i$ and $\bar{\mathfrak{I}}_i$. Precisely (see e.g. [48, Theorem 3.41]), $\mathfrak{I}_i = (\ell_i, \sqrt{-p} - 1)$ and $\bar{\mathfrak{I}}_i = (\ell_i, \sqrt{-p} + 1)$. We stress that $[\bar{\mathfrak{I}}_i] = [\mathfrak{I}_i]^{-1}$ since $\mathfrak{I}_i\bar{\mathfrak{I}}_i = \ell_i\mathcal{O}$, which is principal. The action of these ideals can be efficiently evaluated. In fact, for all $i = 1, \dots, n$ and $[E] \in \mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$, the curves $E/\mathfrak{I}_i$ and $E/\bar{\mathfrak{I}}_i$ are the image curves of the separable isogenies with domain $E$ and kernels

$$E(\mathbb{F}_p) \cap E[\ell_i] \quad \text{and} \quad \{P \in E(\mathbb{F}_{p^2}) \,|\, \pi(P) = -P\} \cap E[\ell_i],$$

respectively. The images $E/\mathfrak{I}_i$ and $E/\bar{\mathfrak{I}}_i$ are uniquely defined, modulo $\mathbb{F}_p$-isomorphism, by their kernel [58, Theorem III.4.1]. Furthermore, they can be explicitly computed from a point $P$ lying on $E(\mathbb{F}_p)$ or $E(\mathbb{F}_{p^2})$ and having order $\ell_i$. This is usually done by means of Vélu's formulas, which we gather in Theorem 1 for elliptic curves in short Weierstrass form. Here we specialize the formulas to the CSIDH setting; a more general version of the formulas below can be found in [32, §25.1.1]. Vélu's formulas can be also adapted to other models of elliptic curves (see Theorem 3 and [50]).

**Theorem 1 (Vélu's formulas).** *Let $E\colon y^2 = x^3 + Ax + B$ be an elliptic curve over $K$, and $P \in E(\overline{K})$ a point of odd order $\ell$. Choose a set $\mathcal{S} \subset \langle P \rangle$ such that*

- $\ell = 1 + 2 \cdot |\mathcal{S}|$,
- $\langle P \rangle = \{\infty\} \cup \mathcal{S} \cup \{-Q \mid Q \in \mathcal{S}\}$,

*e.g. $\mathcal{S} = \{P, [2]P, \dots, [\frac{\ell-1}{2}]P\}$. For each point $Q = (x_Q, y_Q) \in \mathcal{S}$ define*

$$A' = A - 10 \cdot \sum_{Q \in \mathcal{S}} (3x_Q^3 + A) \quad \text{and} \quad B' = B - 14 \cdot \sum_{Q \in \mathcal{S}} (2y_Q^2 + x_Q(3x_Q^3 + A))$$

*as parameters of an elliptic curve $E'$ in short Weierstrass form. Then*

$$\begin{aligned} \varphi\colon E &\to E' \\ (x, y) &\mapsto (x', y') \end{aligned}$$

*with*

$$x' = x + \sum_{Q \in \mathcal{S}} \left( 2 \cdot \frac{(3x_Q^3 + A)}{x - x_Q} + \left( \frac{2y_Q}{x - x_Q} \right)^2 \right),$$

$$y' = y - 2 \cdot \sum_{Q \in \mathcal{S}} \left( 4y_Q^2 \frac{y}{(x - x_Q)^3} + (3x_Q^3 + A) \frac{y}{(x - x_Q)^2} \right)$$

*is a separable isogeny with kernel $\langle P \rangle$.*

The computational cost of finding the image of a given point in the domain and the parameters of the image curve, using the above formulas, is $\Theta(\ell)$. Optimizing Vélu's formulas is an effective strategy to speed up the CSIDH class-group-action evaluation, as we are going to show in Section 4.

## 2.4. *Other models of elliptic curves*

The short Weierstrass equation (2) is not the only possible way to represent an elliptic curve up to $K$-isomorphism: alternative models are often used in order to speed up computations. Throughout this section, we will consider three of them: Montgomery curves, Tate curves and Edwards curves.

### 2.4.1. *Montgomery*

A *Montgomery curve* over $K$ is an algebraic curve defined by an affine equation of the form

$$E_{A,B}\colon By^2 = x^3 + Ax^2 + x \tag{3}$$

with $B \neq 0$ and $A^2 \neq 4$. Each Montgomery curve over $K$ is $K$-birationally equivalent to an elliptic curve over $K$ in short Weierstrass form with the following parameters [22, §2.4]:

$$\begin{cases} A' = B^2\left(1 - \dfrac{A^2}{3}\right), \\ B' = \dfrac{B^3 A}{3}\left(\dfrac{2A^2}{9} - 1\right). \end{cases}$$

The next result from [52] deals with the converse, specialized to the case $K = \mathbb{F}_q$ where $q$ is a prime power.

**Proposition 1.** *Let $E'\colon y^2 = x^3 + A'x + B'$ be an elliptic curve in short Weierstrass form over $\mathbb{F}_q$. Then $E'$ is $\mathbb{F}_q$-birationally equivalent to a Montgomery curve $E$ over $\mathbb{F}_q$ if and only if $E'$ has an $\mathbb{F}_q$-rational 2-torsion point $(\alpha, 0)$ and $3\alpha^2 + A' = s^2$ for some $s \in \mathbb{F}_q$.*

*If that case, the parameters of $E$ are*

$$\begin{cases} A = 3\alpha s, \\ B = s. \end{cases}$$

The main reason why Montgomery curves are often preferred over elliptic curves in Weierstrass form is that they allow for faster scalar multiplications. In fact, given a point $P$ of a Montgomery curve, say $P = (x_P, y_P)$, and a positive integer $m$, the Montgomery ladder [22, §3] provides efficient formulas to compute the x-coordinate of $[m]P$ given $x_P$ (and $E$).

Furthermore, the following theorem [15, Proposition 8] shows how, in the CSIDH setting, Montgomery curves allow representing each element of $\mathcal{E}\ell\ell_p(\mathbb{Z}[\sqrt{-p}], \pi)$ with a single element of the base field $\mathbb{F}_p$.

**Theorem 2.** *Let $p > 3$ be a prime such that $p \equiv 3 \pmod 8$ and $E$ a supersingular elliptic curve over $\mathbb{F}_p$. Then, $E$ lies on the floor if and only if there exists $A \in \mathbb{F}_p$ such that $E$ is $\mathbb{F}_p$-isomorphic to the Montgomery curve $E_{A,1} : y^2 = x^3 + Ax^2 + x$. Moreover, if such $A$ exists, it is unique.*

Since $[E] \in \mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$ is the class of supersingular elliptic curves that are $\mathbb{F}_p$-isomorphic to $E$, all the curves therein are $\mathbb{F}_p$-isomorphic to the same Montgomery curve $E_{A,1}$. Therefore, $[E]$ can be identified with $A$.

The following result (see [21, Theorem 1; 44, Theorem 1]) adapts Vélu's formulas to Montgomery curves.

**Theorem 3.** *Consider a point $P \in E_{A,1}(\overline{\mathbb{F}_p})\backslash\{\infty\}$ of odd order $\ell = 2d + 1$ on a Montgomery curve $E_{A,1} : y^2 = x^3 + Ax^2 + x$ defined over $\mathbb{F}_p$. Define*

$$\sigma = \sum_{i=1}^{d} x_{[i]P}, \quad \tilde{\sigma} = \sum_{i=1}^{d} (1/x_{[i]P}) \quad \text{and} \quad \omega = \prod_{i=1}^{d} x_{[i]P},$$

*where $x_{[i]P}$ denotes the $x$-coordinate of the point $[i]P$. Then, the Montgomery curve*

$$E_{A',1} : y^2 = x^3 + A'x^2 + x$$

*with*

$$A' = (6\tilde{\sigma} - 6\sigma + a) \cdot \omega^2$$

*is the codomain of the separable isogeny $\phi : E_{A,1} \to E_{A',1}$ of degree $\ell$ and kernel $\langle P \rangle$, whose coordinates are given by the map*

$$\phi : (x, y) \mapsto (f(x), \omega y f'(x)),$$

*where*

$$f(x) = x \cdot \prod_{i=1}^{d} \left( \frac{x \cdot x_{[i]P} - 1}{x - x_{[i]P}} \right)^2 \tag{4}$$

*and $f'(x)$ is its derivative.*

### 2.4.2. *Tate*

A *Tate curve* over $K$ is an algebraic curve defined by an affine equation

$$E\colon y^2 + (1 - C)xy - By = x^3 - Bx^2$$

with

$$(1 - C)^4 B^3 - (1 - C)^3 B^3 - 8(1 - C)^2 B^4 + 36(1 - C)B^4 - 27B^4 + 16B^5 \neq 0.$$

The main advantage of using Tate curves is that the parameters $B$ and $C$ can be chosen in such a way that $[m]P = \infty$ for $P = (0, 0)$ and some small integer $m \geq 4$. Namely, the latter condition is equivalent to imposing a polynomial condition $f_m(B, C) = 0$ [34, §4.4]. For example, the condition $[5]P = \infty$ is equivalent to $B - C = 0$.

In the CSIDH setting, Tate curves — in particular the fact that $P = (0, 0)$ is a torsion point of suitable order — are used to speed up isogeny computations, as we are going to show in Section 4.6.

#### 2.4.3. *Twisted Edwards*

A *twisted Edwards curve* over $K$ is an algebraic curve defined by an affine equation

$$E\colon ax^2 + y^2 = 1 + dx^2y^2 \tag{5}$$

with $a, d$ distinct nonzero elements. When $a = 1$, we simply call $E$ an *Edwards curve* and denote it by $E_d$. Each twisted Edwards curve is $K$-birationally equivalent to a Montgomery curve [4, Theorem 3.2] with parameters

$$\begin{cases} A = 2 \cdot \dfrac{a+d}{a-d}, \\ B = \dfrac{4}{a-d}. \end{cases}$$

The above formulas can be easily inverted to compute the twisted Edwards curve corresponding to a given Montgomery curve:

$$\begin{cases} a = \dfrac{A+2}{B}, \\ d = \dfrac{A-2}{B}. \end{cases}$$

The main advantage of twisted Edwards curves consists in having efficient formulas to compute the image curve corresponding to a given isogeny [50, Corollary 1]. We gather them below.

**Theorem 4.** *Consider a point $P \in E(\overline{\mathbb{F}_p})\backslash\{\infty\}$ of odd order $\ell = 2s+1$ on a twisted Edwards curve $E : ax^2 + y^2 = 1 + dx^2y^2$ defined over $\mathbb{F}_p$. Define*

$$B = \prod_{i=1}^{s} y_{[i]P} \quad \textit{and} \quad \hat{d} = B^8 \left(\frac{d}{a}\right)^{\ell}, \tag{6}$$

*where $y_{[i]P}$ denotes the $y$-coordinate of the point $[i]P$. Then, there exists a separable isogeny $\varphi\colon E \to E'$ with kernel $\langle P \rangle$, where $E'\colon x^2 + y^2 = 1 + \hat{d}x^2y^2$.*

Twisted Edwards curves will be considered only to compute the codomain of isogenies of given kernels (see Section 4.1), rather than evaluating these isogenies on given points.

## 2.5. *Finite field arithmetic*

Additions, multiplications and — more rarely — inversions over $\mathbb{F}_{p^n}$ are essentially the only operations involved in CSIDH. As a reference for the rest of this paper, we summarize their computational costs in the following table, setting $q = p^n$. We refer the interested reader to [61] for further details.

| | Computation time | Reference |
|---|---|---|
| Addition | $O(\log q)$ | [38, p. 171] |
| Multiplication | $O(\log q \log\log q)$ | [33] |
| Inversion | $O(\log q(\log\log q)^2)$ | [63, Theorem 11.10] |

## 3. CSIDH

In this section, we describe the original CSIDH key-exchange protocol [15] and detail its proof-of-concept implementation.

