The supersingular endomorphism ring problem given one endomorphism

Arthur Herlédan Le Merdy, Benjamin Wesolowski

The supersingular endomorphism ring problem given one endomorphism

Arthur Herlédan Le Merdy[1,2] and Benjamin Wesolowski[1]

[1] ENS de Lyon, CNRS, UMPA, UMR 5669, Lyon France
[2] ENS de Lyon, LIP (CNRS, U. Lyon, ENS de Lyon, Inria, UCBL), UMR 5668, Lyon France

Abstract. Given a supersingular elliptic curve E and a non-scalar endomorphism α of E, we prove that the endomorphism ring of E can be computed in classical time about $|\operatorname{disc}(\mathbb{Z}[\alpha])|^{1/4}$, and in quantum subexponential time, assuming the generalised Riemann hypothesis. Previous results either had higher complexities, or relied on heuristic assumptions.
Along the way, we describe an analyse a general algorithm to divide isogenies in polynomial time, and to solve the Primitivisation problem in polynomial time. Following the attacks on SIDH, isogenies in high dimension are a central ingredient of our results.

Keywords: Isogeny-based cryptography · Endomorphism ring · Supersingular elliptic curve · Orientation · Class group · Cryptanalysis

1 Introduction

Isogeny-based cryptography is an active and promising branch of post-quantum cryptography. Isogenies are certain kinds of maps between elliptic curves. The security of cryptosystems in this family relies mainly on the algorithmic hardness of constructing an isogeny between two supersingular elliptic curves: the *supersingular isogeny path problem*.

Endomorphisms of an elliptic curve E are isogenies from E to itself, and their collection forms the endomorphism ring $\mathrm{End}(E)$. The *endomorphism ring problem*, denoted EndRing, consists in computing the endomorphism ring of a supersingular elliptic curve. Under the generalised Riemann hypothesis, the isogeny problem is equivalent to EndRing [EHL+18,Wes22b]. This equivalence has placed EndRing at the heart of isogeny-based cryptography, and its hardness has been proved to relate to the security of the `CGL` hash function [CLG09,EHL+18], the `CSIDH` key exchange protocol [CD20,CPV20,Wes22a] and the `SQISign` signature scheme [DFKL+20].

In certain cryptosystems, the elliptic curves involved are equipped with one public endomorphism. For instance, in CSIDH [CD20], all elliptic curves are defined over $\mathbb{F}_p$, and thus the Frobenius endomorphism is an accessible non-trivial endomorphism. The situation is similar in [CS22,FFK+23]. The endomorphism ring problem then asks to find all the *other* endomorphisms. This yields the following question:

- How much does knowing one endomorphism simplify the computation of the endomorphism ring of a supersingular elliptic curve?

A closely related question was studied in [ACL+23]: given two curves E and E', together with two endomorphisms $\alpha \in \mathrm{End}(E)$ and $\beta \in \mathrm{End}(E')$, how hard is it to find an isogeny between them? Under several heuristic assumptions, they provide a classical exponential algorithm and a quantum subexponential algorithm solving this problem. With the equivalence between the isogeny path problem and EndRing, their work provides a first answer

Date of this document: 2025-02-10.

to the above question. Yet that answer has limitations: first, as stated, it is only heuristic. Second, the output of the algorithm of [ACL+23] may have exponential size, which could considerably increase the cost of applying the equivalence.

The schemes of [CD20, CS22, FFK+23] have in common the notion of *orientation*, introduced by Colò and Kohel [CK20]. Given an order $\mathfrak{O}$ in a quadratic number field, an $\mathfrak{O}$-orientation of a curve E is a subring of $\mathrm{End}(E)$ isomorphic to $\mathfrak{O}$. The interest in this notion lies in the fact that the set of $\mathfrak{O}$-oriented curves comes with an action of the class group of $\mathfrak{O}$. The problem of inverting this group action is known as the Vectorisation problem. The presumed hardness of this problem was already at the heart of the security of the CRS protocol [Cou06, RS06] where the action of ideal class groups came from the complex multiplication theory of ordinary elliptic curves. Today, it is behind the security of CSIDH [CD20] and its variants [CS22, FFK+23].

Any endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$ gives rise to a $\mathbb{Z}[\alpha]$-orientation, hinting at the connection between the EndRing problem given one endomorphism, and problems involving orientations. The link between Vectorisation and EndRing has first been studied in the particular case of CSIDH in [CPV20]. That article proves that there is a subexponential-time reduction from breaking CSIDH to computing the endomorphism rings. Then it has been improved and extended to a polynomial-time equivalence between Vectorisation and EndRing in [Wes22a]. However, these results necessitate the orientations to be *primitive*: the quadratic suborder must be maximal in the endomorphism ring. Obtaining a primitive orientation from a given orientation is not trivial — this problem is called the Primitivisation problem. This leads us to this second question:

- How hard is it to get a primitive orientation from an orientation?

The Primitivisation problem was first introduced in [ACL+23] as a presumably hard problem, and they gave a quantum subexponential algorithm solving it.

1.1 Orientations and variants of EndRing

We now give an informal overview of orientations and related hard problems. For formal definitions, we refer the reader to Section 2 about orientations and general notations, to Section 3 about the different hard problems and to [Sil86] for a detailed reference about elliptic curves and isogenies.

We fix a prime integer p and we denote E a supersingular elliptic curve defined over $\bar{\mathbb{F}}_p$. An isogeny of elliptic curves is a morphism between elliptic curves seen as abelian varieties. We denote by $\mathrm{End}(E)$ the ring formed by isogenies from E to itself, i.e. the endomorphisms of E. We consider the following supposedly hard problem EndRing.

- EndRing: Given a supersingular elliptic curve E, compute $\mathrm{End}(E)$.

The current best classical algorithms to solve EndRing run in expected time $\tilde{O}(p^{1/2})$, see for instance [EHL+20], and the best quantum algorithms have complexity in $\tilde{O}(p^{1/4})$, see for example [BJS14].

Let $\mathfrak{O}$ be an order of a quadratic number field. An orientation ι is an embedding from $\mathfrak{O}$ into $\mathrm{End}(E)$. This is mainly equivalent to knowing an endomorphism in $\mathrm{End}(E)\setminus\mathbb{Z}$. If this embedding cannot be extended to any superorder of $\mathfrak{O}$, we say that ι is a primitive orientation. When a (primitive) orientation ι exists, we say that E is (primitively) $\mathfrak{O}$-orientable and that the pair (E, ι) is a (primitively) $\mathfrak{O}$-oriented elliptic curve.

This notion of orientation comes together with variants of EndRing where partial information on the endomorphism ring is given. Let α be an element of the quadratic order $\mathfrak{O}$.

- α-EndRing: Given a supersingular elliptic curve E together with an orientation $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$, compute $\mathrm{End}(E)$.

- $\mathfrak{O}$-EndRing: Given a primitively $\mathfrak{O}$-oriented supersingular elliptic curve (E, ι), compute $\mathrm{End}(E)$.

These two problems are tightly related. On the one hand, there is a direct reduction from $\mathfrak{O}$-EndRing to α-EndRing as the inputs of the former are also inputs of the latter. On the other hand, the reduction from α-EndRing to $\mathfrak{O}$-EndRing is not trivial as it requires to compute a primitive orientation from any given orientation. This computation has been introduced in [ACL+23] as a hard problem together with a quantum algorithm for solving it in subexponential time under some heuristics.

- Primitivisation: Given a supersingular elliptic curve E together with an orientation $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$, find the primitive orientation $\iota' : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$ such that $\mathbb{Z}[\alpha] \subseteq \mathfrak{O}$.

To get a classical reduction from α-EndRing to $\mathfrak{O}$-EndRing, the most direct approach consists in extending the given orientation to a primitive one by solving the Primitivisation problem.

The $\mathfrak{O}$-EndRing problem is not only interesting to investigate the complexity of EndRing given some additional information, it also has an important place in isogeny-based cryptography. To see that, we first need to consider the Vectorisation problem induced by primitive orientations. From a primitive orientation, one can construct a free action of the class group $\mathrm{Cl}(\mathfrak{O})$ over the set of primitively $\mathfrak{O}$-oriented elliptic curves. We denote this group action as

$$\begin{aligned} \mathrm{Cl}(\mathfrak{O}) \times SS_{\mathfrak{O}}(p) &\to SS_{\mathfrak{O}}(p) \\ ([\mathfrak{a}], (E, \iota)) &\mapsto \mathfrak{a} \star (E, \iota) := (E^{\mathfrak{a}}, \iota^{\mathfrak{a}}), \end{aligned}$$

where $SS_{\mathfrak{O}}(p)$ is the set of primitively $\mathfrak{O}$-oriented supersingular elliptic curves defined over $\bar{\mathbb{F}}_p$ up to isomorphism. This group action allows one to define a Vectorisation problem, giving a framework to study security of CSIDH-like protocols.

- $\mathfrak{O}$-Vectorisation: Given (E, ι) and (E', ι') in $SS_{\mathfrak{O}}(p)$ find an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $E^{\mathfrak{a}} \simeq E'$.

Under the generalised Riemann hypothesis and given the factorisation of $\mathrm{disc}(\mathfrak{O})$, the $\mathfrak{O}$-EndRing problem is equivalent to $\mathfrak{O}$-Vectorisation in probabilistic polynomial time, see [Wes22a]. Therefore, the security of many protocols such as CSIDH [CD20], CSI-FiSh [BKV19] and CSURF [CD20] reduces to $\mathfrak{O}$-EndRing see [Wes22a].

In the current state of the art, using l to denote the length of the input, the problem of $\mathfrak{O}$-Vectorisation can heuristically be solved in expected classical time $l^{O(1)}|\mathrm{disc}(\mathfrak{O})|^{1/4}$, using for instance approaches close to the ones in [DG16]. Quantumly, it can heuristically be solved in time $l^{O(1)}L_{|\mathrm{disc}(\mathfrak{O})|}[1/2]$, see [Wes22a, Proposition 4].

1.2 Contributions

We provide algorithms whose asymptotic complexity matches or improves upon previous results in the literature. Unlike previous results, our proofs do not rely on heuristic assumptions. In this list of contributions, we suppose that the input and output of the algorithms are always in efficient representation, we refer the reader to Section 2.3 for more information about representations and encodings.

- In Section 4, we develop the first ingredient for the rest of the paper: an algorithm to divide isogenies, Corollary 1. Explicitly, given two isogenies φ and η, the algorithm returns the unique isogeny ψ such that $\varphi = \psi \circ \eta$ (or asserts that such ψ does not exist). This is the right-division of φ by η (dualizing, one can also divide on the left).

  The heart of the method is an algorithm to divide isogenies by integers in polynomial time. It is a generalization of a division algorithm for translates of the Frobenius introduced by Robert to compute the endomorphism ring of an ordinary elliptic curve, [Rob22b, Theorem 4.2]. Before the attacks of SIDH, dividing isogenies by integers either required superpolynomial time, or degraded the quality of the representation (getting exponentially worse with each application). It is not the case here.

- In Section 5, we use this division algorithm to solve the Primitivisation problem. This result adapts Robert's algorithm for computing in polynomial time the endomorphism ring of ordinary elliptic curves [Rob22b], which can be seen as an ordinary counterpart of the Primitivisation problem. We prove that, when the factorisation of $\mathrm{disc}(\mathfrak{O})$ is known, there is a classical polynomial time algorithm solving Primitivisation.

  As an application, we provide a polynomial time algorithm for computing the action of smooth ideals. Previous polynomial-time algorithms for this task required the norm of the input ideal to be powersmooth.

We now use l to denote the length of the input, and use the standard L-notation for subexponential complexities (Definition 3).

- In Section 6, under the generalised Riemann hypothesis, we provide
  - a classical algorithm solving $\mathfrak{O}$-Vectorisation in expected time $l^{O(1)}|\mathrm{disc}(\mathfrak{O})|^{1/4}$.
  - a quantum algorithm solving $\mathfrak{O}$-Vectorisation in expected time $l^{O(1)}L_{|\mathrm{disc}(\mathfrak{O})|}[1/2]$.

  This directly leads to
  - a classical algorithm solving $\mathfrak{O}$-EndRing in expected time $l^{O(1)}|\mathrm{disc}(\mathfrak{O})|^{1/4}$.
  - a quantum algorithm solving $\mathfrak{O}$-EndRing in expected time $l^{O(1)}L_{|\mathrm{disc}(\mathfrak{O})|}[1/2]$.

  Combined with our resolution of Primitivisation, we obtain the following theorems on solving the endomorphism ring problem knowing an endomorphism, rigorously.

  Theorem 1 (**GRH**). *There is a classical algorithm that given a supersingular curve E, and an endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$, computes the endomorphism ring of E in expected time $l^{O(1)}|\mathrm{disc}(\mathbb{Z}[\alpha])|^{1/4}$ where l is the length of the input.*

  Theorem 2 (**GRH**). *There is a quantum algorithm that given a supersingular curve E, and an endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$, computes the endomorphism ring of E in expected time $l^{O(1)}L_{|\mathrm{disc}(\mathbb{Z}[\alpha])|}[1/2]$ where l is the length of the input.*

- In section 7, we detail how the algorithmic improvements of Section 4 allow one to navigate efficiently in the volcano of oriented isogenies. In the previous literature, the number of steps that one could efficiently take in a volcano was limited because of the degrading quality of representations.

  As a direct application, we present an optimisation of the resolution of $\mathfrak{O}$-EndRing through the following reduction:
  - Under the generalised Riemann hypothesis, there is a probabilistic reduction from $(\mathbb{Z}+c\mathfrak{O})$-EndRing to $\mathfrak{O}$-EndRing taking a time polynomial in the length of the input and in the largest prime factor of c.

This last result improves the probabilistic polynomial reduction given by [Wes22a, Theorem 5] by relaxing the powersmoothness constraint on c. It also leads to a classical algorithm solving $(\mathbb{Z} + c\mathfrak{O})$-EndRing in expected time polynomial in the length of the input, in $\mathrm{disc}(\mathfrak{O})$ and in the largest prime factor of c. This improves and removes the heuristics of [Wes22a, Corollary 6.].

Acknowledgements

The authors would like to extend their gratitude to Guillaume Hanrot for helpful discussions and feedback which have significantly contributed to the writing of this paper. They also wish to thank Damien Robert for his invaluable answers to our questions. The authors were supported by the CHARM ANR-NSF grant (ANR-21-CE94-0003), by the HQI initiative (ANR-22-PNCQ-0002) and by the PEPR quantique France 2030 programme (ANR-22-PETQ-0008).

2 Definitions and notations

In this article, some results will be proved assuming the generalised Riemann hypothesis. They will be marked by **GRH**, see for instance Theorem 1 and 2.

We denote by $\mathbb{F}_q$ the finite field with q elements and by $\bar{\mathbb{F}}_q$ its algebraic closure. The cardinality of a set S is denoted by $\#S$. For any order $\mathcal{O}$, $\mathrm{disc}(\mathcal{O})$ is the notation of its discriminant. We use the standard O-notation together with the $\tilde{O}$-notation which removes the logarithmic factors of the O-notation, i.e. $O(f(x)\log^k(x)) = \tilde{O}(f(x))$ for any positive integer k. When a time complexity is given without specifying the type of operations, it is assumed to count binary operations.

In the algorithms presented here, we use $x \leftarrow \mathrm{Unif}\{y \in S\}$ to denote that x is sampled uniformly at random among the elements of S.

Definition 1 ((Power)Smoothness bound)**.** Let n be an integer of prime decomposition $\ell_1^{e_1} \dots \ell_r^{e_r}$. We say that an integer B is a **smoothness bound** on n and n is said to be ***B*-smooth** if

$$B \geq \max_{i \in [\![1,r]\!]} \ell_i;$$

if further

$$B \geq \max_{i \in [\![1,r]\!]} \ell_i^{e_i}$$

then B is a **powersmoothness bound** on n and n is ***B*-powersmooth**. We denote by $P^+(n)$ the integer $\max_{i \in [\![1,r]\!]} \ell_i$ and by $P^*(n)$ the integer $\max_{i \in [\![1,r]\!]} \ell_i^{e_i}$.

Definition 2 (Extension degree)**.** For any elliptic curve E defined over a finite field $\mathbb{F}_{p^k}$ and integer n of prime decomposition $\ell_1^{e_1} \dots \ell_r^{e_r}$, we use the following notations

- $\delta_E(n) := \max_{i \in [\![1,r]\!]} [\mathbb{F}_{p^k}(E[\ell_i^{e_i}]) : \mathbb{F}_{p^k}]$,
- $\delta_{E,2}(n) := \max_{(i,j) \in [\![1,r]\!]^2, i \neq j} [\mathbb{F}_{p^k}(E[\ell_i^{e_i}\ell_j^{e_j}]) : \mathbb{F}_{p^k}]$,

where, for any integer m, $\mathbb{F}_{p^k}(E[m])$ stands for the smallest field extension of $\mathbb{F}_{p^k}$ where the coordinates of the points of $E[m]$ live.

Definition 3 (L-notation)**.** Let a, b, x be three real numbers. To handle subexponential complexities, we define the following standard ***L*-notation**

$$L_x[a,b] := \exp(b(\log x)^a (\log\log x)^{(1-a)}),$$

as well as this L-notation for unknown constants

$$L_x[a] := \exp(O((\log x)^a (\log\log x)^{(1-a)})).$$

2.1 Cayley graph

Definition 4 (Cayley graph)**.** Let G be a finite group and let $S \subseteq G$ be a generating subset of G. The **Cayley graph Cay (G, S)** is the graph whose vertices are the elements of G and such that there exists an edge between two vertices g_1, g_2 if and only there exists an $s \in S$ such that $g_2 = sg_1$.

We shall use the following result of Childs, Jao and Soukkharev regarding random walks over Cayley graphs of class groups.

Proposition 1 (**(GRH**,) Theorem 2.1 in [CJS14])**.** *Let $\mathfrak{O}$ be an imaginary quadratic order of discriminant Δ and conductor $f_{\mathfrak{O}}$. Let $\varepsilon > 0$ and x be a real number such that $x \geq (\log|\Delta|)^{2+\varepsilon}$. Let Σ_x be the set*

$$\{[\mathfrak{p}] \in \mathrm{Cl}(\mathfrak{O}) \text{ such that } \gcd(f_{\mathfrak{O}}, \mathfrak{p}) = 1 \text{ and } N(\mathfrak{p}) \leq x \text{ prime}\}$$

from which we define the set S_x to be

$$\Sigma_x \cup \{[\mathfrak{p}]^{-1} \text{ for } [\mathfrak{p}] \in \Sigma_x\}.$$

Then there exists a positive constant $C > 1$, depending only on ε, such that for all Δ sufficiently large, a random walk of length

$$t \geq C \frac{\log \#\mathrm{Cl}(\mathfrak{O})}{\log\log|\Delta|}$$

in the Cayley graph $\mathrm{Cay}(\mathrm{Cl}(\mathfrak{O}), S_x)$ from any starting vertex lands in any fixed subset $H \subset \mathrm{Cl}(\mathfrak{O})$ with probability P such that

$$\frac{1}{2}\frac{\#H}{\#\mathrm{Cl}(\mathfrak{O})} \leq P.$$

2.2 Elliptic curves and orientations

In this section, we recall some basic definitions and notations about elliptic curves before introducing the recent notions of orientations [CK20]. For more details about elliptic curves theory, we refer the reader to Silverman's book [Sil86].

An **elliptic curve** is an abelian variety of dimension 1. **Isogenies** of elliptic curves are non-trivial homomorphisms between them. An isogeny from an elliptic curve to itself is called an **endomorphism**. The set of all endomorphisms of an elliptic curve E together with the trivial map form the **endomorphism ring** $(\mathrm{End}(E), +, \circ)$ where $+$ is the point-wise addition and $\circ$ is the composition of maps. For any integer n and elliptic curve E, we denote by $[n]$ the **multiplication-by-n map** on E and by $E[n]$ its kernel, called the **n-torsion subgroup** of E. An elliptic curve E defined over a finite field of characteristic p is said to be **supersingular** if $E[p] \simeq \{0\}$.

In this paper, we only work with supersingular elliptic curves defined over a field of characteristic p, where p is a fixed prime number. The p^k-Frobenius isogeny from a curve E is the isogeny $\phi_E^{p^k} : E \to E^{(p^k)} : (x, y) \mapsto (x^q, y^q)$, where $E^{(p^k)}$ is defined by the equation of E with coefficients raised to the power p^k. An isogeny $\varphi : E \to E'$ is **inseparable** if it factors as $\varphi = \psi \circ \phi_E^p$ for some isogeny ψ. Otherwise, the isogeny is **separable**. Any isogeny can be written as $\varphi = \psi \circ \phi_E^{p^k}$ where ψ is separable, and we define the **degree** of φ as $\deg(\varphi) = p^k \cdot \#(\ker\varphi)$. For any isogeny $\varphi : E \to E'$, there exists a unique isogeny $\hat{\varphi}$, the **dual isogeny** of φ, such that $\hat{\varphi} \circ \varphi = [\deg(\varphi)]$. For any prime $\ell \neq p$, an **ℓ-isogeny** is an isogeny of degree ℓ.

An important property of supersingular elliptic curves is that their endomorphism ring is isomorphic to a maximal order of the quaternion algebra over $\mathbb{Q}$ ramified only at p and at infinity. This quaternion algebra is unique up to isomorphism and we denote it by $B_{p,\infty}$. More explicitly, we have the isomorphism of $\mathbb{Q}$-algebras

$$B_{p,\infty} \simeq \mathbb{Q} + \mathbb{Q}i + \mathbb{Q}j + \mathbb{Q}ij \text{ such that } i^2 = -p, j^2 = -q_p \text{ and } ij = -ji$$

where q_p is a positive integer depending only on p. We refer the reader to [Voi21] for more information about quaternion algebras.

In a way, quaternion algebras can be seen as two imaginary quadratic number fields combined to get a non commutative 4-dimensional $\mathbb{Q}$-algebra. It is in fact possible to embed an infinite number of imaginary quadratic number fields into a given $\mathbb{Q}$-algebra $B_{p,\infty}$. Naturally, one can then study how orders of imaginary quadratic number fields embed into a given endomorphism ring of supersingular elliptic curves. The study of such embeddings in isogeny-based cryptography originates from [CK20], where they are introduced as **orientations**.

Let K be an imaginary quadratic number field such that p does not split in K.

Definition 5 (Orientation)**.** An elliptic curve E is said to be ***K*-orientable** if there exists an embedding $\iota : K \hookrightarrow \mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$. Then the embedding ι is a ***K*-orientation** and the pair (E, ι) is a ***K*-oriented** elliptic curve.

For any order $\mathcal{O}$ of K, we say that ι is an ***O*-orientation** and (E, ι) is an ***O*-oriented** elliptic curve if $\iota(\mathcal{O}) \subseteq \mathrm{End}(E)$. In this case, ι will often be considered as the embedding $\mathcal{O} \hookrightarrow \mathrm{End}(E)$. An $\mathcal{O}$-orientation ι is **primitive** if $\iota(\mathcal{O}) = \mathrm{End}(E) \cap \iota(K)$. In this case, the oriented elliptic curve (E, ι) is said to be **primitively *O*-oriented**.

Definition 6 (Oriented isogeny)**.** Let (E, ι) and (E', ι') be two K-oriented elliptic curves, and let $\varphi : E \to E'$ be an isogeny. We denote by $\varphi_*(\iota)$ the K-orientation induced by φ. Explicitly, this orientation is given by

$$\varphi_*(\iota)(\kappa) = (\varphi \circ \iota(\kappa) \circ \hat{\varphi}) \otimes \frac{1}{\deg(\varphi)}, \forall \kappa \in K.$$

We say that φ is ***K*-oriented** if $\varphi_*(\iota)$, is equal to ι'. In particular, a K-oriented isogeny of degree 1 is called a K-oriented isomorphism.

For any order $\mathfrak{O}$ in K, let $SS_{\mathfrak{O}}(p)$ be the set of primitively $\mathfrak{O}$-oriented supersingular elliptic curves over $\overline{\mathbb{F}}_p$, up to K-oriented isomorphism. By [Onu21, Proposition 3.2], if p does not divide the conductor of $\mathfrak{O}$ then $SS_{\mathfrak{O}}(p)$ is not empty. From now, we assume that this is always the case. One can then define a free action of the class group $\mathrm{Cl}(\mathfrak{O})$ on the set of curves $SS_{\mathfrak{O}}(p)$. This is analogous to the well-known action of $\mathrm{Cl}(\mathfrak{O})$ on the set of ordinary elliptic curves whose endomorphism rings are isomorphic to $\mathfrak{O}$.

Let us describe precisely how $\mathrm{Cl}(\mathfrak{O})$ acts on $SS_{\mathfrak{O}}(p)$.

We consider the action of an invertible $\mathfrak{O}$-ideal $\mathfrak{a}$ prime to p on a primitively $\mathfrak{O}$-oriented elliptic curve $(E, \iota) \in SS_{\mathfrak{O}}(p)$. First, we consider the finite subgroup $E[\mathfrak{a}]$ of E, called the **$\mathfrak{a}$-torsion** of E, given by

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

It induces a separable isogeny $\varphi_{\mathfrak{a}} : E \to E/E[\mathfrak{a}]$ of kernel $E[\mathfrak{a}]$. We call this isogeny $\varphi_{\mathfrak{a}}$ the **$\mathfrak{a}$-multiplication** and its image curve $E/E[\mathfrak{a}]$, also denoted $E^{\mathfrak{a}}$, the **$\mathfrak{a}$-transform**. Then the action of $\mathfrak{a}$ on (E, ι) is the primitively $\mathfrak{O}$-oriented supersingular elliptic curve $(E^{\mathfrak{a}}, (\varphi_{\mathfrak{a}})_*(\iota))$ up to K-isomorphism. By factorisation, we get the whole action of $\mathrm{Cl}(\mathfrak{O})$ on $SS_{\mathfrak{O}}(p)$.

Proposition 2 ([Onu21]). *The class group* $\mathrm{Cl}(\mathfrak{O})$ *acts over* $SS_{\mathfrak{O}}(p)$ *freely and has at most two orbits. We denote this action as*

$$\begin{aligned}\mathrm{Cl}(\mathfrak{O}) \times SS_{\mathfrak{O}}(p) &\to SS_{\mathfrak{O}}(p)\\ ([\mathfrak{a}], (E,\iota)) &\mapsto \mathfrak{a} \star (E,\iota) := (E^{\mathfrak{a}}, (\varphi_{\mathfrak{a}})_*(\iota)).\end{aligned}$$

In addition, for any given orbit O *and any given primitively* $\mathfrak{O}$*-oriented supersingular elliptic curve* (E,ι)*, either* (E,ι) *or its* $\mathfrak{O}$*-twist* $(E,\bar{\iota})$*, where* $\bar{\iota}(\alpha) = \iota(\bar{\alpha})$*, is in* O*.*

Proof. This proposition is obtained from [Onu21, Theorem 3.4] and from [Onu21, Proposition 3.3]. In particular, inside the proof of [Onu21, Proposition 3.3], it is shown that either (E,ι) or $(E,\bar{\iota})$ is in a given orbit. □

2.3 Representation of isogenies

Definition 7 (Efficient representation, following [Wes24, Definition 1.3])**.** Let $\mathscr{A}$ be a polynomial time algorithm. It is an **efficient isogeny evaluator** if for any $D \in \{0,1\}^*$ such that $\mathscr{A}(\texttt{validity}, D)$ outputs $\top$, there exists an isogeny $\varphi : E \to E'$ (defined over some finite field $\mathbb{F}_q$) such that:

1. on input $(\texttt{curves}, D)$, $\mathscr{A}$ returns (E, E'),
2. on input $(\texttt{degree}, D)$, $\mathscr{A}$ returns $\deg(\varphi)$,
3. on input $(\texttt{eval}, D, P)$ with $P \in E(\mathbb{F}_{q^k})$, $\mathscr{A}$ returns $\varphi(P)$.

If furthermore D is of polynomial size in $\log(\deg\varphi)$ and $\log q$, then D is an **efficient representation** of φ (with respect to $\mathscr{A}$).

There are several ways of representing efficiently isogenies. For instance, by considering an isogeny φ as a chain of smaller degree isogenies $\varphi_1 \circ \cdots \circ \varphi_n$, one can represent it by the list of kernels $\ker(\varphi_i)_{i=1}^n$ and use Vélu's formulae as the main procedure of the efficient isogeny evaluator. When the degree φ is smooth, it then is possible to find a suitable chain of isogenies to represent φ efficiently with this method. This is an important example as it was one of the most used representation in isogeny-based cryptography before SIDH's attack. Thanks to higher dimensional isogenies, we now have access to interpolation algorithms without any smoothness constraints. One can for instance use [Rob23], or directly the results of Section 4, to efficiently represent an isogeny from its image of a large enough subgroup.

For a more exhaustive list of efficient ways to represent isogenies, we refer the reader to [Rob24]. One can consider a single general efficient isogeny evaluator which, depending of some bits of the input, uses a different subalgorithm (higher dimensional interpolation, evaluation of the isogenies as a chain, etc.). Therefore, in the rest of the article, the algorithm $\mathscr{A}$ is left implicit, and we simply say that an isogeny φ is in efficient representation.

Definition 8 (Efficient representation of orientation)**.** Let (E,ι) be an $\mathfrak{O}$-orientated elliptic curve. An **efficient representation** of ι is a pair (ω, D) where ω is a generator of $\mathfrak{O}$, and D is an efficient representation of $\iota(\omega) \in \mathrm{End}(E)$.

We now define a function `enc`, introduced in [Wes22a], which returns a unique encoding of the K-isomorphism class of a $\mathfrak{O}$-oriented elliptic curve. It takes as input a representation of an oriented elliptic curve $(E,\iota) \in SS_{\mathfrak{O}}(p)$ and returns a unique triple (E,P,Q) assuming that we have fixed in advance:

- A canonical representative for each $\overline{\mathbb{F}}_p$-isomorphism class of elliptic curves over $\overline{\mathbb{F}}_p$; for instance, the curve of equation $E : y^2 + xy = x^3 - (36x+1)/(j(E) - 1728)$ for any $j(E) \notin \{0, 1728\}$, see [Sil86, page 52],

- A generator ω of $\mathfrak{O}$, typically one with the smallest possible norm,
- A deterministic procedure that takes as input an elliptic curve E in canonical form and returns a point $P \in E$ of order greater than $4N(\omega)$.

Then the map $\texttt{enc} : (E, \iota) \mapsto (E, P, Q)$ is given by constructing the point P of order greater than $4N(\omega)$ using the deterministic procedure and setting Q to be $\iota(\omega)(P)$. As shown in [Wes22a], this encoding is an unique encoding of the K-isomorphism class of (E, ι). Moreover, when ι is efficiently represented, the encoding $\texttt{enc}(E, \iota)$ can be computed in polynomial time. Thus checking if two $\mathfrak{O}$-oriented elliptic curves are K-isomorphic is done in polynomial time using `enc`.

When the j-invariant of the oriented curve is 0 or 1728, one needs to use another canonical form, see [Sil86, page 52], for instance

$$E : y^2 + y = x^3, \text{ if } j = 0$$

and

$$E : y^2 = x^3 + x, \text{ if } j = 1728.$$

In these cases, one also needs to consider the non-trivial automorphisms of the elliptic curve and thus to replace Q by the set $\{(\sigma_*\iota)(\omega)(P) | \sigma \in \mathrm{Aut}(E)\}$.

Finally, we define the image of any set S of oriented supersingular elliptic curves by `enc` as the set of their unique encoding by `enc`, denoted $\texttt{enc}(S)$.

This unique encoding of K-isomorphism class of $\mathfrak{O}$-oriented elliptic curves is a crucial ingredient for Section 6: First, it provides an efficient method for checking whether two $\mathfrak{O}$-oriented elliptic curves belong to the same class, which is central to finding collisions in the meet-in-the-middle approach for the classical resolution of the $\mathfrak{O}$-Vectorisation problem. Second, to properly use the Kuperberg's algorithm and quantumly solve $\mathfrak{O}$-Vectorisation, it is essential to have a unique representation of the oriented elliptic curves we are dealing with, i.e. up to K-isomorphism.

In this paper, unless otherwise specified, when an algorithm takes as input an isogeny, we mean that the isogeny is given with an efficient representation. It is also the case for orientations taken as input.

3 The endomorphism ring problem and its friends

One of the central problems in (supersingular) isogeny-based cryptography is the following ℓ-IsogenyPath problem, where ℓ is a prime number.

Problem 1 (ℓ-IsogenyPath)**.** *Given two supersingular elliptic curves E and E' over $\mathbb{F}_{p^2}$ and a prime $\ell \neq p$, find a chain of ℓ-isogenies from E to E'.*

The ℓ-IsogenyPath problem is considered to be the fundamental problem at the heart of isogeny-based cryptography. This problem has been shown to be equivalent, first under some heuristics [EHL+18] then only under GRH [Wes22b], to the problem of finding the structure of the endomorphism ring of a supersingular elliptic curve. This second problem is called the endomorphism ring problem; here we refer to it as the EndRing problem. Since for any supersingular elliptic curve E defined over a finite field of characteristic p, $\mathrm{End}(E)$ is isomorphic to a maximal order of the quaternion algebra $B_{p,\infty}$, the EndRing problem comes in two flavors. One can either look for four isogenies generating $\mathrm{End}(E)$ as a lattice or for four quaternions generating a maximal order which is isomorphic to $\mathrm{End}(E)$. The notion of ε-basis unifies those approaches under GRH, see [Wes22b].

Definition 9 (ε-basis)**.** Let $\varepsilon : B_{p,\infty} \to \mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$ be an isomorphism and $L \subseteq B_{p,\infty}$ be a lattice. We call a pair (α, θ), where $(\alpha_i)_{i=1}^{\mathrm{rank}\, L}$ is a basis of L and $\theta_i = \varepsilon(\alpha_i)$, an ε-basis of L. The pair (α, θ) will be called an ε-basis of $\varepsilon(L)$ as well.

Problem 2 (EndRing)**.** *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$, find an ε-basis of* $\mathrm{End}(E)$*.*

The current best classical algorithms to solve EndRing run in expected time $\tilde{O}(p^{1/2})$, see for instance [EHL+20] (or [PW24, Theorem 8.8] for a heuristic-free algorithm), and the best quantum algorithms have complexity in $\tilde{O}(p^{1/4})$, see for example [BJS14].

The notion of orientation (see Section 2.2), introduced by Colò and Kohel in [CK20], induces a variant of EndRing where partial information on the endomorphism ring is given.

Problem 3 ($\mathfrak{O}$-EndRing)**.** *Given a primitively $\mathfrak{O}$-oriented supersingular elliptic curve (E, ι) over a finite field of characteristic p, find an ε-basis of* $\mathrm{End}(E)$*.*

The study of this problem is not only important to see how the complexity of EndRing is impacted by the knowledge of a single non-trivial endomorphism but also because it is in fact equivalent, under GRH, to the $\mathfrak{O}$-Vectorisation problem [Wes22a].

Problem 4 ($\mathfrak{O}$-Vectorisation)**.** *Given $(E, \iota), (E', \iota') \in SS_{\mathfrak{O}}(p)$ two oriented supersingular elliptic curves, find an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $E^{\mathfrak{a}} \simeq E'$.*

This hardness of the problem $\mathfrak{O}$-Vectorisation measures whether or not an action induced by a primitive orientation is a one-way function. Hence, recovering the keys of CSIDH-like protocols, such as [CD20, CS22, FFK+23], reduces to $\mathfrak{O}$-Vectorisation.