### 3.1. *The key-exchange protocol*

As anticipated, the isogeny-based key exchange CSIDH [15] is obtained from the action $\star$ of $\mathcal{C}\ell(\mathcal{O})$ on $\mathcal{E}\ell\ell_p(\mathcal{O},\pi)$, with $\mathcal{O} = \mathbb{Z}[\sqrt{-p}]$, for a fixed CSIDH prime $p = 4\ell_1 \cdot \ldots \cdot \ell_n - 1$. Since — as we have already remarked — evaluating the action of a random element $[\mathfrak{a}] \in \mathcal{C}\ell(\mathcal{O})$ has subexponential complexity, the scheme deals only with the action of the elements of the form $\prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i}$, where the ideals $\mathfrak{I}_i$ are defined as in Section 2.3 and the integral exponents $e_i$ are chosen from some small interval $[-B, B]$, with $B$ a positive integer. In fact, evaluating these actions corresponds to calculating sequences of efficient actions, i.e. sequences of small-degree isogenies, as we have seen in Section 2.3. Thus, a first rough estimate suggests that computing the action of an element $\prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i}$ requires $O(Bn\ell_n)$ time.

In addition to the prime $p$ (which, in turn, determines $n$), the public parameters of the CSIDH protocol specify the value of the small positive integer $B$ and a starting elliptic curve $E_0 : y^2 = x^3 + x$, which is defined over $\mathbb{F}_p$, supersingular [64, Theorem 4.37], and lies on the floor [15, Proposition 8].

The set of private keys is $[-B, B]^n$: each vector $(e_1, \ldots, e_n)$ represents an element $[\mathfrak{a}] = \prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i}$ in $\mathcal{C}\ell(\mathcal{O})$. The public key corresponding to the private key $(e_1, \ldots, e_n)$ is defined as $E_0/\mathfrak{a}$. Based on heuristic observations [15, p.4], the set of public keys is assumed to be $\mathcal{E}\ell\ell_p(\mathcal{O},\pi)$ — equivalently, the equality $\{\prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i} \mid (e_1, \ldots, e_n) \in [-B, B]^n\} = \mathcal{C}\ell(\mathcal{O})$ is assumed to hold.

Below we recall how the key exchange between two users, Alice and Bob, works. At the beginning, Alice and Bob have key pairs $((e_1, \ldots, e_n), E_0/\mathfrak{a})$ and $((f_1, \ldots, f_n), E_0/\mathfrak{b})$, respectively. Here $\mathfrak{a} = \prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i}$ and $\mathfrak{b} = \prod_{i=1}^{n}[\mathfrak{I}_i]^{f_i}$. Then the interaction proceeds as follows:

- Alice and Bob exchange their public keys.
- Alice computes $[\mathfrak{a}] \star [E_0/\mathfrak{b}] = [\mathfrak{a}][\mathfrak{b}] \star [E_0]$.
- Bob computes $[\mathfrak{b}] \star [E_0/\mathfrak{a}] = [\mathfrak{b}][\mathfrak{a}] \star [E_0]$.

The commutativity of $\mathcal{C}\ell(\mathcal{O})$ guarantees that both users obtain the same element of $\mathcal{E}\ell\ell_p(\mathcal{O},\pi)$, which, thanks to the assumption $\mathcal{O} = \mathbb{Z}[\sqrt{-p}]$ and Theorem 2, can be represented by a single element in $\mathbb{F}_p$. Furthermore, given a Montgomery curve

$E_{A,1}$, Theorem 3 explains how to compute the action of $[\mathfrak{I}_i]$ on $[E_{A,1}] \in \mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$. In particular, a point $Q \in E_{A,1}(\mathbb{F}_p)$ having order divisible by $\ell_i$ is used to compute the isogeny $\varphi$ with domain $E_{A,1}$ and a group $G \subseteq E_{A,1}(\mathbb{F}_p)$ of order $\ell_i$ as kernel. Namely, $G = \langle P \rangle$, where $P = [\mathsf{ord}(Q)/\ell_i]Q$.[d] As we have seen in Section 2.3, $E_{A,1}/\mathfrak{I}_i$ is exactly the image of such an isogeny. We remark that the point $\varphi(Q)$ has order $\mathsf{ord}(Q)/\ell_i$ and therefore can potentially be used to compute the action of another element $[\mathfrak{I}_j]$, with $j \neq i$. Both $E_{A,1}/\mathfrak{I}_i$ and $\varphi(Q)$ can be calculated using the formulas in Theorem 3.

In order to compute the action of an element $[\mathfrak{I}_i]^{-1}$, let us consider a Montgomery curve $E_{A,1}$ representing an element in $\mathcal{E}\ell\ell_p(\mathcal{O}, \pi)$. Since the action $\star$ is free and transitive, there exists a unique $[\mathfrak{c}]$ in $\mathcal{C}\ell(\mathcal{O})$ such that $[E_{A,1}] = [\mathfrak{c}] \star [E_0]$, where $E_0$ is the elliptic curve of equation $y^2 = x^3 + x$. It is possible to show [15, Remark 5] that $E_{-A,1}$ is the Montgomery curve representing $[\mathfrak{c}]^{-1} \star [E_0]$. As a consequence, the result of the action of $[\mathfrak{I}_i]^{-1}$ on $[E_{A,1}]$ is $[E_{-A',1}]$, where $[E_{A',1}] = [\mathfrak{I}_i] \star [E_{-A,1}] = ([\mathfrak{I}_i][\mathfrak{c}]^{-1}) \star [E_0]$. In fact, we have

$$[\mathfrak{I}_i]^{-1} \star [E_{A,1}] = ([\mathfrak{I}_i]^{-1}[\mathfrak{c}]) \star [E_0] = ([\mathfrak{I}_i][\mathfrak{c}]^{-1})^{-1} \star [E_0].$$

If a random $x \in \mathbb{F}_p$ is sampled, then either $x$ is the x-coordinate of a point $Q$ in $E_{A,1}(\mathbb{F}_p)$, or $-x$ is the x-coordinate of $Q \in E_{-A,1}(\mathbb{F}_p)$, depending on whether $x^3 + Ax^2 + x$ is a quadratic residue modulo $p$ or not.[e] In the former case, $Q$ can be used to obtain a point $P$ in $E_{A,1}(\mathbb{F}_p)$ having order $\ell_i$, for some $i \in \{1, \dots, n\}$. Hence, with such point $P$, the action of $[\mathfrak{I}_i]$ on $[E_{A,1}]$ can be computed. In the latter case, $Q$ can be used to obtain a point in $E_{-A,1}(\mathbb{F}_p)$ having order $\ell_i$, for some $i \in \{1, \dots, n\}$, and compute the action of $[\mathfrak{I}_i]$ on $[E_{-A,1}]$, from which the action of $[\mathfrak{I}_i]^{-1}$ on $[E_{A,1}]$ is then easily deduced as explained above.

The complete procedure to compute the action of a vector $(e_1, \dots, e_n)$ on $[E_0]$ or a public key $[E_{A,1}]$ is detailed in Algorithm 4, which was first depicted in the original CSIDH paper. Algorithms 1–3 are the subroutines used to sample a rational point on a given Montgomery curve ($\mathsf{RationalPoint}$), compute the codomain of an isogeny with given kernel ($\mathsf{IsoImage}$) and evaluate an isogeny with given kernel at some point ($\mathsf{IsoEval}$), respectively. In the following pages, we will often refer to these subroutines, since nearly all the CSIDH optimizations and variants build on them.

When $A$ and $K$ are given, we remark that Algorithms 2 and 3 use the same x-coordinates of suitable elements in $\langle K \rangle$, as shown in Theorem 3. Therefore, such coordinates can be computed just once and passed on to both the algorithms. In other words, they are supposed to share states. We also stress that their running time, as one can deduce from Theorem 3, is $\Theta(\ell)$ with $\ell = |\langle K \rangle|$. Finally, Algorithm 4 is expected to end as we observe that, for a uniformly random rational point $P \in E_{A,1}(\mathbb{F}_p)$, the probability that its order is not divisible by $\ell_i$ is $1/\ell_i$.

[d] We highlight that, starting from a different point $Q' \in E_{A,1}(\mathbb{F}_p)$, whose order is divisible by $\ell_i$, yields the same group $G$. Therefore, the resulting isogeny $\varphi$ does not depend on the choice of $Q$.

[e] This follows from the fact that $-1$ is not a quadratic residue modulo $p$, since the considered prime $p$ is congruent to 3 modulo 4.

**Algorithm 1.** RationalPoint.

```
Input: $A \in \mathbb{F}_p$.
Output: $(x, s)$, where $x$ is the x-coordinate of a point in $E_{s\cdot A,1}(\mathbb{F}_p)\backslash\{\infty\}$ and
    $s \in \{-1, 1\}$.
Sample a random $x \in \mathbb{F}_p$.
Set $s \leftarrow 1$ if $x^3 + Ax^2 + x$ is a square in $\mathbb{F}_p$, else $s \leftarrow -1$.
return $(x, s)$
```

**Algorithm 2.** IsoImage.

```
Input: $A \in \mathbb{F}_p$ and a point $K \in E_{A,1}(\mathbb{F}_p)$.
Output: $A' \in \mathbb{F}_p$ such that $E_{A',1}$ is the codomain of the isogeny with kernel $\langle K\rangle$.
return $A'$ as defined in Theorem 3.
```

**Algorithm 3.** IsoEval.

```
Input: $A \in \mathbb{F}_p$ and two points $K, P \in E_{A,1}(\mathbb{F}_p)$.
Output: $\varphi(P)$, where $\varphi$ is the isogeny with kernel $\langle K\rangle$.
return $(f(x), \omega y f'(x))$ as defined in Theorem 3.
```

### 3.2. *Parameter choices*

As a reference for the rest of this paper, the following table gathers some of the proposed parameter choices for CSIDH. More details and motivations for such choices will be provided in the next sections.

| Name | $p$ | Secret-key space | Reference |
|---|---|---|---|
| CSIDH-512 | $4 \cdot \underbrace{3 \cdot 5 \cdot \ldots \cdot 373}_{\text{first 73 odd primes}} \cdot 587 - 1$ | $[-5, 5]^{74}$ | §4.1-3, [15] |
| | | $[-202, 202] \times \cdots \times [-1, 1]^{7}$ | §4.6, [14] |
| | | $[0, 10]^{74}$ | §5.1, [46] |
| $\text{SIMBA}_{5,11}$ | | $[0, 13]^{26} \times \cdots \times [0, 5]$ | §5.2, [46] |
| $\text{SIMBA}_{3,8}$ | | $[-5, 5] \times \cdots \times [-1, 1]$ | §5.3, [53] |
| CSURF-512 | $12 \cdot \underbrace{2 \cdot \ldots \cdot 389}_{\substack{\text{first 77 primes} \\ \text{except 347 and 359}}} -1$ | $[-137, 137] \times \cdots \times [-4, 4]^{25}$ | §4.5, [12] |
| CSIDH-1024 | $4 \cdot \underbrace{3 \cdot 5 \cdot \ldots \cdot 733}_{\text{first 129 odd primes}} \cdot 983 - 1$ | $[-2, 2]^{130}$ | [15] |
| | | $[-1, 1]^{130}$ | [18] |

**Algorithm 4.** Evaluating the class group action.

```
Input: $A \in \mathbb{F}_p$, $(e_1, \ldots, e_n) \in \mathbb{Z}^n$.
Output: $A' \in \mathbb{F}_p$ such that $[E_{A',1}] = ([\mathfrak{I}_1]^{e_1} \cdot \ldots \cdot [\mathfrak{I}_n]^{e_n}) \star [E_{A,1}]$.
while some $e_i \neq 0$ do
    $(x, s) \leftarrow \mathsf{RationalPoint}(A)$.
    $S \leftarrow \{i \mid \mathrm{sign}(e_i) = s\}$
    if $S \neq \emptyset$ then
        Set $P$ as a point on $E_{s \cdot A,1}(\mathbb{F}_p)$ whose x-coordinate is $s \cdot x$.
        $k \leftarrow \prod_{i \in S} \ell_i$
        $P \leftarrow [\frac{p+1}{k}]P$
        if $P \neq \infty$ then
            for $i \in S$ do
                $K \leftarrow [\frac{k}{\ell_i}]P$
                if $K \neq \infty$ then
                    $A \leftarrow \mathsf{IsoImage}(A, K)$
                    $P \leftarrow \mathsf{IsoEval}(A, K, P)$
                    $k \leftarrow \frac{k}{\ell_i}$
                    $e_i \leftarrow e_i - s$
                end if
            end for
            $A \leftarrow s \cdot A$
        end if
    end if
end while
return $A$
```

## 4. Speed-Up of the Class-Group-Action Evaluation

In this section, we discuss the techniques that have been proposed so far to speed up the execution of CSIDH. Some of them consist in using more efficient versions of IsoEval and IsoImage, based on optimized variants of Vélu's formulas (Secs. 4.1–4.3). The other speed-ups are obtained by varying the original CSIDH parameters (Secs. 4.4–4.6).