In this paper, we only need the following reduction between the two problems.

Proposition 3 (**GRH**, Proposition 7 in [Wes22a])**.** *Given the factorisation of* $\mathrm{disc}(\mathfrak{O})$*, the $\mathfrak{O}$-EndRing problem reduces to $\mathfrak{O}$-Vectorisation in probabilistic polynomial time in the length of the instance.*

In the current state of the art, the $\mathfrak{O}$-Vectorisation problem can heuristically be solved in expected classical time $l^{O(1)}|\mathrm{disc}(\mathfrak{O})|^{1/4}$, with l the length of the input, using for instance a meet-in-middle approach in a similar way as presented in [DG16]. Quantumly, $\mathfrak{O}$-Vectorisation can heuristically be solved in subexponential time in the length of the discriminant of $\mathfrak{O}$, see [Wes22a, Proposition 4]. Hence, knowing an orientation seems to make a significant difference in the expected runtime to solve EndRing.

However, those resolutions and reductions need the orientation to be primitive and assume several heuristics. When the orientation is not primitive, this is equivalent to knowing one non-trivial endomorphism of the curve. Obviously, knowing an orientation also gives the knowledge of a non-trivial endomorphism. In the other direction, given a non-trivial endomorphism, one can compute in polynomial time its degree and trace [Koh96, Proposition 81] and deduce a quadratic number α such that $\mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$ is an orientation.

We introduce a variant of $\mathfrak{O}$-EndRing where the given orientation is not required to be primitive.

Problem 5 (α-EndRing)**.** *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ and an orientation $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$, find an ε-basis of* $\mathrm{End}(E)$*.*

The following Primitivisation problem has been introduced in [ACL+23] as a hard problem. It forms a bridge between the α-EndRing and $\mathfrak{O}$-EndRing problems.

Problem 6 (PRIMITIVISATION)**.** *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ and an orientation $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$, find a primitive orientation $\iota' : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$ such that the order $\mathbb{Z}[\alpha]$ is contained in the order $\mathfrak{O}$.*

In [ACL+23] the authors also give a quantum algorithm for solving it in subexponential time under some heuristics, and conjecture that it is hard to solve in general. Moreover, they give a quantum algorithm to solve the ℓ-IsogenyPath problem given non-primitive orientation that uses their Primitivisation algorithm as a subprocedure. Inevitably, their algorithm inherits the need for heuristics of the subprocedure. In Section 4, we develop tools involving higher dimensional isogenies. These tools are used in Section 5 in there is an algorithm solving PRIMITIVISATION for an orientation $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$ in classical polynomial time given the factorisation of $\mathrm{disc}(\mathbb{Z}[\alpha])$. It directly yields a classical subexponential and a quantum polynomial reduction of α-EndRing to $\mathfrak{O}$-EndRing.

This implies, together with Proposition 3, reductions of α-EndRing to $\mathfrak{O}$-Vectorisation. This reasoning is formalized in Section 6 together with a rigorous analysis of the complexity of $\mathfrak{O}$-Vectorisation.

4 Efficient division of isogenies

In this section, we discuss higher dimensional isogenies and how they can be used to efficiently divide isogenies by integers. The goal of this section is to prove Theorem 3 below (and its more precise formulation Theorem 4).

Theorem 3. *Algorithm 1 takes as input*

- *Two elliptic curves E_1 and E_2 defined over $\mathbb{F}_{p^k}$,*
- *An isogeny $\varphi : E_1 \to E_2$ over $\mathbb{F}_{p^k}$ in efficient representation,*
- *An integer $n < \deg \varphi$,*

and returns an efficient representation of φ/n if this quotient is an isogeny(and otherwise returns `False`), and runs in time polynomial in $\log p$ and $\log \deg(\varphi)$.

More precisely, this returned representation of φ/n is of size $O(k \log(p) \log^3(\deg \varphi))$ and allows one to evaluate it at any point in $\tilde{O}(\log^{11}(\deg \varphi))$ operations over its field of definition.

Before proving it, let us state an immediate corollary.

Corollary 1 (General division of isogenies)**.** *There is an algorithm which on input*

- *Three elliptic curves E_1, E_2 and E_3 defined over $\mathbb{F}_{p^k}$,*
- *Two isogenies $\varphi : E_1 \to E_2$, $\eta : E_1 \to E_3$ over $\mathbb{F}_{p^k}$ in efficient representation,*

returns an efficient representation of the isogeny ψ such that $\varphi = \psi \circ \eta$ if it exists (and otherwise returns `False`), and runs in time polynomial in the length of the input.

Proof. Apply Theorem 3 to the isogeny $\tilde{\varphi} = \varphi \circ \hat{\eta}$ and the integer $n = \deg(\eta)$. The isogeny ψ, if it exists, is precisely $\tilde{\varphi}/n$. □

The machinery of higher dimensional isogenies is only used in this section, and the reader may admit the main theorem and skip the rest of the section without impairing global understanding. Some definitions require notions of algebraic geometry which will not be recalled here. If necessary, the reader may refer to [Mil86]. Let us emphasise that the ideas underlying Theorem 3 and its proof originate from [Rob22b]. Theorem 3 and

its proof are simply expressed in higher generality and greater detail than [Rob22b] provides.

Initially, the idea of exploiting higher dimensional isogenies for efficient computation of isogenies between elliptic curves was introduced by Castryck and Decru to attack SIDH, see [CD23]. Among other ingredients, this attack relies on the generalization of Vélu's formulae by Lubicz and Robert, see [LR12]. The attack of Castrick and Decru has since been developed further, notably by Robert who generalised the attack [Rob23] and found other applications to elliptic curves, see [Rob22a] and [Rob22b].

We now turn to a presentation of principally polarised abelian varieties — which constitute the suitable generalisation of elliptic curves to higher dimensions, see e.g. [Mil86].

For any abelian varieties A and A' and any isogeny $\varphi : A \to A'$, the dual variety of A is denoted by $\hat{A}$ and the dual isogeny of φ is denoted by $\hat{\varphi} : \hat{A}' \to \hat{A}$.

Definition 10 ((Principally) Polarised abelian varieties)**.** Let A be an abelian variety. One can derive from an ample divisor of A an isogeny $\lambda : A \to \hat{A}$. Such an isogeny is called a **polarisation** of A. It is a **principal polarisation** if λ is an isomorphism. For any (principal) polarisation λ of A, the pair (A, λ) is a **(principally) polarised abelian variety**.

Remark 1. Elliptic curves are principally polarised abelian varieties having a unique principal polarisation, simply denoted by λ_E for a given elliptic curve E. Hence, there exists a natural principal polarisation over products of elliptic curves induced by the product of the polarisations. We call this polarisation the **product polarisation** and denote it $\lambda_{E_1 \times \cdots \times E_n}$ for the product of elliptic curve $E_1 \times \cdots \times E_n$. When we consider a product of elliptic curves as a principally polarised abelian variety without specifying the polarisation, it means that it is polarised by the product polarisation.

In this section, we shall mostly focus on isogenies between products of elliptic curves.

Let $E_1, \ldots, E_n, E'_1, \ldots, E'_m$ be elliptic curves and $\varphi_{i,j} : E_j \to E'_i$ be isogenies of elliptic curves where $i \in [\![1, m]\!], j \in [\![1, n]\!]$. From this set of isogenies, we naturally get the following map between products of elliptic curves

$$\begin{aligned} E_1 \times \cdots \times E_n &\longrightarrow E'_1 \times \cdots \times E'_m \\ (P_1, \ldots, P_n) &\longmapsto (\sum_{j=1}^{n} \varphi_{1,j}(P_j), \ldots, \sum_{j=1}^{n} \varphi_{m,j}(P_j)). \end{aligned}$$

This map can be represented by the matrix $(\varphi_{i,j})_{i \in [\![1,m]\!], j \in [\![1,n]\!]}$ called the **matrix form**. If it has a finite kernel and $n = m$ then it is an isogeny.

Given an isogeny $\varphi : E \to E'$ between elliptic curves, one can construct an isogeny, denoted $\varphi^{\times n}$, in dimension n from E^n to E'^n by setting

$$\varphi^{\times n}(P) = (\varphi(P_1), \ldots, \varphi(P_n)), \forall P = (P_1, \ldots, P_n) \in E^n.$$

The matrix form of $\varphi^{\times n}$ is then the identity matrix of dimension n "multiplied" by φ. Any isogeny $F : E_1 \times \cdots \times E_n \to E'_1 \times \cdots \times E'_n$ between two products of elliptic curves can be written using a matrix form with the following injection maps

$$\begin{aligned} \tau_j : \quad E_j &\longrightarrow E_1 \times \cdots \times E_n \\ P &\longmapsto (\underbrace{0, \ldots, 0}_{i-1}, P, \underbrace{0, \ldots, 0}_{n-i}) \end{aligned}$$

and projection maps

$$\begin{array}{rccc} \pi_i : & E'_1 \times \cdots \times E'_n & \longrightarrow & E'_i \\ & (P_1, \ldots, P_n) & \longmapsto & P_i \end{array}$$

with $i, j \in [\![1, n]\!]$. Indeed, by defining the isogeny $F_{i,j} : E_j \to E'_1$ as $\pi_i \circ F \circ \tau_j$, for all $i, j \in [\![1, n]\!]$, we get

$$F(P_1, \ldots, P_n) = \Big(\sum_{j=1}^{n} F_{i,j}(P_j) \Big)_{1 \leq i \leq n},$$

for any $(P_1, \ldots, P_n) \in E_1 \times \cdots \times E_n$. We thus define the matrix form of the isogeny F as $M(F) = (F_{i,j})_{i,j \in [\![1,n]\!]}$.

Remark 2. In this paper, we shall consider only isogenies in higher dimensions whose domain is a product of elliptic curves that are principally polarised by the product polarisation. However, to compute such isogenies, one needs to use a coordinate system capable of handling any principally polarised abelian varieties of the same dimension. Currently, the most commonly used coordinate system is the theta model. In particular, this is the case in the generalisation of the Vélus' formulae we shall use.

The theta coordinates of a product of elliptic curves can be obtained by multiplying the theta coordinates of each elliptic curve in the product. One can compute theta coordinates of an elliptic curve directly from its 4-torsion subgroup. Therefore, converting a principally polarised product of elliptic curves to the theta model is innexpensive. We shall not delve deeper into the machinery of theta functions here; further information can be found in [DLRW24] and [Rob21].

Thanks to the previous notations, we can provide a formal definition of what we mean by embedding an isogeny in higher dimensions. For convenience, we generalise the notion of efficient representation of isogenies between elliptic curves to isogenies between products of elliptic curves. Mainly, we require an efficient representation of isogenies between elliptic curves to be associated to an algorithm such that one can compute the image of any tuple of points in time polynomial in the length of the representation and in the size of the field over which those points are defined.

Definition 11 (Embedding representation)**.** Let n be an integer and $E_1, \ldots, E_n, E'_1, \ldots, E'_n$ be elliptic curves. Let $\varphi : E \to E'$ be an isogeny such that $E \in \{E_1, \ldots, E_n\}$ and $E' \in \{E'_1, \ldots, E'_n\}$. An **embedding representation** of φ in dimension n is a triplet (F, i, j) associated to a representation of F, where $F : E_1 \times \cdots \times E_n \to E'_1 \times \cdots \times E'_n$, $i, j \in [\![1, n]\!]$ and such that $\varphi(P) = \pi_j \circ F \circ \tau_i$ for any $P \in E$.

We now introduce a notion of duality with respect to the principal polarisations allowing us to define a notion of isogenies between principally polarised abelian varieties behaving in a very similar way to elliptic curve isogenies.

Definition 12 (N-Isogenies)**.** Let (A, λ) and (A', λ') be two principally polarised abelian varieties. Let $\varphi : A \to A'$ be an isogeny. We define the **dual isogeny of φ with respect to the principal polarisations** as the isogeny $\tilde{\varphi} := \lambda^{-1} \circ \hat{\varphi} \circ \lambda' : A' \to A$. We say that $\varphi : (A, \lambda) \to (A', \lambda')$ is an ***N*-isogeny of principally polarised abelian varities** if $\tilde{\varphi} \circ \varphi = [N]$.

Let M be the matrix form of an isogeny between products of elliptic curves. The **adjoint matrix** of M is $\tilde{M} := (\hat{M}_{j,i})_{i,j \in [\![1,n]\!]}$ which is the transpose of the matrix whose entries are the dual entries of M. The dual isogeny, with respect to the product polarisations, of the isogeny given by M has for matrix form the adjoint matrix of M.

The notions of **algorithms of evaluation of isogenies** and **representations of an isogeny** naturally extend to N-isogenies. Notice that each N-isogeny is associated to

some principal polarisations and thus algorithms of evaluation of N-isogenies also return the principal polarisation of the codomains.

Separable isogenies between elliptic curves are determined by their kernel (up to isomorphisms of the target curve). Given a kernel, the corresponding isogeny can be evaluated using Vélu's formulae, see [Vé71]. We have similar results for N-isogenies with N prime to the characteristic of the field of definition. Indeed, such isogenies are determined by their kernel, which is maximal isotropic, and there exists an analogue to Vélu's formulae for them. This notion of maximal isotropy is central and requires to introduce the Weil pairing for principally polarised abelian varieties.

Definition 13 (Polarised Weil pairing)**.** Let (A, λ) be a polarised abelian variety over a field and N be prime to the characteristic of this field. There exists a **canonical nondegenerate pairing** $e_N : A[N] \times \hat{A}[N] \to \mu_N(\bar{\mathbb{F}})$, where $\mu_N(\bar{\mathbb{F}})$ is the group of Nth roots of 1 in $\bar{\mathbb{F}}$. This pairing is called the Weil N-pairing. The **polarised Weil N-pairing** $e_{N,\lambda}$ is then the canonical nondegenerate pairing $A[N] \times A[N] \to \mu_N(\bar{\mathbb{F}}), (P, Q) \mapsto e_N(P, \lambda(Q))$.

Definition 14 (Maximal isotropic subgroup)**.** With the same notations as in Definition 13. Let H be a proper subgroup of $A[N]$. The subgroup H is **maximal isotropic in $A[N]$** if the polarised Weil pairing $e_{N,\lambda}$ is trivial over H but is not over any proper supergroup of H. For an isogeny of A having a maximal isotropic kernel in $A[N]$ is equivalent to be an N-isogeny.

Lemma 1 (Proposition 1.1 in [Kan97])**.** *Let $(A, \lambda), (A', \lambda')$ and (A'', λ'') be principally polarised abelian varieties such that there exist $\varphi' : (A, \lambda) \to (A', \lambda')$ and $\varphi'' : (A, \lambda) \to (A'', \lambda'')$ two N-isogenies with $\ker \varphi' = \ker \varphi''$, where N is coprime to the characteristic of the abelian varieties' field of definition. Then there is an isomorphism γ between A' and A'' such that $\varphi'' = \gamma \circ \varphi'$ and $\lambda' = \widehat{\gamma} \circ \lambda'' \circ \gamma$, i.e. $\gamma : (A', \lambda') \to (A'', \lambda'')$ is a 1-isogeny. We say that γ is an **isomorphism of principally polarised abelian varieties**.*

In further results of this section, we shall need to recover endomorphisms of a given product of elliptic curves $E^n \times E'^n$ from its kernel. Thus, it is important to have a description of the group of the automorphisms of $E^n \times E'^n$ as, by Lemma 1, endomorphisms with the same kernel differ only by an automorphism.

Lemma 2. *Let E_1 and E_2 be two elliptic curves and n, m be two integers. Let $\mathrm{Aut}(E_1^n \times E_2^m, \lambda_{E_1^n \times E_2^m})$ be the group of automorphisms of the principally polarised abelian variety $(E_1^n \times E_2^m, \lambda_{E_1^n \times E_2^m})$. Then for any element ψ of $\mathrm{Aut}(E_1^n \times E_2^m, \lambda_{E_1^n \times E_2^m})$, we have*

$$M(\psi) = \begin{pmatrix} A_1 & B_{1,2} \\ B_{2,1} & A_2 \end{pmatrix},$$

where for any $i, j \in \{1, 2\}$, A_i is a matrix of dimension n_i with entries in $\mathrm{Aut}(E_i) \cup \{0\}$ and $B_{i,j}$ is a matrix of dimension $n_i \times n_j$ with entries in $\mathrm{Iso}(E_j, E_i) \cup \{0\}$. Moreover $M(\psi)$ contains only one non-zero entry per column and per row.

Proof. Let $\psi \in \mathrm{Aut}(E_1^{n_1} \times E_2^{n_2}, \lambda_{E_1^{n_1} \times E_2^{n_2}})$. As ψ is an automorphism of a principally polarised abelian variety, we have $\hat{\psi} \circ \lambda \circ \psi = \lambda$ thus $\psi\tilde{\psi} = [1]$, which, in matrix form, gives $M(\psi)\tilde{M}(\psi) = \mathrm{I}_{n_1+n_2}$. Let us denote by $(\psi_{i,j})_{1 \le i,j \le n_1+n_2}$ the coefficients of the matrix form $M(\psi)$. For any $i \in [\![1, n_1 + n_2]\!]$, we have

$$[1] = \sum_{j=1}^{n_1+n_2} \psi_{i,j} \circ \hat{\psi}_{i,j} = \sum_{j=1}^{n_1+n_2} [\mathrm{degree}(\psi_{i,j})].$$

This implies that for any i, exactly one of the isogenies $\psi_{i,j}$ is non-zero, and that isogeny has degree one, hence it is an isomorphism. The identity $\psi\psi = [1]$ yields the same results

for columns. Moreover, for ψ to be well-defined, the domain and codomain of each $\psi_{i,j}$ must be

$$\text{domain}(\psi_{i,j}) = \begin{cases} E_1, & \text{if } 1 \leq j \leq n_1, \\ E_2, & \text{if } n+1 \leq j \leq n_1+n_2, \end{cases}$$
$$\text{codomain}(\psi_{i,j}) = \begin{cases} E_1, & \text{if } 1 \leq i \leq n_1, \\ E_2, & \text{if } n+1 \leq i \leq n_1+n_2. \end{cases}$$

□

As presented by Robert in [Rob22b], isogenies between abelian varieties can be embedded into isogenies of higher dimensions. Namely, given an isogeny φ between abelian varieties, one can construct a higher dimensional isogeny such that one of its matrix form coefficients is equal to f, up to isomorphism. This result is a generalisation of a construction in dimension 1 given by Kani in [Kan97].

Lemma 3 (Lemma 3.4. in [Rob23])**.** *Let A and B be two principally polarised abelian varieties of dimension g over a base field of characteristic p. Let φ_1, φ_2' be two d_1-isogenies and φ_2, φ_1' be two d_2-isogenies such that $(d_1+d_2, p) = 1$ and $\varphi_1' \circ \varphi_1 = \varphi_2' \circ \varphi_2$ is a d_1d_2-isogeny from A onto B, i.e.*

$$\begin{array}{ccc} A & \xrightarrow{\varphi_1} & \varphi_1(A) \\ {\scriptstyle\varphi_2}\downarrow & & \downarrow{\scriptstyle\varphi_1'} \\ \varphi_2(A) & \xrightarrow{\varphi_2'} & B \end{array}.$$

Then

$$\begin{pmatrix} \varphi_1 & \widetilde{\varphi_1'} \\ -\varphi_2 & \widetilde{\varphi_2'} \end{pmatrix}$$

is the matrix form of a (d_1+d_2)-isogeny $F : A \times B \to \varphi_1(A) \times \varphi_2(A)$. Moreover, if $\gcd(d_1, d_2) = 1$ then the kernel of F is $\widetilde{F}(\varphi_1(A)[d_1+d_2] \times \{0\})$ and is of rank $2g$.

Lemma 4 below describes how an isogeny φ of degree N between elliptic curves can be embedded into an N'-endomorphism in dimension 8, for some $N' > N$. This lemma exploits the recent techniques developed to attack SIDH [CD23, MMP+23, Rob23].

Lemma 4. *Let E_1 and E_2 be two elliptic curves over a finite field $\mathbb{F}_{p^k}$ and $\varphi : E_1 \to E_2$ be an isogeny of degree N. Let $N' > N$ be an integer such that $(N', Np) = 1$. Let m_1, m_2, m_3 m_4 be integers such that $m_1^2 + m_2^2 + m_3^2 + m_4^2 = N' - N$ and let α_{E_1} (resp. α_{E_2}) be the endomorphism over E_1^4 (resp. E_2^4) given by the matrix*

$$\begin{pmatrix} m_1 & -m_2 & -m_3 & -m_4 \\ m_2 & m_1 & m_4 & -m_3 \\ m_3 & -m_4 & m_1 & m_2 \\ m_4 & m_3 & -m_2 & m_1 \end{pmatrix}.$$

Let $H := \{(\tilde{\alpha}_{E_1}(P), \varphi^{\times 4}(P)) | P \in E_1^4[N']\}$; then there exists an N'-isogeny of $E_1^4 \times E_2^4$ of kernel H.
Furthermore, the following holds for any N'-isogeny G of $E_1^4 \times E_2^4$ of kernel H.

- *The codomain of G is isomorphic to $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ as principally polarised abelian varieties.*

- *For any isomorphism* $\gamma : G(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4}) \to (E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$*, there exist an integer* $i \in [\![1, 8]\!]$ *and an isomorphism* ψ *in* $\mathrm{Iso}(E_2, E')$*, where* $E' \in \{E_1, E_2\}$*, such that the following diagram commutes*

$$\begin{array}{ccc} E_1 & \xrightarrow{G \circ \tau_1} & G(E_1^4 \times E_2^4) \\ {\scriptstyle\varphi}\downarrow & & \downarrow{\scriptstyle\pi_i \circ \gamma} \\ E_2 & \xrightarrow{\psi} & E' \end{array}$$

i.e. $\pi_i(\gamma(G(\tau_1(P)))) = \psi(\varphi(P))$*, for all* $P \in E_1$*, and thus* $(\gamma \circ G, 1, i)$ *is an embedding representation of* $\psi \circ \varphi$*.*

Proof. We use the same notations as above.
Since the matrix form of $\varphi^{\times 4}$ is diagonal, we have the following commutative diagram

$$\begin{array}{ccc} E_1^4 & \xrightarrow{\alpha_{E_1}} & E_1^4 \\ {\scriptstyle\varphi^{\times 4}}\downarrow & & \downarrow{\scriptstyle\varphi^{\times 4}} \\ E_2^4 & \xrightarrow{\alpha_{E_2}} & E_2^4. \end{array}$$

By construction of $\varphi^{\times 4}$ is an N-isogeny and α_{E_1} and α_{E_2} are $(N' - N)$-isogenies. Thus, the sum of the degree of $\varphi^{\times 4}$ with the degree of α_{E_1} or α_{E_2} is equal to N'. By assumption, N' is coprime to Np. In particular N' is coprime to p and N is coprime to $N' - N$. Hence, by taking $A := E_1^4, B := E_2^4, \varphi_1 := \alpha_{E_1}, \varphi_2' := \alpha_{E_2}, \varphi_2 := \varphi^{\times 4}$ and $\varphi_1' := \varphi^{\times 4}$, we have $d_1 = N' - N$, $d_2 = N$ and all the assumptions of Lemma 3 are satisfied. Its application gives us an N'-endomorphism F of $E_1^4 \times E_2^4$ with kernel

$$\ker F = \{(\tilde{\alpha}_{E_1}(P), \varphi^{\times 4}(P)) | P \in E_1^4[N']\}$$

and matrix form

$$M(F) = \begin{pmatrix} M(\alpha_{E_1}) & M(\widetilde{\varphi^{\times 4}}) \\ -M(\varphi^{\times 4}) & M(\widetilde{\alpha_{E_2}}) \end{pmatrix}.$$

Then, for any $P \in E_1$, we have

$$F(\tau_1(P)) = (m_1 P, m_2 P, m_3 P, m_4 P, -\varphi(P), 0, 0, 0) \tag{1}$$

Let G be an N'-isogeny of $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ with $\ker G = \ker F$. By Lemma 1, $G(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ and $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ are isomorphic and for any isomorphism γ from $G(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ to $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ there exists an automorphism ψ of $E_1^4 \times E_2^4$ such that we have

$$\gamma \circ G = \psi \circ F. \tag{2}$$

By Lemma 2, there exist 8 isomorphisms $\psi_1, \dots \psi_8$ such that

$$\psi_i \in \begin{cases} \mathrm{Aut}(E_1) \cup \mathrm{Iso}(E_1, E_2), & \text{if } i \in [\![1, 4]\!], \\ \mathrm{Aut}(E_2) \cup \mathrm{Iso}(E_2, E_1), & \text{if } i \in [\![5, 8]\!] \end{cases}$$

and a map σ permuting coordinates of the points of $E_1^4 \times E_2^4$ such that, for any point $(P_1, \dots, P_8)$ of $E_1^4 \times E_2^4$, we have

$$\psi(P_1, \dots, P_8) = \sigma(\psi_1(P_1), \dots, \psi_8(P_8)). \tag{3}$$

Let i be the integer such that $\pi_i(\sigma(Q_1, \dots, Q_8)) = Q_5$ for any $(Q_1, \dots, Q_8) \in E_1^4 \times E_2^4$. Then, for any $P \in E_1$,

$$\begin{aligned}\pi_i(\gamma(G(\tau_1(P)))) &= \pi_i(\psi(F(\tau_1(P)))), \text{ by (2)},\\ &= \pi_i(\sigma((\psi_1(m_1P), \dots, \psi_4(m_4P), \psi_5(-\varphi(P)), 0, 0, 0))), \text{ by (1)},\\ &= -\psi_5(\varphi(P)), \text{ by construction } \pi_i.\end{aligned}$$

This conclude the proof as $-\psi_5$ is an element of $\mathrm{Aut}(E_2) \cup \mathrm{Iso}(E_2, E_1)$. □

This embedding can be evaluated efficiently using the analogue of Vélu's formulae in higher dimension introduced by Lubicz and Robert, [LR12]. The next Lemma state the complexity we

Lemma 5. *[Rob22a] Let $(E_1 \times \cdots \times E_n, \lambda_{E_1 \times \cdots \times E_n})$ be a principally polarised abelian varity where E_i are elliptic curves defined over $\mathbb{F}_{p^k}$. Let H be a maximal isotropic subgroup of $(E_1 \times \cdots \times E_n)[N']$ where N' is an integer coprime to p of prime factorisation $\prod_{i=1} \ell_i^{e_i}$.*

Given H as a set of generators living in $(E_1 \times \cdots \times E_n)[\ell_i^{e_i}]$, one can compute a representation of an N'-isogeny G of $(E_1 \times \cdots \times E_n, \lambda_{E_1 \times \cdots \times E_n})$ with kernel H such that

- *it takes $O(B^8 D \log^2(N') \log(B))$ arithmetic operations over $\mathbb{F}_{p^k}$ to get this representation,*
- *the representation has size $O(kM \log(N') \log p)$ bits,*
- *the representation allows to evaluate G on a point in $O(B^8 M \log(N') \log(B))$ operations over its field of definition,*

where B, M and D are any bounds such that $B \geq P^*(N')$, $M \geq \max_{i=1} \delta_{E_i}(N')$ and $D \geq \max_{i=1} \delta_{E_i,2}(N')$.

Proof. This result is simply a rephrasing of [Rob22a, 4. The algorithm]. The main idea behind achieving this complexity is to compute the higher dimensional isogeny as a chains of power prime isogenies. In the original description of the algorithm, Robert uses [LR23] to compute representation of each of these isogenies. For the same complexity, we suggest using [DLRW24, Theorem 53] instead, as it provides a more convenient statement. In particular, we have access to an explicit description of the theta coordinates of the output. □

Remark 3. In this section, we always assume that the isogenies are embedded into dimension 8. Lemma 5, and so all the results derived from it, could be more efficient if the isogenies were embedded into dimensions 2 or 4, unfortunately, it is not always possible. Indeed, for dimension 8, we decompose $N' - N$ as a sum of four squares to construct an endomorphism of E^4 using the Zarhin's trick, see [Zar74]. For dimension 2 (resp. 4), $N' - N$ needs to be a square (resp. a sum of two squares) to construct easily an endomorphism of E (resp. E^2) and to embed the isogenies into dimension 2 (resp. 4). It is possible to relax these conditions, under some heuristics and when the endomorphism ring is known. Here, we neither want to rely on heuristics, nor presume that we know the endomorphism ring, so we only consider the case of dimension 8.

By Lemma 4, if an isogeny $\varphi : E_1 \to E_2$ is divisible by an integer n then it can be embedded into an isogeny in dimension 8 of kernel

$$H := \{(\tilde{\alpha_{E_1}}(P), (\varphi/n)^{\times 4}(P)) | P \in E_1^4[N']\},$$

with N' and α_{E_1} defined as in the lemma. Thanks to Lemma 5, one can compute in polynomial time this 8-dimensional isogeny from its kernel. It only remains to show how we can compute the kernel H to get a complete division algorithm.

By construction N' is an integer coprime to $\deg(\varphi)$ and n, then we have

$$(\varphi/n)^{\times 4}(P) = (s\varphi)^{4\times}(P) \text{ with } n^{-1} = s \mod N'.$$

This trick allows us to compute the kernel H from an efficient representation of φ. Lemma 5 ensures that computing H this way always provides a maximal isotropic subgroup of $E_1^4 \times E_2^4$ even if φ/n is not a well-defined isogeny. This will be crucial to verify the divisibility of an isogeny by an integer.

Lemma 6. *Let $\varphi : E_1 \to E_2$ be an isogeny. Let n^2 be a divisor of $\deg(\varphi)$ and $N = \deg(\varphi)/n^2$. Let $N' > N$ such that $(N', p\deg(\varphi)) = 1$ and $s = n^{-1} \mod N'$. Let α_{E_1} be an m-endomorphism of E_1^4 with $m = N' - N$.*
Then $H := \{(\tilde{\alpha}_{E_1}(P), s\varphi^{\times 4}(P)) | P \in E_1^4[N']\}$ is a maximal isotropic subgroup of $(E_1^4 \times E_2^4)[N']$.

Proof. The subgroup structure of H comes immediately by construction. We claim that H is maximal isotropic. Let λ_1 be the product polarisation over E_1^4 and λ_2 be the product polarisation over E_2^4. Let us show that the Weil pairing $e_{N',\lambda_1\times\lambda_2}$ is trivial between $(\tilde{\alpha}_{E_1}(P), s\varphi^{\times 4}(P))$ and $(\tilde{\alpha}_{E_1}(Q), s\varphi^{\times 4}(Q))$ for any $P = (P_1, P_2, P_3, P_4)$ and $Q = (Q_1, Q_2, Q_3, Q_4)$ in $E_1^4[N']$. We have

$$\begin{aligned}
&e_{N',\lambda_1\times\lambda_2}((\tilde{\alpha}_{E_1}(P), s\varphi^{\times 4}(P)), (\tilde{\alpha}_{E_1}(Q), s\varphi^{\times 4}(Q))) \\
&= e_{N',\lambda_1}(\tilde{\alpha}_{E_1}(P), \tilde{\alpha}_{E_1}(Q)) \cdot e_{N',\lambda_2}(s\varphi^{\times 4}(P), s\varphi^{\times 4}(Q)) \\
&= e_{N'}(\tilde{\alpha}_{E_1}(P), \lambda_1\lambda_1^{-1}\hat{\alpha}_{E_1}\lambda_1(Q)) \cdot e_{N'}(P, \widehat{s\varphi^{\times 4}} \circ \lambda_2 \circ s\varphi^{\times 4}(Q)) \\
&= e_{N'}(\alpha_{E_1}\tilde{\alpha}_{E_1}(P), \lambda_1(Q)) \cdot e_{N'}(P, \lambda_1 \circ \widetilde{s\varphi^{\times 4}} \circ s\varphi^{\times 4}(Q)) \\
&= e_{N',\lambda_1}([m](P), Q) \cdot e_{N'}(P, \lambda_1([s^2n^2N]Q)) \\
&= e_{N',\lambda_1}(P, Q)^m \cdot e_{N',\lambda_1}(P, Q)^{s^2n^2N} \\
&= e_{N',\lambda_1}(P, Q)^{m+s^2n^2N} = 1, \text{ as } m + s^2m^2N \equiv 0 \mod N'.
\end{aligned}$$

Thus H is isotropic with respect to the product polarisation. Finally, it is also maximal since it has order N'^8 which is the square root of the order of $(E_1^4 \times E_2^4)[N']$, see [Mum70, p 233]. □

It is now possible to provide Algorithm 1 which efficiently divides isogenies by integers. This algorithm is similar to those presented by Robert in [Rob22a, 4. The algorithm] and the section 4 of [Rob22b].

Theorem 4. *Algorithm 1 is correct and runs in*

- $O(\max(M^2, D)B^8 \log^2(N')\log(B))$ *operations over $\mathbb{F}_{p^k}$,*
- *plus the cost of the factorisation of N',*
- *plus the cost of the computation of the bases of $E_1[\ell^e]$ for each prime power divisor ℓ^e of N',*
- *plus the cost of $O(\log N')$ evaluations of φ over these bases,*
- *plus the cost of decomposing $N' - N$ as a sum of four squares (which takes $O(\log^2 N')$ arithmetic operations over integers),*

where B, M, D give the following bounds $B \geq P^*(N'), M \geq \delta_E(N')$ and $D \geq \delta_{E,2}(N')$. Moreover, if φ/n is indeed an isogeny, the output representation of φ/n has the following properties:

Algorithm 1 `Isogeny division`

Input : An isogeny $\varphi : E_1 \to E_2$, where E_1, E_2 are elliptic curves defined over $\mathbb{F}_{p^k}$, and two integers n and $N' > \deg(\varphi)$ such that $(N', p\deg(\varphi)) = 1$

Output : A representation of φ/n if it is a well-defined isogeny, `False` otherwise.