### 4.1. *Optimizing* IsoImage — *twisted Edwards curves*

While Montgomery curves enjoy efficient formulas for evaluating isogenies, twisted Edwards curves are well-suited for an efficient computation of image curves. This fact is exploited in [47], taking advantage of the correspondence between Montgomery curves and twisted Edwards curves (see Section 2.4.3). The gist of their method is to determine a twisted Edwards curve $\mathbb{F}_p$-birationally equivalent to $E_{A,1}$, then compute the twisted Edwards image curve corresponding to the action of $[\mathfrak{I}_i]$, and finally convert such curve to an $\mathbb{F}_p$-birationally equivalent Montgomery curve, which is exactly $E_{A',1} = E_{A,1}/\mathfrak{J}_i$. This procedure saves some field multiplications with respect to the formulas in Theorem 3, and is particularly effective for the largest values of $\ell_i$.

### 4.2. *Optimizing* IsoEval — *projective arithmetic*

Despite providing a practical method to calculate the action of a special element $[\mathfrak{J}_i] \in \mathcal{C}\ell(\mathcal{O})$, the formulas from Theorem 3 involve several inversions in $\mathbb{F}_p$, which negatively affect the running time of the resulting algorithm. A natural way to avoid these inversions is to use projective arithmetic, as already observed in [15, §8].

Given a Montgomery curve $E_{A,1} : y^2 = x^3 + Ax^2 + x$ over a field $K$, its projective form is given by the equation

$$E^*_{A,1} : Y^2 Z = X^3 + AX^2 Z + XZ^2.$$

We recall that every point $(x, y) \in E_{A,1}$ corresponds to a point in $E^*_{A,1}$, namely $[x : y : 1]$. Conversely, a point $[u : v : z] \in E^*_{A,1}$ with $z \neq 0$ corresponds to $(u/z, v/z) \in E_{A,1}$. The unique point $[u : v : z]$ in $E^*_{A,1}$ having $z = 0$ is equal to $[0 : 1 : 0]$ and corresponds to the point at infinity $\infty$ in $E_{A,1}$. This bijection allows the extension of the group law of $E_{A,1}$ to $E^*_{A,1}$.

It is possible to projectivize Theorem 3 as follows. Consider a Montgomery curve $E_{A,1} : y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_p$, and a point $P \in E_{A,1}(\mathbb{F}_p)$ of order $\ell = 2d + 1$. We denote by $\varphi$ the isogeny with domain $E_{A,1}$ and kernel $\langle P \rangle$. Let $Q$ be any point of $E_{A,1}$, and denote by $[x_i : y_i : z_i]$, $[x : y : z]$ and $[x' : y' : z']$ the coordinates of the projective points corresponding to $[i]P$, $Q$ and $\varphi(Q)$, respectively. Then we have [21, §5]

$$x' = x \left( \prod_{i=1}^{d} (x \cdot x_i - z \cdot z_i) \right)^2, \quad z' = z \left( \prod_{i=1}^{d} (x \cdot z_i - x_i \cdot z) \right)^2. \tag{7}$$

The above formulas are exploited in [15]. In particular, after computing $x_i$ and $z_i$ for $i = 1, \ldots, d$, the values $x \cdot x_i - z \cdot z_i$ and $x \cdot z_i - x_i \cdot z$ are obtained at the cost of $4d$ field multiplications and $2d$ field additions.

Costello and Hisil [21] also highlight the possibility of trading $2d$ field multiplications for $2d + 2$ extra field additions, obtaining an efficiency improvement. Namely, the formulas in (7) are replaced by

$$x' = x \left( \prod_{i=1}^{d} [(x - z)(x_i + z_i) + (x + z)(x_i - z_i)] \right)^2, \tag{8}$$

$$z' = z \left( \prod_{i=1}^{d} [(x - z)(x_i + z_i) - (x + z)(x_i - z_i)] \right)^2. \tag{9}$$

### 4.3. *Further optimizations of* IsoEval *and* IsoImage

Given an elliptic curve $E$ and an $\ell$-torsion point $P$ of $E$, Vélu's formulas provide an explicit expression for both a curve $E'$ and an isogeny $\varphi\colon E \to E'$ having kernel $\langle P \rangle$ (see Section 2.3). However, Algorithm 4 only requires the value of $\varphi$ at some points, together with the equation of the image curve. In fact, the expression of $\varphi$ as a rational function is unnecessary to evaluate the group action in CSIDH. Starting from this remark, the two independent papers that we discuss below, [5; 39], propose efficient methods to evaluate an isogeny of odd degree $\ell$.

#### 4.3.1. *Square-root Vélu's formulas*

The first method, introduced by Bernstein *et al.* in [5], starts with a Montgomery curve, so that the formulas from Theorem 3 can be used. In particular, the rational function $f(x)$ defined in (4) can be written as

$$f(x) = x^\ell \cdot \left( \frac{h_S(1/x)}{h_S(x)} \right)^2,$$

where $S = \{1, 2, \ldots, (\ell - 1)/2\}$ and, for any finite set of integers $S'$ containing no multiples of $\ell$,

$$h_{S'}(x) = \prod_{s \in S'} (x - x_{[s]P})$$

with $x_{[i]P}$ being the x-coordinate of the point $[i]P$.

Thus, in order to compute the image of $\varphi$, the isogeny with kernel $\langle P \rangle$, on some point $Q$, the goal is to evaluate the polynomial $h_S$ in an efficient way, i.e. with complexity $\tilde{O}(\sqrt{\ell})$ instead of $O(\ell)$. To do so, two subsets $I$ and $J$ of $S$ satisfying the following properties are constructed:

- the sizes of $I$ and $J$ are close to $\sqrt{\ell}$;
- $I + J$ and $I - J$ are as large as possible (that is, they both have size $|I| \cdot |J|$);
- $S \backslash T$ is small, where

$$T = ((I + J) \cup (I - J)).$$

Then the core idea is that of effectively computing $h_T$ in terms of $h_I$ and $h_J$ using resultant-based computations. Consequently, $h_S$ is obtained as $h_S = h_T \cdot h_{S \backslash T}$. Explicit expressions for $I, J$ and $h_T$ are provided in [5, §4].

Additionally, the proposed efficient evaluation of the isogeny $\varphi$ can also be exploited to find the parameter $A'$ of the image curve. In fact, by Theorem 3, one can just evaluate $\varphi$ at some 2-torsion point $(\alpha, 0)$ with $\alpha \in \mathbb{F}_{p^2}^*$ and set $A' = -(f(\alpha) + 1/f(\alpha))$. Alternatively, making use of Theorem 4, the image curve can be written as a twisted Edwards curve with

$$\hat{d} = \left( \frac{h_S(1)}{h_S(-1)} \right)^8 \left( \frac{A - 2}{A + 2} \right)^\ell,$$

where $A$ is the coefficient of the Montgomery curve $E_{A,1}$ from which the isogeny $\varphi$ is originating.

The resulting algorithm is asymptotically faster than the conventional algorithm based on Vélu's formulas from Theorem 3. Indeed, as anticipated, its computational cost is $\tilde{O}(\sqrt{\ell})$ instead of $O(\ell)$. This motivates the name *square-root Vélu*. The authors implement both their algorithm and the conventional one, observing a computational speed-up for $\ell \geq 113$ (see [5, A.3]). A more recent optimization of the same algorithm, realized by Adj *et al.* in [1], pushes this threshold down to 89.

#### 4.3.2. 2-*ADD-Skip method*

Kodera *et al.* [39] suggest a different method to evaluate an $\ell$-degree isogeny of kernel $\langle P \rangle$ and produce its image curve. The gist of this method is to avoid the explicit computation of all the points $P, [2]P, \ldots, [(\ell-1)/2]P$.

More precisely, the authors start with a Montgomery curve $E$ and an $\ell$-torsion point $P$ of $E$, and remark that, for any pair of distinct integers $m, n$, the x-coordinates of $[m+n]P$ and $[m-n]P$ can be expressed in terms of $[m]P$ and $[n]P$. The resulting formulas can be plugged into the map

$$f(x) = x \cdot \prod_{i=1}^{d} \left( \frac{x \cdot x_{[i]P} - 1}{x - x_{[i]P}} \right)^2$$

defined in Theorem 3, in such a way that the explicit computation of $[m+n]P$ and $[m-n]P$ is no longer required. This reason motivates the name 2-*ADD-Skip* for the proposed method. Indeed, whenever two distinct points $[m]P, [n]P$ have already been computed, there is no need to compute $[m \pm n]P$ explicitly. Thus, once some starting points are found using the conventional addition and doubling formulas (see e.g. [22, §3]), most of the remaining factors of $f(x)$ can be computed taking advantage of the 2-ADD-Skip method. For example, in [39, §4.2] it is shown that starting with $P, [2]P$ and $[3]P$ is an effective choice. A similar approach is adopted to compute the image curve. In particular, formula (6) for twisted Edwards curves can be recursively constructed making use of the 2-ADD-Skip method.

In [39, §5], the authors observe that their algorithm is not asymptotically faster than Bernstein, De Feo, Leroux and Smith's algorithm [5]. However, it requires a smaller number of field operations to compute some of the prime-degree isogenies when working with the CSIDH-512 set of parameters.

### 4.4. *The secret-key space*

Every subset $\mathcal{A}$ of $\mathbb{Z}^n$ determines a set of ideal classes

$$\mathcal{IC}_{\mathcal{A}} = \left\{ \prod_{i=1}^{n} [\mathfrak{I}_i]^{e_i} \,\middle|\, (e_1, \ldots, e_n) \in \mathcal{A} \right\} \subseteq \mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}]).$$

We say that $\mathcal{A}$ *covers* $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ if the equality $\mathcal{IC}_{\mathcal{A}} = \mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ holds. Given $\mathcal{A} \subset \mathbb{Z}^n$, it is usually challenging to prove that $\mathcal{A}$ covers $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$. Therefore, weaker conditions (for example, $|\mathcal{A}| \geq |\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])|$) are considered instead. From now on, with the name CSIDH we will also refer to all the variants of the original scheme that consider alternative secret-key spaces $\mathcal{A} \subset \mathbb{Z}^n$ that are heuristically assumed to cover $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$.