Set $N \leftarrow \deg(\varphi)/n^2$.
if $N \notin \mathbb{N}$ **then**
    return `False`
Set $m \leftarrow N' - N$.
Decompose m as $m_1^2 + m_2^2 + m_3^2 + m_4^2$.
Set $M \leftarrow \begin{pmatrix} m_1 & -m_2 & -m_3 & -m_4 \\ m_2 & m_1 & m_4 & -m_3 \\ m_3 & -m_4 & m_1 & m_2 \\ m_4 & m_3 & -m_2 & m_1 \end{pmatrix}$.
Let α be the m-endomorphism over E_1^4 given by the matrix M.
Let $\tilde{\alpha}$ be the dual isogeny of α with respect to the product polarisation.
$s \leftarrow n^{-1} \mod N'$.
Compute a factorisation $\ell_1^{e_1} \dots \ell_r^{e_r}$ of N'.
Compute bases $(P_{1,i}, P_{2,i})$ of $E_1[\ell_i^{e_i}]$ for $i \in [\![1, r]\!]$.
Set $(P_1, P_2) \leftarrow (\sum_{i=1}^r P_{1,i}, \sum_{i=1}^r P_{2,i})$ a basis of $E_1[N']$.
Compute a representation of an N'-isogeny F of $E_1^4 \times E_2^4$ of kernel

$$\ker F = \{(\tilde{\alpha}_{E_1}(\tau_i(P_j)), s\varphi^{\times 4}(\tau_i(P_j)))|\forall i \in [\![1, 4]\!], \forall j \in \{1, 2\}\}.$$

if $F(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4}) \not\simeq (E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ **then**
    return `False`.
Set $\gamma : F(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4}) \to (E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ be an isomorphism of principally polarised abelian varieties.
if $E_1 \simeq E_2$ **then**
    Set $\psi_0 : E_2 \to E_1$ to be an isomorphism.
else
    Set $\psi_0 : E_2 \to E_1$ to be the zero map.
Compute the sets of maps $S_{E_1} := \mathrm{Aut}(E_1)\psi_0$ and $S_{E_2} := \mathrm{Aut}(E_2)$.
for $t \in [\![1, 8]\!]$ **do**
    for $\psi \in S_{E_1}$, if $1 \le t \le 8$, or $\psi \in S_{E_2}$, if $5 \le t \le 8$, **do**
        if $n(\psi^{-1} \circ \pi_t \circ \gamma \circ F \circ \tau_1(P_{i,j})) = \varphi(P_{i,j}), \forall i \in \{1, 2\}, \forall j \in [\![1, r]\!]$ **then**
            return The representation of φ/n induced by $\psi^{-1} \circ \pi_t \circ \gamma \circ F \circ \tau_1$.
return `False`

- *It has size* $O(kM\log(N')\log(p))$ *bits.*
- *It allows to evaluate* φ/n *in* $O(B^8 M\log(N')\log(B))$ *operations over the field of definition of the input.*

Proof. Let us prove the correctness of Algorithm 1.

First, by Lemma 6, $\ker F$ is always a maximal isotropic subgroup of $(E_1^4\times E_2^4)[N']$ and thus the isogeny F is well defined.

When $\varphi/n : E_1 \to E_2$ is an isogeny, we have that $(\varphi/n)|_{E[N']} = (s\varphi)|_{E[N']}$. Hence, by Lemma 4, F is isomorphic to an N'-isogeny that embeds φ/n. More precisely, $F(E_1^4\times E_2^4, \lambda_{E_1^4\times E_2^4}) \simeq (E_1^4\times E_2^4, \lambda_{E_1^4\times E_2^4})$ as principally polarised abelian varieties and for any isomorphism ψ between them, there exist an isomorphism $\psi : E_2 \to E'$, where $E' \in \{E_1, E_2\}$, and an integer $t \in [\![1, 8]\!]$ such that

$$\pi_t(\gamma(F(\tau_1(P)))) = \psi(\varphi/n)(P), \forall P \in E_1. \tag{4}$$

We check at line 15 if we can find such isomorphism γ. If this is the case, we fix an isomorphism $\gamma : F(E_1^4\times E_2^4, \lambda_{E_1^4\times E_2^4}) \to (E_1^4\times E_2^4, \lambda_{E_1^4\times E_2^4})$. Otherwise, by Lemma 4, φ/n is not an isogeny and the algorithm must return `False`.

We then look for an isomorphism ψ and an integer t verifying Equality (4). To satisfy the equality, ψ needs to have the same codomain as $\pi_t\circ\gamma\circ F$ which is equal to E_1 if $1 \le t \le 4$ and E_2 if $5 \le t \le 8$. In the first case, ψ is an element of $\mathrm{Aut}(E_1)\psi_0$, where $\psi_0 : E_2 \to E_1$ is an isomorphism. In the second case, ψ is an automorphism of E_2.

In the for loop, we search for a solution (ψ, t) of Equality (4) over the bases of $E_1[\ell_i^{e_i}], \forall i \in [\![1, r]\!]$. It is equivalent to checking Equality (4) over $E_1[N']$ as $(\ell_i, \ell_j) = 1, \forall i \ne j \in [\![1, r]\!]$. Moreover as $N' > \deg\varphi$ and $\deg\varphi \ge 2$, a solution of (4) over $E_1[N']$ is a solution over the entire elliptic curve E_1. Indeed, if two isogenies φ and φ' of same degree are equal over $E_1[N']$ with $N' > \deg\varphi \ge 2$, then they are equal everywhere. Let us assume that $\phi := \varphi - \varphi'$ is a non-zero isogeny. We have $\phi(E_1[N']) = 0$, hence

$$4\deg\varphi \le \deg\varphi^2 < {N'}^2 = \#E[N'] \le \deg\phi \le ((\deg\varphi)^{1/2} + (\deg\varphi')^{1/2})^2 = 4\deg\varphi,$$

this is a contradiction. Notice that we assumed that $\deg\varphi \ge 2$. In fact, we can even assume that $\deg\varphi \ge 4$, otherwise $\deg(\varphi)/n^2$ is not an integer when $n > 1$. Moreover, the division is trivial if n =1.

Since we are doing an exhaustive search at line 22, if φ/n is an isogeny, the algorithm will find an embedding representation $(F, 1, t)$ of φ/n up the two isomorphims ψ and γ. If no such coefficient of F is found, Lemma 4 implies that φ/n is not an isogeny.
The output representation of φ/n is then given by the composition of the representation of ψ^{-1} with the embedding representation $(\gamma\circ F, 1, t)$.

Let us now turn to the complexity analysis of the different steps. We consider the following bounds $B \ge P^*(N'), M \ge \delta_E(N'), D \ge \delta_{E,2}(N')$.

[1-9] The decomposition of m at the line 5 can be done in $O(\log^2 N')$ arithmetic operations over the integers, see [PT18]. This is the only complexity of the algorithm where operations are not counted over the finite field but over integers. The computational cost of the other lines is negligible compared to the rest of the algorithm.

[10 - 11] We do not estimate the complexity of these steps now, we simply acknowledge them in the overall analysis.

[12-16] At line 12, we denote a basis of $E_1[N']$ by (P_1, P_2) only formally to get simple notations. The computation are always done with the $(P_{1,i}, P_{2,i})$, where $i \in [\![1, r]\!]$.

By Lemma 5, getting a representation of the N'-isogeny F takes :

– $O(B^8 D \log^2(N')\log(B))$ arithmetic operations over $\mathbb{F}_{p^k}$,
– $O(\log N')$ evaluations of φ over the bases of $E[\ell_i^{e_i}]$, for $i \in [\![1, r]\!]$.

Then, it remains to compute an isomorphism γ between the codomain of F and $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$. One can perform an exhaustive search over the symplectic group $\mathrm{Sp}_{16}(\mathbb{Z}/4\mathbb{Z})$, which consists of the set of 16×16 matrices that preserve symplectic form, to find a matrix that sends the theta coordinates of the codomain of F to the theta coordinates of $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ (see Remark 2 for the construction of the latter). This search takes a constant time as describe in [DLRW24, Appendix F.3]. If no such linear transformation is found, then the two principally polarised abelian varities are not isomorphic.

Note that this method of finding the isomorphism is not suitable for deployment in cryptographic schemes due to its high computational cost. For practical usage, it is preferable to directly compute the matrix we are looking for (see [DLRW24, Appendinx F.3]).

[17 - 21] Thoses steps are done efficiently using basics of elliptic curves' theory, see [Sil86].

Checking if two elliptic curves are isomorphic can be done using the j-invariant. Then one can get an explicit isomorphism with a small computation from the Weierstrass equations of the elliptic curves.

The group of automorphisms of an elliptic curve E, for $p > 3$, is generated by

$$\begin{cases} \{(x,y) \mapsto (x,-y), (x,y) \mapsto (-x, iy)\} & \text{if } j(E) = 0, \\ \{(x,y) \mapsto (x,-y), (x,y) \mapsto (\zeta_3 x, y)\} & \text{if } j(E) = 1728, \\ \{(x,y) \mapsto (x,-y)\} & \text{otherwise,} \end{cases}$$

where i is a primitive 2-nd root of unity and ζ_3 is a primitive 3-rd root of unity, both in $\bar{\mathbb{F}}_p$. When $p = 2$ or 3, there are at most respectively 24 and 12 automorphisms. One can find their explicit description in [Sil86, Appendix A].

[22 - 26] The loop at line 22 has $O(\log N')$ iterations where the evaluations of $\tau_1, \gamma, \pi_t, \psi^{-1}$ are negligeable. Thus it takes in total $O(\log N')$ evaluations of φ over $E_1[\ell_i^{e_i}]$, $\forall i \in [\![1, r]\!]$, plus $O(B^8 M \log^2(N')\log(B))$ operations over extension of degree at most M to evaluate F.

We get the claimed complexity by summing all those steps.

In addition, the size of the ouput representation of φ/n is mainly the size of the representation of F thus it has size $O(kM \log(N')\log p)$ bits.

Finally, it allows to evaluate φ/n at a point in $O(B^8 M \log(N')\log(B))$ operations over its field of definition because all the computations are negligeable in comparison to the evaluation of F. □

When the input of Algorithm 1 is efficiently represented, it leads to Theorem 3 which concludes this section about efficient division of isogenies.

Proof of Theorem 3. To get this result, one only needs to find a suitable powersmooth integer N' and to take advantage of the efficient representation of φ. One can use the approach proposed in [Rob22b] to get such an integer N': we compute N' by multiplying successive primes, coprime to $p\deg(\varphi)$, until their product is greater than $\deg(\varphi)$. It takes $O(\log\deg(\varphi))$ arithmetic operations and gives an integer N' which is $O(\log\deg(\varphi))$-powersmooth and such that $\log N' = O(\log\deg(\varphi))$. Then, with the same notation as in Theorem 4, we have $M = B^2$ and $D = B^4$ which directly gives the claimed size of the representation of φ/n and also the complexity to evaluate it.

Finally, by construction of N' and because φ is efficiently represented, all the remaining costs of Algorithm 1 are polynomial in $\log p$, $\log\deg(\varphi)$. □

5 Solving Primitivisation

The Primitivisation problem has been introduced recently in [ACL+23] together with a quantum subexponential algorithm solving it. However, it can be seen as a generalisation of the important problem of computing the endomorphism ring of an ordinary elliptic curve. Indeed, for ordinary elliptic curves, the Frobenius endomorphism π is non-scalar, hence we always have an orientation by $\mathbb{Z}[\pi]$, and the endomorphism ring is a quadratic order containing $\mathbb{Z}[\pi]$. Therefore computing the endomorphism ring of an ordinary curve really is a case of the Primitivisation problem.

One initial idea to solve Primitivisation is to adapt the best algorithms solving the ordinary version of EndRing. Until recently, these algorithms had a subexponential complexity. Techniques involving higher dimensional isogenies changed the state of the art: in [Rob22b, Section 4], Robert shows how to compute the endomorphism ring of ordinary elliptic curves in polynomial time (when the factorisation of the discriminant of the Frobenius is known), essentially by *dividing* translates of the Frobenius.

In this section, we describe how Robert's method can be adapted to solve the Primitivisation Problem and we give, as a direct consequence of this result, a polynomial algorithm to compute action of smooth ideals.

First, Theorem 5 and its proof describe the algorithm and its complexity without assuming anything on the representation of the input endomorphism. Notice that it requires computations only over a large enough torsion subgroup. Hence, the complexity depends on the degree of the extension where this torsion lives and on the difficulty to evaluate the endomorphism on it. Then, Corollary 2 specifies this theorem to the case where the endomorphism is given in efficient representation. The two results assume that the factorisation of the discriminant of the order generated by the endomorphism is known.

Theorem 5 (Primitivisation)**.** *There exists an algorithm that takes as input:*

- *A supersingular elliptic curve E defined over a finite field $\mathbb{F}_{p^2}$,*
- *An endomorphism $\theta \in \mathrm{End}(E) \setminus \mathbb{Z}$ of degree N together with the factorisation of $\mathrm{disc}(\mathbb{Z}[\theta])$,*
- *An integer $N' > N$ such that $(N', pN) = 1$ with three bounds $B \geq P^*(N')$, $M \geq \delta_E(N')$ and $D \geq \delta_{E,2}(N')$,*

and returns a primitive orientation $\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$ with $\mathfrak{O} \supseteq \mathbb{Z}[\theta]$ such that

- *The orientation ι takes $O(M \log(N') \log(p))$ bits to store,*
- *The endomorphism $\iota(\omega)$ can be evaluated at a point in $O(B^8 M \log(N') \log(B))$ operations over its field of definition.*

This algorithm runs in $O(\max(M^2, D) B^8 \log^2(N') \log(N) \log(B))$ operations over $\mathbb{F}_{p^2}$, plus the cost of the computation of the bases $E[\ell^e]$ for each prime power divisor ℓ^e of N' plus the cost of the computation of $O(\log N')$ evaluations of θ over these bases plus the cost of decomposing $N' - N$ as a sum of four squares.

Proof. Let $\alpha \in \bar{\mathbb{Q}}$ be a root of the minimal polynomial of θ and $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$ be the orientation defined by $\iota(\alpha) = \theta$. Let $K = \mathbb{Q}(\alpha)$, f_α be the conductor of the order $\mathbb{Z}[\alpha]$ and O_K be the ring of integers K. The factorisation of the conductor f_α can be deduced from the known factorisation of $\mathrm{disc}(\mathbb{Z}[\theta])$. Indeed, let Δ_K be the discriminant of K which is given by

$$\Delta_K = \begin{cases} d, & \text{if } d \equiv 1 \mod 4 \\ 4d, & \text{otherwise,} \end{cases}$$

where d is the squarefree part of $\mathrm{disc}(\mathbb{Z}[\alpha])$. The integer d is easy to compute since we have the factorisation of $\mathrm{disc}(\mathbb{Z}[\theta]) = \mathrm{disc}(\mathbb{Z}[\alpha])$. As $f_\alpha^2 = \mathrm{disc}(\mathbb{Z}[\alpha])/\Delta_K$ one can directly deduce the factorisation of f_α.

Let $\mathfrak{O} \subseteq O_K$ be the largest order such that ι extends to an embedding $\mathfrak{O} \hookrightarrow \mathrm{End}(E)$. That embedding is the primitivisation of ι, so the algorithm aims at determining $\mathfrak{O}$. The inclusions $\mathbb{Z}[\alpha] \subseteq \mathfrak{O} \subseteq O_K$ suggest that $\mathfrak{O}$ can be determined by starting from $\mathbb{Z}[\alpha]$, and testing if the orientation can be extended locally at each prime factor of the conductor, as in the computation of the endomorphism ring of ordinary elliptic curves (see [Rob22b]). This is described in Algorithm 2.

Algorithm 2 Primitivisation

Input : E a supersingular elliptic curve, $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$ an orientation such that $\iota(\alpha) = \theta$ is a non scalar endomorphism of degree N, an integer N' such that $N' > N$ and $(N', pN) = 1$ and the factorisation of f_α the conductor of $\mathbb{Z}[\alpha]$.

Output : A pair (α', θ') describing the primitivisation of ι.

$t \leftarrow \bar{\alpha} + \alpha$.
$\alpha' \leftarrow 2\alpha - t$. ▷ $\mathbb{Z}[\alpha'] = \mathbb{Z}[2\alpha]$.
$\theta' \leftarrow 2\theta - [t]$.
$(\ell_i)_{i=1}^n \leftarrow$ the list of distinct prime factors of $2f_\alpha$.
for $i \in [\![1, n]\!]$ **do**
    while $\theta'/\ell_i \in \mathrm{End}(E)$ **do** ▷ using Algorithm 1 with the input (θ', ℓ_i, N').
        $\alpha' \leftarrow \alpha'/\ell_i$.
        $\theta' \leftarrow \theta'/\ell_i$.
if $(\theta' + 1)/2 \in \mathrm{End}(E)$ **then** ▷ using Algorithm 1 with the input $(\theta' + 1, 2, N')$.
    $(\alpha', \theta') \leftarrow ((\alpha' + 1)/2, (\theta' + 1)/2)$.
return (α', θ').

Let us prove that Algorithm 2 is correct. Write $t = \alpha + \bar{\alpha}$. Since $\mathrm{disc}(\mathbb{Z}[\alpha]) = t^2 - 4\alpha\overline{\alpha}$, we have

$$\alpha' := 2\alpha - t = \pm\sqrt{\mathrm{disc}(\mathbb{Z}[\alpha])} = \pm f_\alpha \sqrt{\Delta_K}, \text{ where } \Delta_K \text{ is the discriminant of } K.$$

Also define $\theta' := 2\theta - [t]$. Note that for any divisor $m \mid f_\alpha$, we have $\mathbb{Z}[(f_\alpha/m)\sqrt{\Delta_K}] \subseteq \mathfrak{O}$ if and only if $\alpha'/[m] \in \mathrm{End}(E)$. The for-loop of Algorithm 2 finds the largest such integer m, hence the resulting pair $(\alpha'/m, \theta'/m)$ satisfies $\mathbb{Z}[\alpha'/m] = \mathfrak{O} \cap \mathbb{Z}[\sqrt{\Delta_K}]$.

The case $\ell_i = 2$ and the final if-statement account for the fact that $\mathbb{Z}[\sqrt{\Delta_K}]$ is not the maximal order: it has conductor 2 (we have $O_K = \mathbb{Z}[\sqrt{\Delta_K}/2]$ if $\Delta_K \equiv 0 \bmod 4$ and $O_K = \mathbb{Z}[(\sqrt{\Delta_K} + 1)/2]$ if $\Delta_K \equiv 1 \bmod 4$). That final correction accounted for, we actually obtain $\mathbb{Z}[\alpha'] = \mathfrak{O}$.

Let us now describe the complexity of Algorithm 2.

Since $f_\alpha \leq 4N(\alpha) = 4N$, there are $O(\log(N))$ divisions using Algorithm 1. By Theorem 4, both checking if $\theta'/p_i \in \mathrm{End}(E)$ and getting a representation of the new endomorphism can be done in $O(\max(D, M^2)B^8 \log^2(N') \log(B))$ operations over $\mathbb{F}_{p^2}$ plus the cost of the computation of the bases $E[\ell^e]$ for each prime power divisor ℓ^e of N' plus the cost of the computation of $O(\log N')$ evaluations of θ' over these bases plus the cost of decomposing $N' - N$ as a sum of four squares.

After each update of the generating endomorphism using this theorem, the length of the representation will be in $O(M \log(N') \log(p))$ bits and will allow to evaluate a point in $O(B^8 M \log(N') \log(B))$ operations over the field of definition of the input. Hence, all the divisions after the first one will run in $O(\max(M^2, D)B^8 \log^2(N') \log(B))$ operations over

$\mathbb{F}_{p^2}$ and the final output will have the claimed properties.

It leads to a global complexity in $O(\max(M^2, D)B^8 \log^2(N') \log(N) \log(B))$ operations over $\mathbb{F}_{p^2}$ plus the cost of the computation of the torsion group bases and the $O(\log N')$ evaluations of θ over them plus the cost of the decomposing of $N' - N$ as a sum of four squares.

□

Corollary 2 demonstrates that PRIMITIVISATION can be solved in polynomial time when the input is efficiently represented by applying Theorem 5.

Corollary 2. *There exists an algorithm that takes as input:*

- *A supersingular elliptic curve E defined over a finite field $\mathbb{F}_{p^2}$,*
- *An endomorphism $\theta \in \mathrm{End}(E) \setminus \mathbb{Z}$ of degree N together with the factorisation of $\mathrm{disc}(\mathbb{Z}[\theta])$,*

and returns an efficiently represented primitive orientation $\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$ with $\mathfrak{O} \supseteq \mathbb{Z}[\theta]$ such that the orientation ι takes $O(\log^3(N) \log(p))$ bits to store. This algorithm runs in time polynomial in $\log N$ and $\log p$.

Proof. As for Theorem 3, this corollary is obtained by computing a suitable powersmooth N' and by taking $M = B^2$ and $D = B^4$. One also needs to use the fact that we are dealing with efficiently represented isogenies. □

5.1 A direct application: Computing the action of smooth ideals

The class group action over $SS_{\mathfrak{O}}(p)$ is a key notion of the orientations' theory and is a central ingredient of the algorithms presented in Section 6 to solve the oriented problems we are interested in, such as the $\mathfrak{O}$-VECTORISATION problem.

In the previous state of the art, the classical method to evaluate the action of an ideal $\mathfrak{a}$ on an $\mathfrak{O}$-oriented elliptic curve (E, ι) was to factor $\mathfrak{a}$ into primes ideals and then to apply the action of each factor ideals successively before dividing by the degree. This method takes a time polynomial in the largest prime power factor of the norm of $\mathfrak{a}$ and in length of the input. The quality of the representation obtained "degrades" as actions are computed (so applying the action of several powersmooth ideal iteratively would actually take exponential time).

We emphasise that the powersmooth constraint and the progressive "degradation" of the representations come from the division by the norm of $\mathfrak{a}$ in the computation of the induced orientation

$$\varphi_{\mathfrak{a}*}(\iota)(\omega) = (\varphi_{\mathfrak{a}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{a}}) \otimes \frac{1}{N(\mathfrak{a})}, \text{where } \omega \text{ is a generator of } \mathfrak{O}.$$

Indeed, an efficient representation of the induced orientation times the norm of $\mathfrak{a}$ is given by the composition of the efficient representations of $\varphi_{\mathfrak{a}}, \iota(\omega)$ and $\hat{\varphi}_{\mathfrak{a}}$ as

$$N(\mathfrak{a})\varphi_{\mathfrak{a}*}(\iota) = \varphi_{\mathfrak{a}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{a}},$$

where efficient representations of $\varphi_{\mathfrak{a}}$ and $\hat{\varphi}_{\mathfrak{a}}$ can be obtained in time polynomial in a smooth bound of the norm of $\mathfrak{a}$. This representation now "degrades" linearly with the number of successive actions and there are no more powersmooth constraints. The issue with this division-free computation is that the induced orientation by the order $\mathbb{Z} + N(\mathfrak{a})\mathfrak{O}$ is obviously not primitive anymore, as it can still be divided by the norm of $\mathfrak{a}$. Thanks to Corollary 2, primitivising this orientation can be done efficiently given the factorisation of the conductor of the order, i.e. the factorisation of $N(\mathfrak{a})$ here. This provides a new polynomial algorithm to compute action of smooth ideals.

Corollary 3. *Let (E, ι) be an $\mathfrak{O}$-oriented supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. Let $\mathfrak{a}$ be an invertible $\mathfrak{O}$-ideal of B-smooth norm. If (E, ι) is efficiently represented, then one can compute an efficient representation of $[\mathfrak{a}] \star (E, \iota)$ in time polynomial in B, $\log p$ and in the length of the representation of ι.*

Proof. The prime factorisation $\ell_1^{e_1} \dots \ell_m^{e_m}$ of the norm of $\mathfrak{a}$ can be computed in time polynomial in B. From this factorisation, one deduces the decomposition of $\mathfrak{a}$ as a product of e_1 prime ideals of norm ℓ_1 with e_2 prime ideals of norm ℓ_2 and so on. Then, using a process similar to the CSIDH evaluation algorithm, generalised to a arbitrary orientations as in [CK20], one computes an efficient representation of $\varphi_{\mathfrak{a}}$ and $\hat{\varphi}_{\mathfrak{a}}$ in time polynomial in B, $k \log p$ and in the length of the representation of ι. Finally, one just has to primitivise the orientation $\varphi_{\mathfrak{a}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{a}}$ using Corollary 2 to get an efficient representation of $\mathfrak{a} \star (E, \iota)$ in polynomial time. □

Remark 4. We remind the reader that the authors' objective in this article is to develop rigorous algorithms under GRH. In this section, our aim is to provide an algorithm solving Primitivisation and to give applications of this resolution, such as computing smooth ideal actions with better complexity than the current state of the art without relying on any heuristics. Nevertheless, one might also be interested in using higher dimensional isogenies as a practical approach to computing group actions induced by oriented elliptic curves, even at the cost of relying on heuristics. This direction is studied, for example, in SCALLOP-HD [CLP24].

To compute the action of any ideal, we shall use the polynomial time `Clapoti` algorithm introduced in [PR23] to avoid the smoothness constraint.

Proposition 4 ([PR23, Theorem 2.9])**.** *Let (E, ι) be an $\mathfrak{O}$-oriented supersingular elliptic curve defined over $\mathbb{F}_{p^2}$ with $\mathfrak{O}$ a quadratic imaginary order of discriminant Δ. Given an integral invertible $\mathfrak{O}$-ideal $\mathfrak{a}$, one can compute an efficient representation of $\varphi_{\mathfrak{a}} : E \to E_{\mathfrak{a}}$ in probabilistic time polynomial in $\log |\Delta|$, $\log p$ and in $\log N(\mathfrak{a})$.*

Corollary 4. *Let (E, ι) be an $\mathfrak{O}$-oriented supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. If (E, ι) is efficiently represented, then one can compute an efficient representation of $[\mathfrak{a}] \star (E, \iota)$ in time polynomial in $\log p$, in $\log N(\mathfrak{a})$ and in the length of the representation of ι.*

Remark 5. The first version of the present article, which appeared prior to `Clapoti`, presented an algorithm for computing the action of any ideal with a rigorously proven complexity under GRH. This complexity matched the previous state of the art while removing all heuristic assumption that were previously required. The approach involved computing a smooth representative of the input ideal using algorithms with a subexponential complexity proven under GRH, such as [CJS14, Algorithm 1], followed by the application of Corollary 3.

6 Resolution of $\mathfrak{O}$-Vectorisation and α-EndRing

In this section, we prove, under GRH only, the complexity of a classical and a quantum resolution of $\mathfrak{O}$-Vectorisation which are as good as the current best algorithms based on heuristics. We then use these rigorous solutions to solve the α-EndRing problem.

6.1 Classical algorithm

Currently, the best complexity we can expect for a classical algorithm solving the $\mathfrak{O}$-Vectorisation problem is $l^{O(1)} |\operatorname{disc}(\mathfrak{O})|^{1/4}$, with l the length of the input, for instance with a meet-in-middle approach as in [DG16]. Before the results presented in this paper,

such complexity analyses were based on heuristics. Indeed, one needs to compute multiple actions of ideals to solve $\mathfrak{O}$-Vectorisation and without using higher dimensional isogenies to compute efficiently smooth ideal actions, one could only handle powersmooth ideals. Thus, one had to assume some heuristics about the distribution of powersmooth ideals. Thanks to Corollary 3, it is now possible to get rid of the constraint on powersmoothness and to rigorously prove this complexity.

To solve $\mathfrak{O}$-Vectorisation, we first study the Effective $\mathfrak{O}$-Vectorisation problem where one also asks the $\mathfrak{O}$-ideal to send the orientation of the first $\mathfrak{O}$-oriented elliptic curve to the orientation of the second one. Moreover, we want to be able to evaluate the isogeny induced by this ideal on another given $\mathfrak{O}$-orientable elliptic curve. Notice that $\mathfrak{O}$-Vectorisation and Effective $\mathfrak{O}$-Vectorisation are in fact both equivalent to $\mathfrak{O}$-EndRing, see [Wes22a].

Problem 7 (Effective $\mathfrak{O}$-Vectorisation)**.** *Given $(E,\iota),(E',\iota'),(F,\jmath)$ in $SS_{\mathfrak{O}}(p)$, find an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $\mathfrak{a}\star(E,\iota)\simeq(E',\iota')$, and an efficient representation of $\varphi_{\mathfrak{a}}:(F,\jmath)\to\mathfrak{a}\star(F,\jmath)$.*

Algorithm 3 almost solves Effective $\mathfrak{O}$-Vectorisation — it does not give an efficient representation and it only handles the case where (E,ι) and (E',ι') are in the same orbit, i.e. when there exists an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $\mathfrak{a}\star(E,\iota)=(E',\iota')$. This algorithm follows a meet-in-the-middle approach, namely successive actions of $\mathfrak{O}$-ideals are computed on (E,ι) and (E',ι') until a collision is found.

Algorithm 3 Almost Effective $\mathfrak{O}$-Vectorisation

Input : $(E,\iota),(E',\iota')\in SS_{\mathfrak{O}}(p)$ two efficiently represented oriented elliptic curves in the same orbit and a real $\varepsilon>0$.

Output : A $\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$-smooth $\mathfrak{O}$-ideal with at most $2\lceil\log|\operatorname{disc}\mathfrak{O}|\rceil$ prime factors which sends (E,ι) to (E',ι').

$x\leftarrow\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$.
$\Sigma_x\leftarrow\{[\mathfrak{p}]\in\operatorname{Cl}(\mathfrak{O})$, such that $\gcd(f_{\mathfrak{O}},\mathfrak{p})=1$ and $N(\mathfrak{p})\leq x$ prime$\}$.
$S_x\leftarrow\Sigma_x\cup\{[\mathfrak{p}]^{-1}$ for $[\mathfrak{p}]\in\Sigma_x\}$.
$T[\mathtt{enc}((E,\iota))]\leftarrow(1)$.
while $\#T<\sqrt{\#\operatorname{Cl}(\mathfrak{O})}$ **do**
    $y\leftarrow\operatorname{Unif}\{y\in\mathbb{N}^{\#S_x}$ such that $||y||_1=\lceil\log|\operatorname{disc}\mathfrak{O}|\rceil\}$.
    $\mathfrak{a}\leftarrow S_x^y$.
    if $T[\mathtt{enc}(\mathfrak{a}\star(E,\iota))]$ is empty **then**
        $T[\mathtt{enc}(\mathfrak{a}\star(E,\iota))]\leftarrow\mathfrak{a}$.
$\mathfrak{a}\leftarrow(1)$.
while $T[\mathtt{enc}(\mathfrak{a}\star(E',\iota'))]$ is empty **do**
    $y\leftarrow\operatorname{Unif}\{y\in\mathbb{N}^{\#S_x}$ such that $||y||_1=\lceil\log|\operatorname{disc}\mathfrak{O}|\rceil\}$.
    $\mathfrak{a}\leftarrow S_x^y$.
return $\bar{\mathfrak{a}}T[\mathtt{enc}(\mathfrak{a}\star(E',\iota'))]$.

Lemma 7 (GRH). *Algorithm 3 runs in expected time $l^{O_\varepsilon(1)}|\operatorname{disc}(\mathfrak{O})|^{1/4}$ where l is the length of the input, and is correct.*

Proof. First of all, notice that using a dictionary structure for the table T, one can add and search for elements in time $O(\log\#T)$. From [DFDdSGF+21, Section 5.3], we have the estimate $\#\operatorname{Cl}(\mathfrak{O})=O(\log(|\operatorname{disc}(\mathfrak{O})|)\sqrt{|\operatorname{disc}(\mathfrak{O})|})$. Then insertions and searches in the table T can be done in $O(\log|\operatorname{disc}(\mathfrak{O})|)$. Moreover, we use the `enc` function, see Section 2.3, to have a unique encoding of oriented elliptic curves.

[1-4] Those steps are polynomial in $\log^{2+\varepsilon} \operatorname{disc}(\mathfrak{O})$.

[5-9] It is expected that this first while loop will end after $O(\sqrt{\#\operatorname{Cl}(\mathfrak{O})})$ iterations. Indeed, by Proposition 1, one can expect to add a new element to the table T after at most 2 draws of random smooth ideals.

By Corollary 3, computing an efficient representation of $\mathfrak{a}\star(E,\iota)$ is done in polynomial time in l_1, in $\log p$ and in $\log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|$, where l_1 is the length of the representation of ι.

[11-14] This while loop is also expected to end after $O(\sqrt{\#\operatorname{Cl}((\mathfrak{O})})$ iterations, since, thanks again to Proposition 1, each iteration has a probability of success greater than $\frac{1}{2\sqrt{\#\operatorname{Cl}(\mathfrak{O})}}$.

Moreover, as in the first loop, using Corollary 3, one can compute the action of $\mathfrak{a}$ in time polynomial in l_2, $\log p$ and $\log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|$, where l_2 is the length of the representation of ι'.

This leads to a global runtime in $(\max(l_1,l_2)\log p \log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|)^{O(1)}\sqrt{\#\operatorname{Cl}(\mathfrak{O})}$. Thanks again to the estimate $\#\operatorname{Cl}(\mathfrak{O}) = O(\log(|\operatorname{disc}(\mathfrak{O})|)\sqrt{|\operatorname{disc}(\mathfrak{O})|})$, we get the claimed complexity.

The correctness of the algorithm is given by a short computation. By construction, the output $\mathfrak{O}$-ideal $\mathfrak{a}$ verifies

$$T[\mathtt{enc}(\mathfrak{a}\star(E',\iota'))]\star(E,\iota) \simeq \mathfrak{a}\star(E',\iota').$$

Hence,

$$\begin{aligned}(\bar{\mathfrak{a}}T[\mathtt{enc}(\mathfrak{a}\star(E',\iota'))])\star(E,\iota) &= \bar{\mathfrak{a}}\star(T[\mathtt{enc}(\mathfrak{a}\star(E',\iota'))]\star(E,\iota))\\ &\simeq \bar{\mathfrak{a}}\star(\mathfrak{a}\star(E',\iota')) = (\bar{\mathfrak{a}}\mathfrak{a})\star(E',\iota') = (E',\iota).\end{aligned}$$

Finally, the output ideal is a product of two $\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$-smooth $\mathfrak{O}$-ideals with at most $\lceil\log|\operatorname{disc}(\mathfrak{O})|\rceil$ prime factors thus it is a $\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$-smooth $\mathfrak{O}$-ideal with at most $2\lceil\log|\operatorname{disc}(\mathfrak{O})|\rceil$ prime factors. □

Remark 6. Algorithm 3 needs space exponential in the length of the input. A space-efficient algorithm is conceivable using a Pollard-ρ approach, as it is used to find isogenies between ordinary elliptic curves in [BS12]. A rigorous analysis of such algorithms typically requires access to a random oracle, and we do not pursue this direction here.

Algorithm 3 is a central subprocedure in our classical resolution of $\mathfrak{O}$-Vectorisation and α-EndRing, as well as Effective $\mathfrak{O}$-Vectorisation. These applications of Algorithm 3 require to move from one orbit to the other using the $\mathfrak{O}$-twists.

Theorem 6 (**GRH**, Effective $\mathfrak{O}$-Vectorisation)**.** *There is a classical algorithm taking as input three oriented elliptic curves $(E,\iota),(E',\iota')$ and $(F,\jmath)$ in $SS_{\mathfrak{O}}(p)$ and a real number $\varepsilon>0$ which returns an $\mathfrak{a}$-ideal $\lceil\log^{2+\varepsilon}\rceil$-smooth such that $E^{\mathfrak{a}}\sim E'$ together with a representation of $\varphi_{\mathfrak{a}}:(F,\jmath)\to\mathfrak{a}\star(F,\jmath)$ in expected time $l^{O_\varepsilon(1)}|\operatorname{disc}(\mathfrak{O})|^{1/4}$ where l is the length of the input. The returned representation of $\varphi_{\mathfrak{a}}$ is given by $O(\log|\operatorname{disc}(\mathfrak{O})|)$ isogeny kernels of order at most $\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$.*

Proof. Suppose we are given some positive real ε and two oriented supersingular elliptic curves $(E,\iota)\not\simeq(E',\iota')\in SS_{\mathfrak{O}}(p)$, where $\mathfrak{O}$ is an order of some quadratic field K. First, we check if p is inert or ramified in K. Recall that p does not split over K otherwise $SS_{\mathfrak{O}}(p)$ would be empty [Onu21, Proposition 3.2].

By [ACL+24, Theorem 4.4], if p is ramified in K, then the action of $\mathrm{Cl}(\mathfrak{O})$ has only one orbit. Thus by running Algorithm 3 with the inputs $(E,\iota),(E',\iota')$ and ε, we get an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $\mathfrak{a}\star(E,\iota)\simeq(E',\iota')$.