In the original CSIDH proof-of-concept implementation, the secret-key space is the set $[-B, B]^n$, where $B$ is a small positive integer for which the inequality $(2B+1)^n \geq |\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])|$ holds. In [15, §7.1], some heuristic arguments are presented to corroborate the assumption that this choice suffices to cover the entire ideal class group $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$.

Secret-key spaces have been targeted to improve the security and the efficiency of CSIDH, as we are going to show in the next two sections.

### 4.4.1. *Collisions*

*A priori*, *collisions* may happen when considering $[-B, B]^n$ as secret-key space. That is, there might exist two distinct vectors $(e_1, \dots, e_n)$, $(e'_1, \dots, e'_n) \in [-B, B]^n$ such that $[\mathfrak{I}_1]^{e_1} \cdot \dots \cdot [\mathfrak{I}_n]^{e_n} = [\mathfrak{I}_1]^{e'_1} \cdot \dots \cdot [\mathfrak{I}_n]^{e'_n}$. Onuki and Takagi [54] provide a family of them, showing that collisions actually exist. In particular, in [54, Theorem 3] it is proved that the element $[\mathfrak{I}_1] \cdot \dots \cdot [\mathfrak{I}_n]$ has order 3 in $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ (the same was independently observed in [16, Lemma 8]), when $p$ is a prime from a CSIDH set of parameters. Equivalently, the vectors

$$\dots, (e_1 - 3, \dots, e_n - 3), \quad (e_1, \dots, e_n), \quad (e_1 + 3, \dots, e_n + 3), \dots$$

represent the same ideal class. Therefore, if the secret-key space is $[-B, B]^n$ for some $B \geq 3$, then some keys represent the same elements in $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$. An effective workaround is the use of a different secret-key space having the form $[-B_1, B_1] \times \dots \times [-B_n, B_n]$, where at least one $B_i$ is less than 3.

As a consequence, both the action of $[\mathfrak{I}_1] \cdot \dots \cdot [\mathfrak{I}_n]$ and that of its inverse can be evaluated efficiently [54, Theorem 7].

**Theorem 5.** *Let $A \in \mathbb{F}_p$ such that $[E_{A,1}] \in \mathcal{E}\ell\ell_p(\mathbb{Z}[\sqrt{-p}], \pi)$. Then*

$$[\mathfrak{I}_1] \cdot \dots \cdot [\mathfrak{I}_n] \star [E_{A,1}] = [E_{A',1}] \quad \textit{and} \quad [\mathfrak{I}_1]^{-1} \cdot \dots \cdot [\mathfrak{I}_n]^{-1} \star [E_{A,1}] = [E_{A'',1}],$$

*where*

$$A' = -2 \cdot \frac{A+6}{A-2} \quad \textit{and} \quad A'' = 2 \cdot \frac{A-6}{A+2}.$$

The above formulas can be exploited in Algorithm 4 whenever $S$ (line 3) has cardinality $n$, i.e. whenever we need to evaluate the action of elements $\prod_{i=1}^{n} [\mathfrak{I}_i]^{e_i}$ such that no component of $(e_1, \dots, e_n)$ is zero.

### 4.4.2. *Optimized variants*

A subset $\mathcal{A} \subset \mathbb{Z}^n$ which covers $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ does not need to be of the form $[-B, B]^n$ for some positive integer $B$. Nakagawa *et al.* [51] deal with the problem of determining an optimal subset $\mathcal{A}_{\text{opt}} \subset \mathbb{Z}^n$ which minimizes $\mathbb{E}_{\mathbf{e} \in \mathcal{A}}[T(\mathbf{e})]$ among the subsets

$\mathcal{A} \subset \mathbb{Z}^n$ such that $|\mathcal{A}| \geq |\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])|$. Here, $T(\mathbf{e})$ denotes the number of field multiplications required to compute the action of $[\mathfrak{I}_1]^{e_1} \cdot \ldots \cdot [\mathfrak{I}_n]^{e_n}$, where $\mathbf{e} = (e_1, \ldots, e_n)$, while $\mathbb{E}_{\mathbf{e}} \in \mathcal{A}[T(\mathbf{e})]$ denotes the expected value of $T(\mathbf{e})$ taken over a uniform choice of $\mathbf{e} \in \mathcal{A}$.

In [51], a procedure to tackle the above-mentioned optimization problem is presented. In particular, restricting its attention to sets $\mathcal{A}$ such that $\mathcal{A} = \{\mathbf{e} \in \mathbb{Z}^n \mid T(\mathbf{e}) \leq r\}$ for some $r \in \mathbb{Z}_{\geq 0}$, the procedure computes an approximate lower bound for all $r$ such that $\{\mathbf{e} \in \mathbb{Z}^n \mid T(\mathbf{e}) \leq r\}$ has cardinality greater than or equal to $|\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])|$. A concrete value for the approximate lower bound of $r$, denoted by $r_{\text{opt}}$, is provided for CSIDH-512, whose parameters consist of the 512-bit prime $p$ of the form $p = 4 \cdot \ell_1 \cdot \ldots \cdot \ell_{74} - 1$, where $\ell_1$ through $\ell_{73}$ are the first 73 odd primes and $\ell_{74} = 587$. In this case, $r_{\text{opt}} = 336428$, while the average computational cost $T_{\text{opt}}$ in $\mathcal{A}_{\text{opt}} = \{e \in \mathbb{Z}^n \mid T(e) \leq r_{\text{opt}}\}$ is $T_{\text{opt}} \sim 332000$.

However, uniformly sampling from the proposed optimal key space $\mathcal{A}_{\text{opt}}$ appears to be very hard. Therefore, in [51] a "transposition" of this set is provided. To be more precise, it is proven that the average value of $T(\mathbf{e})$ for $\mathbf{e}$ varying in the $L_1$-ball in $\mathbb{Z}^n$ with center $(0, \ldots, 0)$ and radius 152 is 355804, which is only 1.08 times bigger than $T_{\text{opt}}$. Hence, there is the option of replacing $\mathcal{A}_{\text{opt}}$ with this $L_1$-ball, in order to easily sample from the chosen secret-key space while retaining (almost) the same efficiency provided by the secret-key space $\mathcal{A}_{\text{opt}}$.

### 4.5. *CSIDH on the surface*

The CSIDH protocol and its optimizations focus on supersingular elliptic curves over $\mathbb{F}_p$ (where $p$ is a prime) that lie on the floor. This choice is due to the possibility of representing each element of $\mathcal{E}\ell\ell_p(\mathbb{Z}[\sqrt{-p}], \pi)$ with a unique element of the base field $\mathbb{F}_p$ (see Theorem 2).

In [12], Castryck and Decru exhibit a new elliptic curve model that allows transposing the above-mentioned property, and consequently the CSIDH scheme, to supersingular elliptic curves on the surface. Moreover, they propose a new 512-bit prime and a suitable secret-key space which produces a speed-up of about 5.68% with respect to CSIDH instantiated with the CSIDH-512 set of parameters.[f] The proposed prime is $p = 2^3 \cdot 3 \cdot \ell_1 \cdot \ldots \cdot \ell_{74} - 1$, where the $\ell_i$'s are the 74 consecutive primes from 3 to 389 skipping 347 and 357. The secret keys are sampled from $[-137, 137] \times [-4, 4]^3 \times [-5, 5]^{46} \times [-4, 4]^{25}$. The following theorem [12, Proposition 4] ensures that each $\mathbb{F}_p$-isomorphism class in $\mathcal{E}\ell\ell_p(\mathcal{O}_{\mathbb{Q}(\sqrt{-p})}, \pi)$ can be identified by a unique element of $\mathbb{F}_p$.

**Theorem 6.** *Let $p$ be a prime such that $p \equiv 7 \pmod 8$ and $E$ a supersingular elliptic curve over $\mathbb{F}_p$. Then, $E$ lies on the surface if and only if there exists $A \in \mathbb{F}_p$*

[f] Note that the comparison is made with the original proof-of-concept CSIDH implementation, without taking into account some of the improvements seen in this section.

*such that $E$ is $\mathbb{F}_p$-isomorphic to the elliptic curve $y^2 = x^3 + Ax^2 - x$. Moreover, if such $A$ exists, it is unique.*

### 4.6. *Radical isogenies*

As suggested by Algorithm 4, generating a point to compute an isogeny of assigned degree is a costly operation. In fact, not only does it require some scalar multiplications, but it may also fail. This problem is particularly relevant when isogenies of small degrees must be computed. To mitigate this issue, Castryck *et al.* [14] describe a deterministic procedure to obtain a chain of $\ell$-isogenies starting from an elliptic curve with a given $\ell$-torsion point. For $\ell \leq 37$, this method — improved in the follow-up paper by Castryck *et al.* [13] — achieves a computational speed-up compared to the standard method (that is, calling RationalPoint to sample a new $\ell$-torsion point for each $\ell$-isogeny).

In more detail, the procedure is given as input a supersingular elliptic curve $E$ defined over $\mathbb{F}_p$ and such that $P = (0, 0)$ is an $\ell$-torsion point (for $\ell > 3$, the Tate curve from Section 2.4.2 can be used). The goal is to determine the coordinates of an $\ell$-torsion point $P' \in E' = E/\langle P\rangle$ such that composing the isogeny $\varphi\colon E \to E'$ with kernel $\langle P\rangle$ and the isogeny $\psi\colon E' \to E'/\langle P'\rangle$ with kernel $\langle P'\rangle$ yields a cyclic isogeny of degree $\ell^2$. Castryck, Decru and Vercauteren prove that the coordinates of $P'$ can be expressed as an algebraic combination of the coefficients of $E$ and an $\ell$th root of a suitable element $\rho \in \mathbb{F}_p$. Explicit formulas are derived in [13, §4–5] for $\ell \leq 37$. Once $P'$ has been computed, it can be "transformed" into $(0, 0)$ by finding an isomorphism $E' \to \tilde{E}$ mapping $P'$ to $(0, 0)$. The whole process can be then iterated arbitrarily many times. The coordinates of $P'$ (and consequently the coordinates of $\tilde{E}$), though, lie in $\mathbb{F}_p(\sqrt[\ell]{\rho})$, which is in general larger than $\mathbb{F}_p$. Nonetheless, it is possible to control the field of definition of $P'$ by choosing a suitable prime $p$ [14, §5]. For example, in [14, §6] the prime $p$ mentioned in Section 4.5 is used.

In order to fully take advantage of the procedure, the following secret-key space is proposed:

$$\begin{gathered}[-202, 202] \times [-170, 170] \times [-95, 95] \times [-91, 91] \times [-33, 33] \times [-29, 29] \\ \times [-6, 6]^{20} \times [-5, 5]^{14} \times [-4, 4]^{10} \times [-3, 3]^{10} \times [-2, 2]^8 \times [-1, 1]^7.\end{gathered}$$

The class-group-action evaluation is then hybrid: first, isogenies corresponding to the primes $2, 3, \ldots, 37$ are computed via the described procedure. Then, the remaining indices are exhausted by following Algorithm 4. This variant of CSIDH achieves an average speed-up of 26% compared to the CSIDH implementation which exploits the optimization presented in Section 4.3.1.

## 5. Constant-Time Class-Group-Action Evaluation

The security of the CSIDH scheme relies on the hardness of the decisional Diffie–Hellman (DDH) problem for the CSIDH group action — that is, given $E_0$ and two public keys $[\mathfrak{a}] \star [E_0]$ and $[\mathfrak{b}] \star [E_0]$, distinguishing $[\mathfrak{a}] \star ([\mathfrak{b}] \star [E_0])$ from the action $[\mathfrak{c}] \star [E_0]$ of a random element $[\mathfrak{c}]$. In analogy with the classical Diffie–Hellman key-exchange protocol, the DDH problem for CSIDH is assumed to be, in general, no easier than the following *parallelization problem* [23, §2]: computing $[\mathfrak{a}] \star ([\mathfrak{b}] \star [E_0])$ given $E_0$ and two public keys $[\mathfrak{a}] \star [E_0]$ and $[\mathfrak{b}] \star [E_0]$. Montgomery and Zhandry [49] recently showed that this parallelization problem is, in turn, quantumly equivalent to the following *vectorization problem*: recovering the private key $[\mathfrak{a}]$ from the knowledge of $E_0$ and $[\mathfrak{a}] \star [E_0]$.[g] Therefore we will focus on the vectorization problem, also known as Group Action Inverse Problem (GAIP), which we formalize below.