Otherwise, if p is inert in K, again by [ACL+24, Theorem 4.4], the action of $\mathrm{Cl}(\mathfrak{O})$ has two orbits. We then run two instances of Algorithm 3 in parallel, the first one with the inputs $(E,\iota),(E',\iota')$ and ε and the second one with the inputs $(E,\bar{\iota}),(E',\iota')$ and ε. We know that (E,ι) and its $\mathfrak{O}$-twist $(E,\bar{\iota})$ are not in the same orbit, see [Onu21], thus only one procedure will stop. If it is the instance having (E,ι) as input, that means that we find an $\mathfrak{O}$-ideal $\mathfrak{a}$ sending (E,ι) to (E',ι'). Else, it means that $(E,\bar{\iota})$ and (E',ι') are in the same orbit. Hence (E,ι) is not on the same orbit as (E',ι') and there is no solution to the Effective $\mathfrak{O}$-Vectorisation problem. In this case, we return `False`.

Now we have an ideal $\mathfrak{a}$ solving our Effective $\mathfrak{O}$-Vectorisation instance, it remains to compute an efficient representation of the isogeny $\varphi_{\mathfrak{a}}$. Since $\mathfrak{a}$ has been returned by Algorithm 3 it is a $\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$-smooth $\mathfrak{O}$-ideal with at most $2\log|\operatorname{disc}(\mathfrak{O})|$ prime factors. Then using Corollary 3 an efficient representation of $\varphi_{\mathfrak{a}}$ can be computed in time polynomial in $\log p$, $\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$ and in the length of the representation of ι. □

Theorem 7 (**GRH**, Classical $\mathfrak{O}$-Vectorisation)**.** *There is a classical algorithm taking as input two oriented elliptic curves (E,ι) and (E',ι') in $SS_{\mathfrak{O}}(p)$ and a real number $\varepsilon>0$ which returns an $\mathfrak{O}$-ideal $\mathfrak{a}$ of $\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$-smooth norm such that $E^{\mathfrak{a}}\sim E'$ in expected time $l^{O\varepsilon(1)}|\operatorname{disc}(\mathfrak{O})|^{1/4}$ where l is the length of the input.*

Proof. We know that the action of $\mathrm{Cl}(\mathfrak{O})$ on $SS_{\mathfrak{O}}(p)$ has at most 2 orbits, see Proposition 2. Let O be the orbit of (E',ι'). From the same proposition, we know that (E,ι) or its $\mathfrak{O}$-twist $(E,\bar{\iota})$ is in O. Thus, by running two instances of Algorithm 3 until one ends, the first taking as input the oriented elliptic curves (E,ι) and (E',ι') and the second taking $(E,\bar{\iota})$ and (E',ι'), we make sure that we find a suitable ideal in an expected time given by Lemma 7. □

From the previous results, we can now prove Theorem 1.

Proof of Theorem 1. Let $E\in\widetilde{SS}_{\mathfrak{O}}(p)$ be a primitively $\mathfrak{O}$-orientable elliptic curve and $\iota:\mathbb{Z}[\alpha]\hookrightarrow\mathrm{End}(E)$ be an orientation of E such that $\mathbb{Z}[\alpha]\subseteq\mathfrak{O}$. Let us prove that the computation of the endomorphism ring $\mathrm{End}(E)$ can be done in probabilistic time $l^{O(1)}|\operatorname{disc}(\mathbb{Z}[\alpha])|^{1/4}$, where l is the length of the input.

First we compute the factorisation of $\operatorname{disc}(\mathbb{Z}[\alpha])$ in time subexponential in $\log|\operatorname{disc}(\mathbb{Z}[\alpha])|$. Then, by Corollary 2, we can compute, in probabilistic time polynomial in the length of the input, the primitive orientation $\jmath$ such that $(E,\jmath)\in SS_{\mathfrak{O}}(p)$. This reduces the computation of $\mathrm{End}(E)$ to the instance of $\mathfrak{O}$-EndRing given by $(E,\jmath)$ which, in turn, reduces in probabilistic polynomial time to an instance of $\mathfrak{O}$-Vectorisation by Proposition 3. Finally, by Theorem 7 and since $|\operatorname{disc}(\mathbb{Z}[\alpha])|$ is greater than $|\operatorname{disc}(\mathfrak{O})|$, the $\mathfrak{O}$-Vectorisation problem can be solved in $l^{O(1)}|\operatorname{disc}(\mathbb{Z}[\alpha])|^{1/4}$. □

6.2 Quantum algorithm

The subexponential quantum resolution of the $\mathfrak{O}$-Vectorisation proven in this section is based on the work of Childs, Jao and Soukharev to construct an isogeny between two given isogenous ordinary elliptic curves, [CJS14]. In particular, we use the fact that given two oriented elliptic curves $(E_0,\iota_0),(E_1,\iota_1)\in SS_{\mathfrak{O}}(p)$ in the same orbit, finding an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $\mathfrak{a}\star(E_0,\iota_0)=(E_1,\iota_1)$ can be viewed as an instance of the Hidden Shift problem.

Problem 8 (Hidden Shift)**.** *Given a finite abelian group $(A,+)$, a finite set $S\subset\{0,1\}^m$ of encoding length m and two black-box functions $f_0,f_1:A\to S$ where f_0 is injective and*

such that there exists an element $s \in S$ verifying $f_1(x) = f_0(s + x)$ for any $x \in S$, find the element s called the shift hidden by f_0 and f_1.

In this paper, we assume that the abelian group A of any instance of `Hidden shift` is always given as $\mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_k\mathbb{Z}$ for some integers $k, n_1, \dots n_k$. Notice that the Hidden Shift problem can also be considered when A is not abelian. Nevertheless the above formulation of the problem allows us to use Kuperberg's quantum algorithm to solve it in a subexponential number of queries of the black-box functions f_0 and f_1.

Theorem 8 (Theorem 7.1. [Kup05])**.** *There is a quantum algorithm such that the `Hidden Shift problem` for abelian groups can be solved with time and query complexity $2^{O(\sqrt{\log n})}$, where n is the size of the abelian group, uniformly for all finitely generated abelian groups.*

To solve quantumly $\mathfrak{O}$-Vectorisation, we first prove the correctness and the expected subexponential runtime of Algorithm 4 which solves $\mathfrak{O}$-Vectorisation assuming that the two input curves are in the same orbit. This algorithm is analogous to [CJS14, Algorithm 3].

Algorithm 4 Quantum $\mathfrak{O}$-Vectorisation in the same orbit

Input : $(E_0, \iota_0), (E_1, \iota_1) \in SS_{\mathfrak{O}}(p)$ two oriented elliptic curves in the same orbit.
Output : a reduced $\mathfrak{O}$-ideal $\mathfrak{a} \in \mathrm{Cl}(\mathfrak{O})$ such that $\mathfrak{a} \star (E_0, \iota_0) = (E_1, \iota_1)$ and the isogeny $\varphi_{\mathfrak{a}} : (E_0, \iota_0) \to (E_1, \iota_1)$.

1: Compute $\mathrm{Cl}(\mathfrak{O})$ as a decomposition $\langle [\mathfrak{b}_1] \rangle \oplus \cdots \oplus \langle [\mathfrak{b}_k] \rangle$.
2: Denote by n_j the order of $\langle [\mathfrak{b}_j] \rangle$, for $j \in [\![1, \dots, k]\!]$.
3: Solve the `Hidden Shift` problem instance given with the black-box functions, for $j \in \{0, 1\}$,

$$f_j : \mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_k\mathbb{Z} \to \mathtt{enc}(SS_{\mathfrak{O}}(p)), (x_1, \dots, x_k) \mapsto \mathtt{enc}((\mathfrak{b}_1^{x_1} \dots \mathfrak{b}_k^{x_k}) \star (E_j, \iota_j))$$

where $s = (s_1, \dots, s_k)$ denoted the hidden shift.
4: Compute $\mathfrak{a}$ the reduced representative of the ideal class $[\mathfrak{b}_1^{s_1} \dots \mathfrak{b}_k^{s_k}]$.
5: Compute the isogeny $\varphi_{\mathfrak{a}}$ induced by the ideal $\mathfrak{a}$.
6: **return** $\mathfrak{a}$ and $\varphi_{\mathfrak{a}}$.

Lemma 8 (**GRH**)**.** *The Algorithm 4 is correct and runs in expected time $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$ where l is the length of the input.*

Proof. Let us prove the complexity of Algorithm 4:

[1] Under GRH, one can quantumly compute the group structure of $\mathrm{Cl}(\mathfrak{O})$ in time polynomial in $\log|\operatorname{disc}(\mathfrak{O})|$, using for instance [BS16, Theorem 1.2].

[3] By Kuperberg's algorithm, Theorem 8, one can solve the instance of the `Hidden Shift` problem in $L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$ queries on the black-box functions, all of which are computed in time polynomial in the length of the input using `Clapoti`, see Corollary 4. Thus this step in done in $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$, where l is the length of the input.

[4] To compute the reduced representative of the ideal class $[\mathfrak{b}_1^{s_1} \dots \mathfrak{b}_k^{s_k}]$, we use a square-and-multiply approach where the ideal computed at each step is reduced. With this method, $\forall i \in [\![1, k]\!]$, $[\mathfrak{b}_i^{s_i}]$ can be reduced in $O(\lceil \log \#\,\mathrm{Cl}(\mathfrak{O}) \rceil)$ squarings, multiplications and reductions which all can be done in polynomial time in $\log|\operatorname{disc}\mathfrak{O}|$. Then it only remains to compute the reduced representative of $[\mathfrak{b}_1^{s_1} \dots \mathfrak{b}_k^{s_k}]$ from the reduced representatives of $[\mathfrak{b}_1^{s_1}], \dots, [\mathfrak{b}_k^{s_k}]$ in time polynomial in $\log|\operatorname{disc}\mathfrak{O}|$. Hence, using the standard result $\#\,\mathrm{Cl}(\mathfrak{O}) = O(\log(|\operatorname{disc}(\mathfrak{O}))\sqrt{\operatorname{disc}(\mathfrak{O})})$, this whole step is done in time polynomial in $\log|\operatorname{disc}(\mathfrak{O})|$.

[5] Finally with `Clapoti`, we can compute the isogeny $\varphi_{\mathfrak{a}} : (E_0, \iota_0) \to (E_1, \iota_1)$ in time polynomial in the length of the input.

A short computation proves that the shift $s = (s_1, \dots, s_k)$ hidden by f_0 and f_1 gives the ideal class $[\mathfrak{a}] = [\mathfrak{b}_1^{s_1} \dots \mathfrak{b}_k^{s_k}]$ such that $\mathfrak{a} \star (E_0, \iota_0) = (E_1, \iota_1)$. Indeed, for every $[\mathfrak{b}] \in \mathrm{Cl}(\mathfrak{O})$, there is a vector $b = (b_1, \dots, b_k) \in \mathbb{Z}/n_1\mathbb{Z} \times \dots \times \mathbb{Z}/n_k\mathbb{Z}$ such that $[\mathfrak{b}] = [\mathfrak{b}_1^{b_1} \dots \mathfrak{b}_k^{b_k}]$. Then,

$$\begin{aligned} f_1(b) &= \mathtt{enc}((\mathfrak{b}_1^{b_1} \dots \mathfrak{b}_k^{b_k}) \star (E_1, \iota_1)) = \mathtt{enc}(\mathfrak{b} \star (E_1, \iota_1)) \\ &= \mathtt{enc}((\mathfrak{b}\mathfrak{a}) \star (E_0, \iota_0)) = \mathtt{enc}((\mathfrak{b}_1^{a_1+b_1} \dots \mathfrak{b}_k^{a_k+b_k}) \star (E_0, \iota_0)) \\ &= f_0(a + b). \end{aligned}$$

Finally, the Hidden Shift problem is well defined as f_0 is injective because the action of $\mathrm{Cl}(\mathfrak{O})$ over $SS_{\mathfrak{O}}(p)$ is free.

□

Theorem 9 (**GRH**, Quantum $\mathfrak{O}$-Vectorisation)**.** *There is a quantum algorithm taking as input two oriented elliptic curves (E_0, ι_0) and (E_1, ι_1) in $SS_{\mathfrak{O}}(p)$ which returns an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $E_0^{\mathfrak{a}} \sim E_1$ together with the associated isogeny $\varphi_{\mathfrak{a}} : E_0 \to E_1$. This algorithm runs in expected time $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$ where l is the length of the input.*

Proof. As for the classical resolution of $\mathfrak{O}$-Vectorisation, it is sufficient to run two instances of Algorithm 4. The first one with the inputs (E_0, ι_0) and (E_1, ι_1) and the second one with the inputs $(E_0, \bar{\iota_0})$ and (E_1, ι_1). When the action has 2 orbits (see Proposition 2), one of these instances corresponds to a scenario where there is no solution to the Hidden Shift problem. It can be assumed that running Algorithm 4 on this incorrect instance will either yield an erroneous output or fail to terminate. However, since we can check the validity of a solution in polynomial time using with `Clapoti`, this does not pose any issues. Consequently, the complexity in the Theorem 9 directly follows from Lemma 8. □

This leads us to the following proof of Theorem 2.

Proof of Theorem 2. By Corollary 2 and because the factorisation of $\operatorname{disc}(\mathbb{Z}[\alpha])$ can be computed in quantum polynomial time, the α-EndRing problem reduces to $\mathfrak{O}$-EndRing in time polynomial in the length of the instance. Notice that the discriminant of the order returned by this primitivisation step can only decrease in absolute value. Then, by Proposition 3, α-EndRing reduces to $\mathfrak{O}$-Vectorisation in probabilistic time polynomial in the length of the input. Hence, by Theorem 9, α-EndRing can be solved in expected time $l^{O(1)} L_{|\operatorname{disc}\mathbb{Z}[\alpha]|}[1/2]$. □

7 Ascending the volcano

We fix K to be a quadratic number field and we consider supersingular elliptic curves over the finite field $\mathbb{F}_{p^2}$ where p is a prime which does not split in K. Let $\ell \neq p$ be a prime number.

Adding K-orientations to an ℓ-isogeny graph of supersingular elliptic curves gives a structure of volcano to each of its connected components, analogous to the structure of isogeny graphs of ordinary elliptic curves. We now introduce formally this notion before showing how results of Section 4 can be used to naviguate efficiently in this volcano. This is then used to optimise results of the previous section. Notably, we improve [Wes22a, Theorem 5] by proving that $(\mathbb{Z} + c\mathfrak{O})$-EndRing reduces to $\mathfrak{O}$-EndRing in polynomial time in the largest prime factor of c (instead of it largest *prime power* factor).

We define the K**-oriented** ℓ**-isogeny graphs** as the graph having for set of vertices the K-oriented supersingular elliptic curves up to K-isomorphism and for edges the K-oriented isogenies of degree ℓ between them.

Let $(E, \iota), (E', \iota')$ be two K-oriented supersingular elliptic curves, where ι is a primitive $\mathfrak{O}$-orientation and ι' is a primitive $\mathfrak{O}'$-orientation.

For any K-oriented isogeny $\varphi : (E, \iota) \to (E', \iota')$ of degree ℓ, we say that φ is

$$\nearrow \textbf{ ascending} \text{ if } \mathfrak{O} \subsetneq \mathfrak{O}',$$

$$\to \textbf{ horizontal} \text{ if } \mathfrak{O} = \mathfrak{O}',$$

$$\searrow \textbf{ descending} \text{ if } \mathfrak{O} \supsetneq \mathfrak{O}'.$$

We denote by $\left(\frac{\mathrm{disc}(\mathfrak{O})}{\ell}\right)$ the Legendre symbol. From [CK20], the oriented elliptic curve (E, ι) has $\ell - \left(\frac{\mathrm{disc}(\mathfrak{O})}{\ell}\right)$ descending isogenies from it. Moreover, there are in addition

- $\left(\frac{\mathrm{disc}(\mathfrak{O})}{\ell}\right) + 1$ horizontal isogenies, if $\mathfrak{O}$ is maximal at ℓ,
- one ascending isogeny, otherwise.

Moreover, an isogeny between K-oriented elliptic curves of non-prime degree is said to be ascending, horizontal or descending if its factorisation into prime-degree isogenies is only composed of ascending, horizontal or descending isogenies.

Then, we say that each component of the K-oriented ℓ-isogeny graph has a volcano structure as its shape recalls one. Indeed, it has a finite cycle of horizontal isogenies, called the **crater**, the surface or the rim, such that from each vertex starts an infinite tree of vertical isogenies. In particular, an oriented elliptic curve $(E, \iota) \in SS_{\mathfrak{O}}(p)$ is at the crater of the K-oriented ℓ-isogeny graph if and only if $\mathfrak{O}$ is maximal at ℓ. Otherwise, we say that (E, ι) is at **depth** m if the valuation at ℓ of $[O_K : \mathfrak{O}]$ is m, where O_K is the maximal order of K. This means that one can walk from (E, ι) to the crater of the K-oriented ℓ-isogeny graph by taking m ascending steps.

We provide an algorithm to walk to the crater of the volcano as an example of efficient navigation.

Lemma 9 (Walking to the crater)**.** *Let $(E, \iota) \in SS_{\mathfrak{O}}(p)$ be a $\mathfrak{O}$-oriented elliptic curve and $\ell \neq p$ a prime number. If (E, ι) is at depth at least m in the K-oriented ℓ-isogeny volcano, then one can compute the unique ascending isogeny $\varphi : (E, \iota) \to (E', \iota')$ of degree ℓ^m in time polynomial in $\ell, m, \log p$ and in the length of the representation of ι.*

In particular, one can give the representation of φ as m kernels of successive isogenies all defined over an extension of degree $O(\ell^2)$.

Proof. Let $(E, \iota) \in SS_{\mathfrak{O}}(p)$ be an $\mathfrak{O}$-oriented elliptic curve at depth m in the K-oriented ℓ-isogeny volcano and $\varphi : (E, \iota) \to (E', \iota')$ be the unique ascending K-isogeny of degree ℓ^m. We compute the isogeny φ by composing the m successive ascending isogenies of degree ℓ from (E, ι).

Let $\varphi_1 : (E, \iota) \to (E_1, \iota_1)$ be the unique ascending isogeny of degree ℓ from (E, ι). We denote by $\mathfrak{O}_1$ the order such that (E_1, ι_1) is $\mathfrak{O}_1$-primitively oriented and $\mathfrak{O}$ is a suborder of $\mathfrak{O}_1$. Let ω_1 be a generator of $\mathfrak{O}_1$. We assume, without loss of generality, that $\mathfrak{O}$ is given by a generator ω of the form $\omega = \ell\omega_1$. Then as shown in [Wes22a, Lemma 11], $\ker\varphi = \ker(\iota(\omega)) \cap E[\ell]$. As $\iota(\omega)$ is efficiently represented, $\ker\varphi$ can be computed in time polynomial in $\ell, \log p$ and l_0, where l_0 is the length of the representation of ι. One just has to compute a basis of $E[\ell]$ and to take a generator of the cyclic subgroup of $E[\ell]$ that vanishes under $\iota(\omega)$. It provides a representation of φ given by its kernel generated by a point living in an extension of degree $O(\ell^2)$. Thus, it is possible to compute the elliptic

curve $E_1 = E/\ker\varphi$ and its orientation ι_1 induced by φ_1 in time polynomial in $\ell, \log p$ and in l_0.

On the one hand, to recover E_1, we use Vélu's formula [Vé71]. On the other hand, for the computation of the induced orientation, we have

$$\iota_1(\omega_1) = \varphi_{1*}(\iota(\omega_1)) = \frac{\varphi \circ \iota(\omega_1) \circ \hat{\varphi}}{\ell} = \frac{\varphi \circ \iota(\ell\omega_1) \circ \hat{\varphi}}{\ell^2} = \frac{\varphi \circ \iota(\omega) \circ \hat{\varphi}}{\ell^2}.$$

Thus, from the known representations of φ and $\iota(\omega)$, we get an efficient representation of $\varphi \circ \iota(\omega) \circ \hat{\varphi}$ and we just need to divide it by ℓ^2 using Algorithm 1. By Theorem 3, this computation is polynomial in $l, \log p$ and in l_0 and returns a representation of ι_1 of size $O(\log(p)\log^3(\ell^2 l_0))$ such that one can evaluate it on a point in $\tilde{O}(\log^{11}(\ell^2 l_0))$ operations over its field of definition.

We do the same computation to get a representation of the unique ascending isogeny $\varphi_2 : (E_1, \iota_1) \to (E_2, \iota_2)$ of degree ℓ. First, we compute the kernel $\ker\varphi_2 = \ker(\iota_1(\omega_1)) \cap E_1[\ell]$ and deduce the curve $E_2 = E_1/\ker\varphi_2$ together with a representation of φ_2 in time polynomial in $\ell, \log p$ and in l_0. Then we recover in time polynomial in $\ell, \log p$ and l_0 a representation of the induced orientation ι_2, with the same properties as the one of ι_1.

After such m steps, one can provide efficient representations for the totality of the φ_i, for $i \in [\![1, m]\!]$, in polynomial time in $\ell, \log p, l_0$ and m. The representation $\varphi : (E, \iota) \to (E', \iota')$ is then given by the composition of the representations of φ_i, for $i \in [\![1, m]\!]$. Hence, this representation is provided by the kernels of the m successive isogenies, namely by m points living in extension of degree $O(\ell^2)$. □

Theorem 10 (GRH). *Let c be a positive integer and $\mathfrak{O}$ a quadratic order. Then $(\mathbb{Z}+c\mathfrak{O})$-*EndRing *reduces to $\mathfrak{O}$-*EndRing *in probabilistic polynomial time in the length of the input and in the largest prime factor of c.*

Proof. Let $(E, \iota) \in SS_{\mathbb{Z}+c\mathfrak{O}}(p)$ be an instance of $(\mathbb{Z}+c\mathfrak{O})$-EndRing. Let us solve it using an $\mathfrak{O}$-EndRing oracle.

Here, the main objective is to compute a representation of the unique isogeny $\varphi : E \to E'$ of degree c such that $\varphi_*(\iota)$ is an $\mathfrak{O}$-orientation. Indeed, using the $\mathfrak{O}$-EndRing oracle on the instance $(E', \varphi_*(\iota))$ gives an ε-basis of $\mathrm{End}(E')$. Then, from the ε-basis of $\mathrm{End}(E')$ and $\hat{\varphi}$, an ε-basis of $\mathrm{End}(E)$ can be computed, under GRH, in probabilistic polynomial time in the length of the input, [Wes22a, Lemma 12]. Notice that to use directly [Wes22a, Lemma 12], the isogeny $\hat{\varphi}$ needs to be represented by its kernel. It is not an issue for this proof.

First, we compute the prime factorisation of c and denote it $\prod_{i=1}^{r} \ell_i^{e_i}$. This factorisation can be done in polynomial time in $P^+(c)$. Using Lemma 9, we can successively take e_i steps to the crater of the oriented ℓ_i-isogeny volcanoes, for $i \in [\![1, r]\!]$, to reach $(E', \varphi_*(\iota))$ in polynomial time in the length of the input and in $P^+(c)$. Let us denote by (E_i, ι_i) the oriented elliptic curve obtained by walking e_1 steps from $(E_0, \iota_0) := (E, \iota)$ to the crater of the oriented ℓ_1-isogeny volcano then e_2 steps to the crater of the oriented ℓ_2-isogeny volcano and so on until walking e_i steps to the crater of the oriented ℓ_i-isogeny volcano. We denote by φ_i the isogeny of degree $\ell_i^{e_i}$ that maps (E_{i-1}, ι_{i-1}) to (E_i, ι_i). By Lemma 9, every φ_i is given by $\log(c)$ successive kernels of ℓ_i-isogenies living in extension of degree $O(P^+(c)^2)$. We then denote this decomposition of φ_i into ℓ_i isogenies by $\varphi_i = \phi_{i,m_i} \circ \cdots \circ \phi_{i,1}$. Finally, using the decomposition of every φ_i into isogenies of prime degree, we have the following decomposition of $\hat{\varphi} : (E', \iota') \to (E, \iota)$

$$\hat{\varphi} = \hat{\phi}_{1,1} \circ \cdots \circ \hat{\phi}_{1,m_1} \circ \hat{\phi}_{2,m_2} \circ \ldots \hat{\phi}_{2,1} \circ \cdots \circ \hat{\phi}_{r,1} \circ \cdots \circ \hat{\phi}_{r,m_r},$$

where all the kernels of the $\hat{\phi}_{i,j}$ are recoverable in time polynomial in $P^+(c)$ and in $\log p$.

Finally, $\mathrm{End}(E)$ is computable in probabilistic polynomial time in the length of the input and in $P^+(c)$ by propagating the knowledge of the endomorphism ring from (E', ι') to (E, ι) using the $O(\log c)$ dual isogenies of prime degree between them, thanks to [Wes22a, Lemma 12]. □

Corollary 5 (GRH). *Let c be a positive integer and $\mathfrak{O}$ a quadratic order. Then $(\mathbb{Z}+c\mathfrak{O})$-*EndRing *can be solved in probabilistic polynomial time in $(l \cdot P^+(c))^{O(1)} |\operatorname{disc}(\mathfrak{O})|^{1/4}$ where l is the length of the input and $P^+(c)$ is the largest prime factor of c.*

Proof. This is a direct consequence of the reduction of $(\mathbb{Z}+c\mathfrak{O})$-EndRing to $\mathfrak{O}$-EndRing given by Theorem 10 together with the complexity result on $\mathfrak{O}$-EndRing given by Theorem 1. □

References

[ACL+23] Sarah Arpin, Mingjie Chen, Kristin E Lauter, Renate Scheidler, Katherine E Stange, and Ha TN Tran. Orienteering with one endomorphism. *La Matematica*, 2(3):523–582, 2023. Publisher: Springer.

[ACL+24] Sarah Arpin, Mingjie Chen, Kristin E Lauter, Renate Scheidler, Katherine E Stange, and Ha TN Tran. Orientations and cycles in supersingular isogeny graphs. In *Research Directions in Number Theory: Women in Numbers V*, pages 25–86. Springer, 2024.

[BJS14] Jean-François Biasse, David Jao, and Anirudh Sankar. A quantum algorithm for computing isogenies between supersingular elliptic curves. In *International Conference on Cryptology in India*, pages 428–442. Springer, 2014.

[BKV19] Ward Beullens, Thorsten Kleinjung, and Frederik Vercauteren. CSI-FiSh: efficient isogeny based signatures through class group computations. In *International conference on the theory and application of cryptology and information security*, pages 227–247. Springer, 2019.

[BS12] Gaetan Bisson and Andrew V Sutherland. A low-memory algorithm for finding short product representations in finite groups. *Designs, Codes and Cryptography*, 63(1):1–13, 2012. Publisher: Springer.

[BS16] Jean-François Biasse and Fang Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 893–902. SIAM, 2016.

[CD20] Wouter Castryck and Thomas Decru. CSIDH on the surface. In *International Conference on Post-Quantum Cryptography*, pages 111–129. Springer, 2020.

[CD23] Wouter Castryck and Thomas Decru. An efficient key recovery attack on SIDH. In *Advances in cryptology – EUROCRYPT 2023. Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 423–447. Springer, 2023. URL: https://doi.org/10.1007/978-3-031-30589-4_15, doi:10.1007/978-3-031-30589-4_15.

[CJS14] Andrew Childs, David Jao, and Vladimir Soukharev. Constructing elliptic curve isogenies in quantum subexponential time. *Journal of Mathematical Cryptology*, 8(1):1–29, 2014.

[CK20] Leonardo Colo and David Kohel. Orienting supersingular isogeny graphs. *Journal of Mathematical Cryptology*, 14(1):414–437, 2020. Publisher: De Gruyter.

[CLG09] Denis X Charles, Kristin E Lauter, and Eyal Z Goren. Cryptographic hash functions from expander graphs. *Journal of CRYPTOLOGY*, 22(1):93–113, 2009. Publisher: Springer.

[CLP24] Mingjie Chen, Antonin Leroux, and Lorenz Panny. SCALLOP-HD: group action from 2-dimensional isogenies. In *IACR International Conference on Public-Key Cryptography*, pages 190–216. Springer, 2024.

[Cou06] Jean-Marc Couveignes. Hard homogeneous spaces. *Cryptology ePrint Archive*, page 291, 2006. URL: https://eprint.iacr.org/2006/291.

[CPV20] Wouter Castryck, Lorenz Panny, and Frederik Vercauteren. Rational isogenies from irrational endomorphisms. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, volume 12106 of *Lecture Notes in Computer Science*, pages 523–548. Springer, 2020. doi:10.1007/978-3-030-45724-2_18.

[CS22] Mathilde Chenu and Benjamin Smith. Higher-degree supersingular group actions. *Mathematical Cryptology*, 1(2):85–101, March 2022. Publisher: Florida Online Journals.

[DFDdSGF+21] Luca De Feo, Cyprien Delpech de Saint Guilhem, Tako Boris Fouotsa, Péter Kutas, Antonin Leroux, Christophe Petit, Javier Silva, and Benjamin Wesolowski. Séta: supersingular encryption from torsion attacks. In *Advances in Cryptology–ASIACRYPT 2021: 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6–10, 2021, Proceedings, Part IV 27*, pages 249–278. Springer, 2021.

[DFKL+20] Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: compact post-quantum signatures from quaternions and isogenies. In *Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part I 26*, pages 64–93. Springer, 2020.

[DG16] Christina Delfs and Steven D Galbraith. Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. *Designs, Codes and Cryptography*, 78:425–440, 2016. Publisher: Springer.

[DLRW24] Pierrick Dartois, Antonin Leroux, Damien Robert, and Benjamin Wesolowski. Sqisignhd: New dimensions in cryptography. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology – EUROCRYPT 2024*, volume 14651 of *Lecture Notes in Computer Science*, pages 3–32. Springer, 2024. doi:10.1007/978-3-031-58716-0_1.

[EHL+18] Kirsten Eisenträger, Sean Hallgren, Kristin Lauter, Travis Morrison, and Christophe Petit. Supersingular isogeny graphs and endomorphism rings: reductions and solutions. In *Advances in Cryptology–EUROCRYPT 2018: 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29-May 3, 2018 Proceedings, Part III 37*, pages 329–368. Springer, 2018.

[EHL+20] Kirsten Eisenträger, Sean Hallgren, Chris Leonardi, Travis Morrison, and Jennifer Park. Computing endomorphism rings of supersingular elliptic curves and connections to path-finding in isogeny graphs. *Open Book Series*, 4(1):215–232, 2020. Publisher: Mathematical Sciences Publishers.

[FFK+23] Luca De Feo, Tako Boris Fouotsa, Péter Kutas, Antonin Leroux, Simon-Philipp Merz, Lorenz Panny, and Benjamin Wesolowski. SCALLOP: scaling the CSI-FiSh. In *IACR international conference on public-key cryptography*, pages 345–375. Springer, 2023.

[Kan97] Ernst Kani. The number of curves of genus two with elliptic differentials. *Journal für die reine und angewandte Mathematik*, 485:93–122, 1997.

[Koh96] David Kohel. *Endomorphism rings of elliptic curves over finite fields*. PhD thesis, University of California, Berkeley, 1996.

[Kup05] Greg Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM Journal on Computing*, 35(1):170–188, 2005. Publisher: SIAM.

[LR12] David Lubicz and Damien Robert. Computing isogenies between abelian varieties. *Compositio Mathematica*, 148(5):1483–1515, 2012. Publisher: London Mathematical Society.

[LR23] David Lubicz and Damien Robert. Fast change of level and applications to isogenies. *Research in Number Theory*, 9(1):7, 2023. Publisher: Springer.

[Mil86] James S Milne. Abelian varieties. *Arithmetic geometry*, pages 103–150, 1986. Publisher: Springer.

[MMP+23] Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, and Benjamin Wesolowski. A direct key recovery attack on SIDH. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023*, volume 14008 of *Lecture Notes in Computer Science*, pages 448–471. Springer, 2023. doi:10.1007/978-3-031-30589-4_16.

[Mum70] David Mumford. *Abelian Varieties*. Oxford University Press, London, 1970.

[Onu21] Hiroshi Onuki. On oriented supersingular elliptic curves. *Finite Fields and Their Applications*, 69:101777, 2021. Publisher: Elsevier.

[PR23] Aurel Page and Damien Robert. Introducing Clapoti (s): Evaluating the isogeny class group action in polynomial time. *Cryptology ePrint Archive*, page 1766, 2023. URL: https://eprint.iacr.org/2023/1766.

[PT18] Paul Pollack and Enrique Treviño. Finding the Four Squares in Lagrange's Theorem. *Integers*, 18(A15):7–17, 2018.

[PW24] Aurel Page and Benjamin Wesolowski. The supersingular endomorphism ring and one endomorphism problems are equivalent. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology – EUROCRYPT 2024*, volume 14656 of *Lecture Notes in Computer Science*, pages 388–417. Springer, 2024. doi:10.1007/978-3-031-58751-1_14.

[Rob21] Damien Robert. *Efficient algorithms for abelian varieties and their moduli spaces*. HDR Thesis, Université de Bordeaux (UB), 2021.

[Rob22a] Damien Robert. Evaluating isogenies in polylogarithmic time. *Cryptology ePrint Archive*, page 1068, 2022. URL: https://eprint.iacr.org/2022/1068.

[Rob22b] Damien Robert. Some applications of higher dimensional isogenies to elliptic curves (preliminary version). *Cryptology ePrint Archive*, page 1704, 2022. URL: https://eprint.iacr.org/2022/1704.

[Rob23] Damien Robert. Breaking SIDH in polynomial time. In *Advances in cryptology – EUROCRYPT 2023. Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 472–503. Springer, 2023. URL: https://doi.org/10.1007/978-3-031-30589-4_17, doi:10.1007/978-3-031-30589-4_17.

[Rob24] Damien Robert. On the efficient representation of isogenies (a survey). *Cryptology ePrint Archive*, page 1071, 2024. URL: https://eprint.iacr.org/2024/1071.

[RS06] Alexander Rostovtsev and Anton Stolbunov. Public-key cryptosystem based on isogenies. *Cryptology ePrint Archive*, page 145, 2006. URL: https://eprint.iacr.org/2006/145.

[Sil86] Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate texts in mathematics*. Springer, 1986.

[Voi21] John Voight. *Quaternion algebras*. Springer Nature, 2021.

[Vé71] Jacques Vélu. Isogénies entre courbes elliptiques. *C. R. Acad. Sc. Paris, Série A*, t. 273:238–241, 1971.

[Wes22a] Benjamin Wesolowski. Orientations and the supersingular endomorphism ring problem. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology – EUROCRYPT 2022*, volume 13277 of *Lecture Notes in Computer Science*, pages 345–371. Springer, 2022. doi:10.1007/978-3-031-07082-2_13.

[Wes22b] Benjamin Wesolowski. The supersingular isogeny path and endomorphism ring problems are equivalent. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1100–1111. IEEE, 2022.

[Wes24] Benjamin Wesolowski. *Random Walks in Number-theoretic Cryptology*. HDR Thesis, ENS Lyon, 2024. URL: https://bweso.com/hdr.pdf.

[Zar74] Ju G Zarhin. A remark on endomorphisms of abelian varieties over function fields of finite characteristic. *Mathematics of the USSR-Izvestiya*, 8(3):477, 1974. Publisher: IOP Publishing.