**Definition 1 (Group Action Inverse Problem (GAIP)).** Let $[E_0]$ be an element in $\mathcal{E}\ell\ell_p([\mathbb{Z}(\sqrt{-p}], \pi)$, where $p > 3$ is a prime. Given $[E]$ sampleder uniformly at random from $\mathcal{E}\ell\ell_p(\mathbb{Z}[\sqrt{-p}], \pi)$, the $\mathrm{GAIP}_p$ problem consists in finding the unique element $[\mathfrak{a}] \in \mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ such that $[\mathfrak{a}] \star [E_0] = [E]$.

The best known classical algorithm to solve the $\mathrm{GAIP}_p$ problem has time complexity $O(\sqrt{N})$, where $N = |\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])| \sim \sqrt{p}$, while the best known quantum algorithm is Kuperberg's algorithm for the hidden shift problem [41, 42]. The latter has a subexponential complexity, but its concrete estimate is still an active area of research [18, 55].

The original proof-of-concept implementation of CSIDH (and those of the subsequent improvements discussed in the previous section) falls prey to side-channel attacks. For example, timing attacks to Algorithm 4 give some information about the private keys. This is mainly due to the fact that different private keys require a different number of isogeny computations. Two possible strategies to mitigate this leakage have been proposed so far:

(1) In the first one [7], the computational cost for the evaluation of the action of any element $\prod_{i=1}^{n} [\mathfrak{I}_i]^{e_i}$ with $(e_1, \ldots, e_n) \in [-B, B]^n$ is always the same, independently of $(e_1, \ldots, e_n)$ and the randomness used; this strategy has a negligible failure probability.
(2) In the second strategy [46], the computational cost for the evaluation of the action of $\prod_{i=1}^{n} [\mathfrak{I}_i]^{e_i}$ is independent of $(e_1, \ldots, e_n)$ itself, but might vary depending to the randomness used.

Both the strategies are labeled as "constant-time", despite being substantially different. We stress that they address only a certain type of side-channel attacks. In fact, other attacks — based for example on power traces or electromagnetic measurements — require even more involved countermeasures and have so far not been considered much in the isogeny literature.

[g] More precisely, they prove this equivalence for a family of group-action-based key exchanges. On the other hand, they show that the same cannot hold generically for the DDH problem for group-action-based key exchanges.

The gist of constant-time implementations is to perform some extra "dummy" operations that do not affect the output of the algorithm. We consider a first example from [47]: for each prime $\ell_i$ in the definition of $p$, the number of $\ell_i$-isogenies to compute when evaluating the action of $\prod_{i=1}^{n}[\mathfrak{J}_i]^{e_i}$ is fixed to the maximum value, i.e. $B$, independently of $(e_1, \ldots, e_n)$. In other words, if $|e_i| < B$, then $B - |e_i|$ extra artificial isogenies of degree $\ell_i$ are computed. Moreover, to prevent cache-based side-channel attacks, even the literal addresses of instructions executed by the implementation should be independent of the secret key. This means that, in order to avoid conditional branching, isogenies and artificial isogenies (of a given degree $i$) need to be computed by calling the same "physical" code in memory. Since the artificial isogenies are useless for the computation of the action of $\prod_{i=1}^{n}[\mathfrak{J}_i]^{e_i}$, they are called *dummy isogenies* [47].

However, the computational costs for evaluating the actions of two elements $\prod_{i=1}^{n}[\mathfrak{J}_i]^{e_i}$ and $\prod_{i=1}^{n}[\mathfrak{J}_i]^{e'_i}$ of $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ might differ even when $|e_i| = |e'_i|$ for every $i \in \{1, \ldots, n\}$, and therefore even when computing dummy isogenies. For example, Meyer and Reith [47, §6] note that the running time for computing the action of $\prod_{i=1}^{n}[\mathfrak{J}_i]^{5}$ when working with the CSIDH-512 set of parameters[h] is higher than the running time for computing the action of $\prod_{i=1}^{n}[\mathfrak{J}_i]^{(-1)^{i+1}5}$. The reason for this discrepancy is that, once a random x-coordinate is sampled (line 2 of Algorithm 4), it can be used only to compute isogenies for those indices belonging to the set $S$ (line 3 of Algorithm 4). For the element $\prod_{i=1}^{n}[\mathfrak{J}_i]^{5}$, when $s = -1$, the set $S$ is always empty — and so the sampled $x$ cannot be used — while when $s = 1$ the set $S$ can contain up to $n$ elements. When $|S| = n$, the scalar factor in line 7 is 4, while the scalar factors in line 10 of Algorithm 4 are $(p+1)/(4\ell_i)$ for each $i \in \{1, \ldots, n\}$, whose size is close to the size of $p$. For the element $\prod_{i=1}^{n}[\mathfrak{J}_i]^{(-1)^{i+1}5}$, in comparison, the scalar factors in lines 7 and 10 have similar sizes to that of $\sqrt{p}$.

In light of the above observation, Meyer *et al.* [46] propose replacing the symmetric set of private keys $[-B, B]^n$ with the asymmetric set $[0, 2B]^n$. We stress that this choice should not affect the hardness assumption on the GAIP problem, as the heuristic assumption that $[-B, B]^n$ covers $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ extends to $[0, 2B]^n$ (if $(e_1, \ldots, e_n), (e'_1, \ldots, e'_n) \in [-B, B]^n$ are such that $\prod_{i=1}^{n}[\mathfrak{J}_i]^{e_i} = \prod_{i=1}^{n}[\mathfrak{J}_i]^{e'_i}$, then also $(e_1 + 5, \ldots, e_n + 5), (e'_1 + 5, \ldots, e'_n + 5) \in [0, 2B]^n$ determine the same element of $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$, and viceversa).

The following sections detail some further constant-time implementations of the CSIDH scheme.

### 5.1. *Redesigning* **RationalPoint** — *Elligator* 2

Given a Montgomery curve $E_{A,1} \colon y^2 = x^3 + Ax^2 + x$, defined over $\mathbb{F}_p$ and such that $[E_{A,1}] \in \mathcal{E}\ell\ell_p(\mathbb{Z}[\sqrt{-p}], \pi)$, the constant-time implementation of CSIDH presented

[h] They do not use the improvement described in Section 4.4.1. Otherwise, using Theorem 5, the computation would have been trivial.

**Algorithm 5.** Elligator2.

```
Input: The coefficient $A \in \mathbb{F}_p^*$ of a Montgomery curve $E_{A,1}$, $s \in \{1, -1\}$.
Output: x-coordinate of a rational point in $E_{s \cdot A,1}$.
Uniformly sample an element $u$ from $\{2, \ldots, \frac{p-1}{2}\}$.
$v \leftarrow \frac{A}{u^2-1}$
Set $e$ as the Legendre Symbol of $v^3 + Av^2 + v$ over $p$.
if $e == s$ then
    return $v$
else
    return $-v - A$
end if
```

in [7] efficiently samples x-coordinates of random rational points lying on $E_{A,1}$, or on $E_{-A,1}$. In particular, the method exploits the fact that, given $x_0 \in \mathbb{F}_p$ and $y_0 = (x_0^3 + Ax_0^2 + x_0)^{\frac{p+1}{4}}$,

$$y_0^4 = (x_0^3 + Ax_0^2 + x_0)^{p+1} = (x_0^3 + Ax_0^2 + x_0)^2.$$

Then $y_0^2 = \pm(x_0^3 + Ax_0^2 + x_0)$ and, consequently, either $(x_0, y_0)$ is a point of $E_{A,1}$, or $(-x_0, y_0)$ is a point of the curve $E_{-A,1}$, which is used to compute the action of elements $[\mathfrak{I}_i]^{-1}$ (see Section 3.1).

The details of this method to sample x-coordinates of rational points in $E_{A,1}$ or in $E_{-A,1}$ are depicted in Algorithm 5. It is called the *Elligator* 2 *map*, and it was designed by Bernstein *et al.* in [6]. We note that the algorithm takes as input a Montgomery curve $E_{A,1}$, with $A \in \mathbb{F}_p^*$ and $s \in \{-1, 1\}$, and it returns the x-coordinate of a point on $E_{s \cdot A,1}$. The Elligator 2 map can be extended to the case $A = 0$, i.e. the case where $E_{A,1}$ is the curve $E_0$, setting $v = u$ (instead of $v = \frac{A}{u^2-1}$).

A constant-time implementation of CSIDH that relies on the Elligator 2 map is detailed in Algorithm 6 [7, Algorithm 6.1]. For every $i \in \{1, \ldots, n\}$, the same loop is repeated $r_i$ times (for some positive integer $r_i$). If, after the $r_i$ steps, the computation of the action of $[\mathfrak{I}_i]^{e_i}$ is incomplete, the algorithm fails. The parameters $r_i$ can be tuned to increase/decrease the success probability of the algorithm.

The algorithm takes constant time as any of the conditions "$\mathsf{w} \leftarrow \mathsf{w}'$ if $b$" means that $\mathsf{w}'$ and $b$ are computed anyways.

The role of the Elligator 2 map in Algorithm 6 is to sample the x-coordinate of a point $P$ in $E_{A,1}$ or $E_{-A,1}$. However, it may happen that $[\frac{p+1}{\ell_i}]P = \infty$, i.e. the order of $P$ is not divisible by $\ell_i$. We recall that, given a uniformly random rational point $P \in E_{A,1}(\mathbb{F}_p)$, the probability that its order is not divisible by $\ell_i$ is $1/\ell_i$. Moreover, among the $(p-3)/2$ points that could be sampled by the Elligator 2 map, at most $(p+1)/\ell_i$ have order not divisible by $\ell_i$. Then, the probability that the Elligator 2 map samples a point $P$ whose order is not divisible by $\ell_i$ is upper bounded by

$$\left(\frac{2}{\ell_i}\right)\frac{p+1}{p-3} \sim \frac{2}{\ell_i}.$$

**Algorithm 6.** Constant-time class-group-action evaluation.

```
Input: The coefficient $A \in \mathbb{F}_p^*$ of a Montgomery curve $E_{A,1}$, $(e_1, \dots, e_n) \in [-B, B]^n$ and positive integers $r_1, \dots, r_n$.
Output: $A' \in \mathbb{F}_p$ s.t. $[E_{A',1}] = \prod_{i=1}^n [\mathfrak{I}_i]^{e_i} \star [E_{A,1}]$ or failure
for $i \in \{1, \dots, n\}$ do
    for $j \in \{1, \dots, r_i\}$ do
        $s \leftarrow \mathsf{sign}(e_i)$ with $\mathsf{sign}(e_i) \in \{-1, 0, 1\}$
        $P \leftarrow \mathsf{Elligator2}(A, s)$
        $Q \leftarrow [\frac{p+1}{\ell_i}]P$
        Compute $A'$, with $E_{A',1} \simeq E_{A,1}/\langle Q \rangle$, if $Q \neq \infty$
        $A \leftarrow s \cdot A'$ if $Q \neq \infty$ and $s \neq 0$
        $e_i \leftarrow e_i - s$ if $Q \neq \infty$
    end for
end for
if $(e_1, \dots, e_n) == (0, \dots, 0)$ then
    return $A$
else
    return failure
end if
```

In [7], it is claimed that the heuristic probability is almost exactly $1/\ell_i$, i.e. it is heuristically close to the uniform one.

In terms of efficiency, the most expensive step in Algorithm 5 is the computation of $A/(u^2 - 1)$ in line 2, since it requires a field inversion. A possible workaround [7, §4.3] consists in replacing the random sampling in line 1 by a precomputed list of values $1/(u^2 - 1)$, for $u \in \{2, \dots, \frac{p-1}{2}\}$, to be used in line 2. In [46, §5.3], Meyer *et al.* claim that pre-computing $1/(u^2 - 1)$ for 10 values of $u$ is heuristically enough to ensure that, for each $i$, at least one of the output points has order divisible by $\ell_i$. They use this variant of the Elligator 2 map together with dummy isogenies and the asymmetric key-space $[0, 2B]^n$ to obtain an implementation of CSIDH whose running time is independent of the private key. Since their algorithm is no-failure, if the 10 precomputed values for the Elligator 2 map do not give rise to points that can be exploited to compute the remaining isogenies, then their algorithm retreats to the original Elligator 2 map which samples (almost) random points. However, this step makes the evaluation of the action of an element $\prod_{i=1}^n [\mathfrak{I}_i]^{e_i}$ not independent of $(e_1, \dots, e_n)$. In fact, the time required by the Elligator 2 map (initially running the variant with precomputations and then, if necessary, the original one) to produce suitable rational points on the current supersingular curve $E_{A,1}$ depends on $A$, which itself depends on $(e_1, \dots, e_n)$.

To fix this issue, which was noticed in [17], Cervantes-Vázquez *et al.* propose a projective Elligator 2 map which uses randomness while still avoiding inversions

[17, §3]. We observe that this projective variant leaves unchanged the probability of finding a point with a fixed order.

## 5.2. *Simba*

In [46], Meyer *et al.* propose a further technique to speed up their previous constant-time implementation of CSIDH [47]. This technique consists in splitting $\{1, \ldots, n\}$ into subsets, and working separately within each of them. To informally explain the technique and its relevance, we take into consideration Algorithm 4. Despite being substantially different from the algorithm in [46] (partially discussed in the previous section), the scalar multiplications the two algorithms perform are quite similar, and therefore Algorithm 4 is well-suited for the exposition.

Assume that the secret-key space is $[0, 2B]^n$, and set $s = 1$. Then the set $S$ (line 3) gathers all the positive indices corresponding to isogenies that are yet to be computed. As the following examples show, the size of $S$ affects the cost of the point multiplications $[(p+1)/k]P$ (line 7) and those of the form $[k/\ell_i]P$ executed within the for-loop (line 10), where $k = \prod_{i \in S} \ell_i$.

Suppose that, after $j$ iterations of the while loop, the input vector has been updated into a vector $(e_1, \ldots, e_n)$ having Hamming weight equal to $n$ (equivalently, none of its entries are zero). Therefore, $S = \{1, \ldots, n\}$, $k = (p+1)/4$ and $(p+1)/k = 4$. This means that one point multiplication, namely $[(p+1)/k]P$, is really cheap, while many of the others, i.e. $[k/\ell_1]P, [k/(\ell_1\ell_2)]P, \ldots$, are quite expensive, being their scalar factors *close* to $k$. On the other hand, if $(e_1, \ldots, e_n)$ has Hamming weight equal to $\lfloor n/2 \rfloor$, the sizes of the scalar factors could be more balanced. Suppose that $S$ is equal to $\{2, 4, \ldots, h\}$, where $h = n$ if $n$ is even, $h = n - 1$ otherwise. Then both $(p+1)/k$ and $k/\ell_2$ are approximately equal to $\sqrt{p}$ (with the other scalar factors $k/(\ell_2\ell_4), k/(\ell_2\ell_4\ell_6), \ldots$ which progressively decrease). It is then easy to see that the computational cost for the scalar multiplications within a single loop determined by $S = \{1, \ldots, n\}$ is bigger than that for the scalar multiplications within two *complementary* loops determined by $S_1 = \{1, 3, \ldots\}$ and $S_2 = \{2, 4, \ldots\}$, respectively.

In [46], Meyer *et al.* propose splitting the set $\{1, 2, \ldots, n\}$ into multiple batches $S_1, \ldots, S_m$, with $m < n$ being a positive integer, in order to artificially recreate the conditions of the second example discussed above. This is obtained by introducing a for-loop (indexed by $j \in \{1, \ldots, m\}$) before the construction of $S$, and defining $S$ as the set $S = \{i \in S_j \mid e_i > 0\}$ within each iteration of this loop.

This proposal gives rise to two natural questions: which is the optimal value for $m$? And what is the optimal distribution of the indices in the subsets $S_1, \ldots, S_m$? For the CSIDH-512 parameters, setting $S_j = \{j, m+j, 2m+j, \ldots\}$, a heuristic analysis shows that the optimal choice for $m$ is 5 (see [46, §5.3]).

We conclude this section noticing that, given an elliptic curve $E_{A,1}$, the probability that a random rational point has order divisible by a small prime $\ell_i$ is low. Therefore, it is expected that, after some rounds, every batch contains only a few indices $i$ such that $e_i$ is not zero (assuming that the small indices are distributed across the batches). As soon as this happens, it is convenient to merge the batches. The algorithm incorporating the subdivision in $m$ batches and the merging of those batches after $\mu$ rounds is named $\mathrm{SIMBA}_{m,\mu}$.

An alternative asymmetric private-key space in which, for each $i \in \{1, \ldots, n\}$, $e_i$ belongs to a tailored interval $[0, B_i]$ instead of a fixed interval $[0, 2B]$ is also taken into consideration in [46].

### 5.3. *CSIDH keeping two points*

The use of an asymmetric key space was introduced to avoid the information leakage due to the fact that a random x-coordinate determines the value of $s \in \{-1, 1\}$ and allows computing only the isogenies corresponding to the elements of $(e_1, \ldots, e_n)$ whose sign coincides with $s$. An alternative remedy consists in exploiting the Elligator 2 map and the fact that it simultaneously generates points on both the elliptic curves $E_{A,1}$ and $E_{-A,1}$.

Building on this idea, Onuki *et al.* [53] introduce a new secret-key-independent implementation of CSIDH, which works with a secret-key space whose vectors have both negative and positive entries. The implementation considered in [53] for benchmarking — which uses both dummy isogenies and SIMBA — is slightly faster, for the CSIDH-512 set of parameters, than that in [46] for the same set of parameters. In particular, the SIMBA parameters giving the most efficient implementation are $m = 3$ and $\mu = 8$. Furthermore, each secret exponent $e_i$ is chosen from a bespoke interval $[-B_i, B_i]$, where the vector $(B_1, \ldots, B_n)$ is defined as follows:

$$\begin{gathered}
[5, 6, 7, 7, 7, 7, 8, 8, 8, 9, 10, 10, 10, 10, 9, 9, 9, 8, 7, 7, 7, 7, 7, \\
7, 7, 7, 7, 7, 7, 7, 6, 6, 6, 6, 6, 5, 5, 5, 5, 5, 5, 5, 4, 4, 4, 4, 4, 4, \\
4, 4, 4, 4, 4, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 2, 2, 2, 2, 1].
\end{gathered}$$

### 5.4. *Optimal strategies*

In [36, §4], Jao and De Feo present an *optimal strategy approach*[i] for computing isogenies within the isogeny-based SIDH scheme. Hutchinson *et al.* [35] extend this approach to the constant-time implementations in [46; 53] using linear-programming techniques. The result is an implementation for the CSIDH-512 set of parameters which is 5.06% faster than the 2-torsion points variant in [17] that adopts all the improvements enlisted so far (point-evaluation speed up, use of twisted Edwards curves, projective Elligator map and SIMBA algorithm).

Chi-Domínguez and F. Rodríguez-Henríquez have independently extended the use of strategies to the CSIDH scheme [19], achieving a moderate speed-up with

[i] The word "optimal" here refers to the number of field operations as the goal of the approach is to minimize this number.

respect to the implementation proposed in [35], even without embracing the use of the SIMBA algorithm.

### 5.5. *CTIDH*

In [3], Banegas *et al.* introduce a new constant-time implementation of CSIDH, called CTIDH. The gist of CTIDH is to partition into batches the primes $(\ell_1, \ldots, \ell_n)$ forming the chosen prime $p$. Once the number of batches $\mathcal{B}$ is fixed, the sequence of batch sizes is denoted by $N = (N_1, \ldots, N_{\mathcal{B}}) \in (\mathbb{Z}_{\geq 0})^{\mathcal{B}}$, where $\sum_{i=1}^{\mathcal{B}} N_i = n$. Unlike the constant-time implementations we have presented so far, there is no fixed number of isogenies to compute for each prime $\ell_i$. For each batch, a bound $m_i$ is imposed instead. In particular, the secret-key space is

$$\left\{ ((e_{1,1}, \ldots, e_{1,N_1}), \ldots, (e_{\mathcal{B},1}, \ldots, e_{\mathcal{B},N_{\mathcal{B}}})) \,\middle|\, \sum_{j=1}^{N_i} |e_{i,j}| \leq m_i \text{ for } 1 \leq i \leq \mathcal{B} \right\}.$$

For the $i$th batch (of size $N_i$), $m_i$ isogenies are computed via the square-root Vélu's formulas, which are well-suited for a constant-time procedure. More precisely — as one can see from Section 4.3.1 — the computations used for an $\ell_j$-isogeny starting from an elliptic curve $E_{A,1}$ can be used to obtain an $\ell_i$-isogeny from the same curve $E_{A,1}$, where $\ell_i < \ell_j$. Therefore, to compute an isogeny of degree $\ell_i$ in the batch, the coefficients for the isogeny with maximal degree $L$ are calculated first. Then, only the ones necessary to compute the isogeny of degree $\ell_i$ are used.

It is worth noting that the above algorithm is not readily independent from the secret key, since the probability of failure (due to a point of non-suitable order) may still leak some information. To overcome this problem, the success probability is "artificially" tweaked as shown in [3, §5.2.1].

The generation of optimal parameters $\mathcal{B}$, $(N_1, \ldots, N_{\mathcal{B}})$ and $(m_1, \ldots, m_{\mathcal{B}})$ represents a complicated optimization problem. The approach proposed in [3, §6] finds a "local" optimum with respect to an initial choice of $(N'_1, \ldots, N'_{\mathcal{B}})$ and $(m'_1, \ldots, m'_{\mathcal{B}})$. The resulting implementation gives better performance compared to the constant-time implementations we have presented so far.

## 6. Computations in a Class Group with Known Structure

The set of distinct primes $\ell_1, \ldots, \ell_n$ uniquely determines the prime $p$ in a set of parameters for CSIDH. It also determines the order $\mathbb{Z}[\sqrt{-p}]$ in the imaginary quadratic field $\mathbb{Q}(\sqrt{-p})$. The core of CSIDH is the action of the ideal class group $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ on the supersingular elliptic curves $E$ over $\mathbb{F}_p$ lying on the floor, i.e. such that $\mathsf{End}_{\mathbb{F}_p}(E) \simeq \mathbb{Z}[\sqrt{-p}]$. This fact motivates a more in-depth study of the structure of $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$.

The cardinality of $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$, called *class number* of $\mathbb{Z}[\sqrt{-p}]$, is finite. The best known algorithms for the class number computation have sub-exponential complexity with respect to the discriminant of the quadratic field $\mathbb{Q}(\sqrt{-p})$ (which is equal to $p$ in the CSIDH setting, being $p \equiv 3 \pmod 4$). This makes the computation of $|\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])|$ extremely heavy for primes of cryptographic size.

The impracticality of computing the class number of an order exploited by CSIDH for some set of parameters implies that the structure of the ideal class group $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ remains unknown. In particular, the fact that $[-B, B]^n$ covers $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$, along with the fact that a uniform sampling in $[-B, B]^n$ determines a uniform distribution in $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$, can only be assumed on the basis of heuristic arguments. Moreover, it is not known, *a priori*, what are the collisions in $\{\prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i} \mid (e_1, \ldots, e_n) \in [-B, B]^n\}$.

This scenario does not represent a substantial problem for CSIDH and its security, but it is a major obstacle for the design of efficient CSIDH-based digital signature schemes. As a result, the construction of the first practical isogeny-based digital signature scheme, CSI-FiSh [8], required a record class number computation for determining the structure of $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ for the CSIDH-512 set of parameters, mentioned in Section 4.4.2.

The class number computation executed by Beullens *et al.* in [8] determined that, for the prime $p = 4 \cdot \ell_1 \cdot \ldots \cdot \ell_{74} - 1$ in the CSIDH-512 set of parameters, it holds that

$$\begin{aligned} |\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])| = N &= 3 \cdot 37 \cdot 1407181 \cdot 51593604295295867744293584889 \\ &\quad \cdot 31599414504681995853008278745587832204909 \\ &\sim 2^{257.136}. \end{aligned}$$

As a consequence, $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ is a cyclic group,[j] and it is shown that the element $[\mathfrak{I}_1] = [(3, \sqrt{-p}{-}1)]$, which we denote by $\mathfrak{g}$, is a generator. Furthermore, the discrete logarithm of $[\mathfrak{I}_i]$ to the base $\mathfrak{g}$ is known for every $i \in \{2, \ldots, 74\}$.

Since $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ is isomorphic to the additive group $\mathbb{Z}/N\mathbb{Z}$, where $N$ is the class number of $\mathbb{Z}[\sqrt{-p}]$, each of its elements can be identified with an integer in $\{0, \ldots, N-1\}$. In particular, we denote by $e$ the integer corresponding to the secret key $(e_1, \ldots, e_n)$, i.e. $\mathfrak{g}^e = \prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i}$. As we discuss below, the existence of a canonical representation for elements of $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ is essential for the CSIDH-based digital signature CSI-FiSh [8].

CSI-FiSh is a Fiat–Shamir signature, i.e. it is obtained by turning an isogeny-based interactive identification protocol, sketched by Stolbunov in his Ph.D. thesis [59], into a non-interactive one by means of the Fiat–Shamir transformation [31]. The two actors of the interactive protocol are a prover, producing a proof $\sigma$ by means of their private key $(e_1, \ldots, e_n)$, and a verifier, who uses the prover's public key $[E_S] = \prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i} \star [E_0]$ — where $[E_0]$ is a fixed element of $\mathcal{E}\ell\ell_p(\mathbb{Z}[\sqrt{-p}], \pi)$ — to verify the validity of the proof $\sigma$.

[j] We observe that 3, 37, 1407181, 51593604295295867744293584889, 31599414504681995853008278745587832204909 are primes, and that $\mathcal{C}\ell(\mathbb{Z}[\sqrt{-p}])$ is an abelian group.

According to Stolbunov's proposal, while producing the proof $\sigma$, the prover samples a random ephemeral private key $(r_1, \ldots, r_n)$ and computes the commitment $[E_{\mathsf{com}}] = \prod_{i=1}^{n}[\mathfrak{I}_i]^{r_i} \star [E_0]$, which will be part of the proof. Furthermore, depending on the value of a flag bit $b$, the proof will contain also $(r_1, \ldots, r_n)$ when $b = 0$ or $(e_1 - r_1, \ldots, e_n - r_n)$ otherwise. To verify the validity of $\sigma$, the verifier checks that $\prod_{i=1}^{n}[\mathfrak{I}_i]^{r_i} \star [E_0]$ is equal to $[E_{\mathsf{com}}]$ (part of the proof) in the first case, or that $\prod_{i=1}^{n}[\mathfrak{I}_i]^{e_i - r_i} \star [E_{\mathsf{com}}]$ is equal to $[E_S]$.

Unfortunately, the simple scheme sketched above is flawed, as the vector $(e_1 - r_1, \ldots, e_n - r_n)$ leaks information about the private key $(e_1, \ldots, e_n)$. For example, the mean of the distribution of this vector equals the mean of the distribution of $(-r_1, \ldots, -r_n)$, shifted by the secret vector $(e_1, \ldots, e_n)$. In order to fix this issue, an initial remedy was proposed by De Feo and Galbraith in [26]. It consists in adopting a redundant representation of class group elements and performing rejection sampling. The result is a scheme whose proof generation and verification are quite inefficient.

The class number computation for the CSIDH-512 set of parameters has allowed Beullens, Lange and Vercauteren to produce a much better fix. Namely, $(e_1 - r_1, \ldots, e_n - r_n)$ is replaced by its canonical representative in $\mathbb{Z}/N\mathbb{Z}$, that we denote by $\mathsf{rsp}$. The reception of $\mathsf{rsp}$, however, constitutes a problem for the verifier, since they need to compute the action $[\mathfrak{g}^{\mathsf{rsp}}] \star [E_{\mathsf{com}}]$ which, in general, has exponential complexity. In order to obtain an equivalent representation $(f_1, \ldots, f_n)$ of $\mathsf{rsp}$ in $[-B, B]^n$ or in a slightly bigger set $[-B', B']^n$ (i.e. $[\mathfrak{g}^{\mathsf{rsp}}] = \prod_{i=1}^{n}[\mathfrak{I}_i]^{f_i}$), a lattice-based solution is applied. The resulting signature scheme, CSI-FiSh, enjoys practical efficiency in both signature generation and verification, while maintaining the short signature size offered by SeaSign. It should be stressed, however, that CSI-FiSh is tailored to the CSIDH-512 set of parameters, and that generalizing it to other sets of parameters for which the prime $p$ is bigger than that of CSIDH-512 appears to be out of reach. A tight security variant retaining almost the same efficiency of CSI-FiSh is proposed in [37].

### 6.1. *Efficient smooth representation of* $\mathfrak{g}^{\mathsf{rsp}}$

As already mentioned, the practical efficiency of CSI-FiSh is granted by the possibility of representing $[\mathfrak{g}^{\mathsf{rsp}}]$ as $\prod_{i=1}^{n}[\mathfrak{I}_i]^{f_i}$, for some $(f_1, \ldots, f_n) \in [-B', B']^n$, in an efficient way. Since CSI-FiSh is specific for the CSIDH-512 set of parameters, in the following, we will use the concrete values 74 and 5 for $n$ and $B$, respectively.

We observe that the cost for computing the action corresponding to a vector $(f_1, \ldots, f_{74}) \in \mathbb{Z}^{74}$ highly depends on the number of atomic isogenies to be computed, i.e. on the $L_1$-norm $\sum_{i=1}^{74} |f_i|$. Hence, in order to find a representative vector for $\mathsf{rsp}$ that leads to an efficient class-group-action evaluation, a strategy (possibly not the optimal one) is to compute a vector close to $(\mathsf{rsp}, 0, \ldots, 0)$ in the lattice

$$\Lambda = \left\{ (z_1, \ldots, z_{74}) \in \mathbb{Z}^{74} \;\middle|\; \prod_{i=1}^{n} [\mathfrak{I}_i]^{z_i} = [(1)] \right\}$$

with respect to the $L_1$-norm. We remark that the knowledge of a reduced basis for the lattice $\Lambda$ (obtained by computing a Hermite normal form for it) is a side-product of the class number computation executed in [8].

A lattice vector $\underline{z}$ that is close to $(\mathsf{rsp}, 0, \ldots, 0)$ can be first obtained by applying the Babai's Nearest Plane algorithm [2]. In order to find a closer lattice vector, Beullens *et al.* [8] suggest the use of a second algorithm [29, 43]. Namely, on input the vector $\underline{f}' = (\mathsf{rsp}, 0, \ldots, 0) - \underline{z}$ together with a list of 10,000 short vectors of $\Lambda$, the algorithm outputs a lattice vector $\underline{z}'$ that is close to $\underline{f}'$. Thus, $\underline{z} + \underline{z}'$ is a lattice vector close to $(\mathsf{rsp}, 0, \ldots, 0)$, since $(\mathsf{rsp}, 0, \ldots, 0) - (\underline{z} + \underline{z}') = f' - \underline{z}'$. Such vector $\underline{z} + \underline{z}'$ is expected to be closer to $(\mathsf{rsp}, 0, \ldots, 0)$ than $\underline{z}$.

## 7. Conclusions

The isogeny-based protocol CSIDH exploits the action of the ideal class group of a quadratic order on a set of supersingular elliptic curves for the exchange of cryptographic keys. CSIDH enjoys short keys and ciphertexts, requires small bandwidth and has a fairly good running time. However, the scheme is still far from being competitive with other post-quantum cryptosystems (for example those based on lattices) in terms of efficiency, mainly due to the high computational cost of the class group action.

Since the proposal of the CSIDH scheme in 2018, several papers focusing on improving the computation of the class group action and making it independent of the secret key have appeared. Furthermore, a record class group computation led to the first practical isogeny-based signature scheme. In this paper, we reviewed the mathematical and algorithmic aspects of some of these contributions in a unified dissertation.

The above-mentioned advancements have led to slightly more efficient implementations of CSIDH and some secret-key-independent variants. In addition, since the writing of this survey some other relevant papers on the topic have also appeared (e.g. [10, 25]). However, none of them have determined a significant break-through. This might be due to the fact that all have as a backbone the original CSIDH proof-of-concept implementation. We speculate that a major contribution would need a change of paradigm, likely coming from number-theoretic results. We hope that this work would contribute to trigger further research on the topic.

## Acknowledgments

The authors are grateful to the anonymous reviewers for their useful comments which helped improve the paper.

## ORCID

Marzio Mula https://orcid.org/0000-0002-5953-8724
Federico Pintore https://orcid.org/0000-0002-7985-3131

## References

[1] G. Adj, J.-J. Chi-Domínguez and F. Rodríguez-Henríquez, Karatsuba-based square-root Vélu's formulas applied to two isogeny-based protocols, *J. Cryptogr. Eng.* **13**(4) (2022) 1–18.

[2] L. Babai, On Lovász' lattice reduction and the nearest lattice point problem, *Combinatorica* **6**(1) (1986) 1–13.

[3] G. L. D. Banegas, D. J. Bernstein, F. Campos, T. Chou, T. Lange, M. Meyer, B. Smith and J. Sotáková, CTIDH: Faster constant-time CSIDH, *IACR Trans. Cryptogr. Hardw. Embed. Syst.* (4) (2021) 351–387.

[4] D. J. Bernstein, P. Birkner, M. Joye, T. Lange and C. Peters, Twisted Edwards curves, in *Progress in Cryptology — AFRICACRYPT 2008* (Springer Berlin, Heidelberg, 2008), pp. 389–405.

[5] D. J. Bernstein, L. De Feo, A. Leroux and B. Smith, Faster computation of isogenies of large prime degree, in *ANTS-XIV — 14th Algorithmic Number Theory Symposium* (Mathematical Sciences Publishers, 2020), pp. 39–55.

[6] D. J. Bernstein, M. Hamburg, A. Krasnova and T. Lange, Elligator: Elliptic-curve points indistinguishable from uniform random strings, in *Proc. 2013 ACM SIGSAC Conf. Computer & Communications Security* (ACM, 2013), pp. 967–980.

[7] D. Bernstein, T. Lange, C. Martindale and L. Panny, Quantum circuits for the CSIDH: optimizing quantum evaluation of isogenies, in *Advances in Cryptology — EUROCRYPT 2019* (Springer, Cham, 2019), pp. 409–441.

[8] W. Beullens, T. Lange and F. Vercauteren, CSI-FiSh: Efficient isogeny based signatures through class group computations, in *Int. Conf. Theory and Application of Cryptology and Information Security* (Springer, 2019), pp. 227–247.

[9] J.-F. Biasse, C. Fieker and M. J. Jacobson, Fast heuristic algorithms for computing relations in the class group of a quadratic order, with applications to isogeny evaluation, *LMS J. Comput. Math.* **19**(A) (2016) 371–390.

[10] D. Boneh, J. Guan and M. Zhandry, A lower bound on the length of signatures based on group actions and generic isogenies, in *Advances in Cryptology — EUROCRYPT 2023*, eds. C. Hazay and M. Stam (Springer, Switzerland, 2023), pp. 507–531.

[11] J. Buchmann and U. Vollmer, *Binary Quadratic Forms*: *An Algorithmic Approach*, Algorithms and Computation in Mathematics, Vol. 20 (Springer, 2007).

[12] W. Castryck and T. Decru, CSIDH on the surface, in *Int. Conf. Post-Quantum Cryptography* (Springer, 2020), pp. 111–129.

[13] W. Castryck, T. Decru, M. Houben and F. Vercauteren, Horizontal racewalking using radical isogenies, in Advances in Cryptology — ASIACRYPT 2022, eds. S. Agrawal and D. Lin (Springer, Switzerland, 2022), pp. 67–96.

[14] W. Castryck, T. Decru and F. Vercauteren, Radical isogenies, in *Int. Conf, Theory and Application of Cryptology and Information Security* (Springer, 2020), pp. 493–519.

[15] W. Castryck, T. Lange, C. Martindale, L. Panny and J. Renes, CSIDH: An efficient post-quantum commutative group action, in *Int. Conf. Theory and Application of Cryptology and Information Security* (Springer, 2018), pp. 395–427.

[16] W. Castryck, L. Panny and F. Vercauteren, Rational isogenies from irrational endomorphisms, in *Advances in Cryptology–EUROCRYPT 2020*: *39th Annual Int. Conf. Theory and Applications of Cryptographic Techniques*, *Part II* (Springer, 2020), pp. 523–548.

[17] D. Cervantes-Vázquez, M. Chenu, J.-J. Chi-Domínguez, L. De Feo, F. Rodríguez-Henríquez and B. Smith, Stronger and faster side-channel protections for CSIDH, in

*Int. Conf. Cryptology and Information Security in Latin America* (Springer, 2019), pp. 173–193.

[18] J. Chávez-Saab, J.-J. Chi-Domínguez, S. Jaques and F. Rodríguez-Henríquez, The SQALE of CSIDH: Sublinear Vélu quantum-resistant isogeny action with low exponents, *J. Cryptogr. Eng.* **11**(4) (2021) 307–319.

[19] J.-J. Chi-Domínguez and F. Rodríguez-Henríquez, Optimal strategies for CSIDH, *Adv. Math. Commun.* **16**(2) (2022) 383–411.

[20] A. Childs, D. Jao and V. Soukharev, Constructing elliptic curve isogenies in quantum subexponential time, *J. Math. Cryptol.* **8**(1) (2014) 1–29.

[21] C. Costello and H. Hisil, A simple and compact algorithm for SIDH with arbitrary degree isogenies, in *Int. Conf. Theory and Application of Cryptology and Information Security* (Springer, 2017), pp. 303–329.

[22] C. Costello and B. Smith, Montgomery curves and their arithmetic: The case of large characteristic fields, *J. Cryptogr. Eng.* **8**(1) (2018) 27–41.

[23] J.-M. Couveignes, Hard homogeneous spaces, Report 2006/291, Cryptology ePrint Archive (2006), https://ia.cr/2006/291.

[24] D. A. Cox, *Primes of the Form* $x^2 + ny^2$ *: Fermat, Class Field Theory, and Complex Multiplication* (John Wiley & Sons, 1997).

[25] L. De Feo, T. B. Fouotsa, P. Kutas, A. Leroux, S.-P. Merz, L. Panny and B. Wesolowski, SCALLOP: Scaling the CSI-FiSh, in *Public-Key Cryptography — PKC 2023*, eds. A. Boldyreva and V. Kolesnikov (Springer, Switzerland, 2023), pp. 345–375.

[26] L. De Feo and S. D. Galbraith, SeaSign: Compact isogeny signatures from class group actions, in *Annual Int. Conf. Theory and Applications of Cryptographic Techniques* (Springer, 2019), pp. 759–789.

[27] L. De Feo, J. Kieffer and B. Smith, Towards practical key exchange from ordinary isogeny graphs, *Advances in Cryptology–ASIACRYPT 2018*: *24th Int. Conf. Theory and Application of Cryptology and Information Security* (Springer, 2018), pp. 365–394.

[28] C. Delfs and S. D. Galbraith, Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$, *Des. Codes Cryptogr.* **78**(2) (2016) 425–440.

[29] E. Doulgerakis, T. Laarhoven and B. de Weger, Finding closest lattice vectors using approximate voronoi cells, in *Int. Conf. Post-Quantum Cryptography* (Springer, 2019), pp. 3–22.

[30] A. Enge, *Elliptic Curves and Their Applications to Cryptography*: *An Introduction* (Kluwer Academic Publishers, 1999).

[31] A. Fiat and A. Shamir, How to prove yourself: Practical solutions to identification and signature problems, in *Conf. Theory and Application of Cryptographic Techniques* (Springer, 1986), pp. 186–194.

[32] S. D. Galbraith, *Mathematics of Public Key Cryptography* (Cambridge University Press, 2014).

[33] D. Harvey and J. van der Hoeven, Polynomial multiplication over finite fields in time $o(n \log n)$, *J. ACM* **69**(2) (2022) 1–39.

[34] D. Husemöller, *Elliptic Curves* (Springer, New York, 1987).

[35] A. Hutchinson, J. LeGrow, B. Koziel and R. Azarderakhsh, Further optimizations of CSIDH: a systematic approach to efficient strategies, permutations, and bound vectors, in *Int. Conf. Applied Cryptography and Network Security* (Springer, 2020), pp. 481–501.

[36] D. Jao and L. De Feo, Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies, in *Int. Workshop on Post-Quantum Cryptography* (Springer, 2011), pp. 19–34.

[37] A. E. Kaafarani, S. Katsumata and F. Pintore, Lossy csi-fish: Efficient signature scheme with tight reduction to decisional CSIDH-512, in *IACR Int. Conf. Public-Key Cryptography* (Springer, 2020), pp. 157–186.

[38] A. Karatsuba, The complexity of computations, *Proc. Steklov Inst. Math.* **211** (1995) 169–183.

[39] K. Kodera, C.-M. Cheng and A. Miyaji, Efficient algorithm for computing odd-degree isogenies on Montgomery curves, in *Information Security Applications* (Springer International Publishing, 2020), pp. 258–275.

[40] D. Kohel, Endomorphism rings of elliptic curves over finite fields, Ph.D. thesis, University of California at Berkeley (1996).

[41] G. Kuperberg, Another subexponential-time quantum algorithm for the dihedral hidden subgroup problem, in *Theory of Quantum Computation, Communication, and Cryptography* (Springer Berlin, Heidelberg, 2005), pp. 20–34.

[42] G. Kuperberg, A subexponential-time quantum algorithm for the dihedral hidden subgroup problem, *SIAM J. Comput.* **35**(1) (2005) 170–188.

[43] T. Laarhoven, Sieving for closest lattice vectors (with preprocessing), in *Int. Conf. Selected Areas in Cryptography*, Vol. 10532 (Springer, 2016), pp. 523–542.

[44] D. Liu, T. Song and Y. Dai, Isomorphism and generation of montgomery-form elliptic curves suitable for cryptosystems, *Tsinghua Sci. Technol.* **10**(2) (2005) 145–151.

[45] L. Maino and F. Pintore, Mathematical and computational aspects of CSIDH-based algorithms, in *Algebra for Cryptography*, Collectio Ciphrarum, Vol. 1, eds. R. Aragona, N. Gavioli and F. Mignosi (Aracne, 2021), pp. 55–62.

[46] M. Meyer, F. Campos and S. Reith, On lions and elligators: An efficient constant-time implementation of CSIDH, in *Int. Conf. Post-Quantum Cryptography*, Vol. 11505 (Springer, 2019), pp. 307–325.

[47] M. Meyer and S. Reith, A faster way to the CSIDH, in *Progress in Cryptology — INDOCRYPT 2018*, eds. D. Chakraborty and T. Iwata (Springer International Publishing, 2018), pp. 137–152.

[48] J. S. Milne, *Algebraic Number Theory* (2020), p. 166, http://www.jmilne.org/math/"www.jmilne.org/math/.

[49] H. Montgomery and M. Zhandry, Full quantum equivalence of group action DLog and CDH, and more, *Advances in Cryptology–ASIACRYPT 2022: 28th Int. Conf. Theory and Application of Cryptology and Information Security, Part I* (Springer, 2023), pp. 3–32.

[50] D. Moody and D. Shumow, Analogues of Vélu's Formulas for isogenies on alternate models of elliptic curves, *Math. Comp.* **85**(300) (2016) 1929–1951.

[51] K. Nakagawa, H. Onuki, A. Takayasu and T. Takagi, L1-norm ball for CSIDH: Optimal strategy for choosing the secret key space, *Discrete Appl. Math.* **328** (2023) 70–88.

[52] K. Okeya, H. Kurumatani and K. Sakurai, Elliptic curves with the montgomery-form and their cryptographic applications, in *Public Key Cryptography* (Springer Berlin, Heidelberg, 2000), pp. 238–257.

[53] H. Onuki, Y. Aikawa, T. Yamazaki and T. Takagi, A faster constant-time algorithm of CSIDH keeping two points, in *Int. Workshop on Security* (Springer, Cham, 2019), pp. 23–33.

[54] H. Onuki and T. Takagi, On collisions related to an ideal class of order 3 in CSIDH, in *Int. Workshop on Security* (Springer, Cham, 2020), pp. 131–148.

[55] C. Peikert, He gives C-sieves on the CSIDH, in *Annual Int. Conf. Theory and Applications of Cryptographic Techniques* (Springer, Cham, 2020), pp. 463–492.

[56] A. Rostovtsev and A. Stolbunov, Public-key cryptosystem based on isogenies, Tech. Rep. Report 2006/145, Cryptology ePrint Archive (2006).

[57] P. W. Shor, Algorithms for quantum computation: Discrete logarithms and factoring, in *Proc. 35th Annual Symp. Foundations of Computer Science* (IEEE, 1994), pp. 124–134.

[58] J. Silverman, *The Arithmetic of Elliptic Curves*, Vol. 106 (Springer Science & Business Media, 2009).

[59] A. Stolbunov, Cryptographic Schemes Based on Isogenies, Ph.D. Thesis, Norwegian University of Science and Technology (2012).

[60] A. V. Sutherland, Isogeny volcanoes, in *Algorithmic Number Theory 10th Int. Symp.*, Vol. 1 (MSP, 2013), pp. 507–530.

[61] A. V. Sutherland, Finite field arithmetic (2022), https://math.mit.edu/classes/18.783/2022/LectureNotes3.pdf.

[62] J. Tate, Endomorphisms of abelian varieties over finite fields, *Invent. Math.* **2** (1966) 134–144.

[63] J. von zur Gathen and J. Gerhard, *Modern Computer Algebra*, 3rd edn. (Cambridge University Press, 2013).

[64] L. C. Washington, *Elliptic Curves: Number Theory and Cryptography*, 2nd edn. (Chapman and Hall/CRC, 2008).

[65] W. C. Waterhouse, Abelian varieties over finite fields, *Ann. Sci. Éc. Norm. Supér.* **2**(4) (1969) 521–560.