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 AND BENJAMIN WESOLOWSKI

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 $\mathrm{disc}(\mathbb{Z}[\alpha])^{1/4}$, 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 prove that the Primitivisation problem can be solved in polynomial time (a problem previously believed to be hard), and we prove that the action of smooth ideals on oriented elliptic curves can be computed in polynomial time (previous results of this form required the ideal to be powersmooth, i.e., not divisible by any large prime power). Following the attacks on SIDH, isogenies in high dimension are a central ingredient of our results.

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, Wes21]. 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 [CGL06, EHL+18], the `CSIDH` key exchange protocol [CD20, CPV20, Wes22] or 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 anyone knows the Frobenius endomorphism. The situation is similar in [CS21, FFK+23]. The endomorphism ring problem then consists in finding 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+22b]: 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 to the above question. Yet that answer has limitations: first, as stated, it is only heuristic. Second, the output of the algorithm of [ACL+22b] may have exponential size, which could considerably increase the cost of applying the equivalence.

The schemes of [CD20, CS21, 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 [CS21, 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 [Wes22]. 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+22b] 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 insight into 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 [Sil13] 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 and 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+22b] 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 and a 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, an idea is to extend the given orientation to larger quadratic orders until it is no longer possible. The bottleneck of this method is that it requires performing divisions of endomorphisms by integers to check if the orientation can be extended to superorders. Namely, if this division can be done efficiently as many times

as needed, one will get a classical subexponential algorithm and a quantum polynomial algorithm solving PRIMITIVISATION as the only remaining costly step is to factor the discriminant of the order.

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 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 [Wes22]. Therefore, the security of many protocols, such as CSIDH [CD20], CSI-FiSh [BKV19] and CSURF [CD20], reduces to $\mathfrak{O}$-ENDRING see [Wes22].

In the current state of the art, using l to denote the length of the input, the $\mathfrak{O}$-VECTORISATION problem 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 [Wes22, Proposition 4].

1.2. **Contributions.** We prove rigorously, removing all the previous heuristics, the following list of algorithms which are at least as good, regarding their complexity, as previous results in the literature and better for some them. 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 representation and encoding.

- In Section 4, we develop the first ingredient for the rest of the paper: an algorithm to divide endomorphisms by integers in polynomial time such that the output is efficiently represented. It is a straightforward generalisation of the division of Frobenius endomorphisms by integers used in the algorithm of Robert to compute the endomorphism ring of an ordinary elliptic curve, [Rob22b, Theorem 4.2]. Before the attacks of SIDH, it was only possible to divide endomorphisms by integers either in superpolynomial time, or by degrading the quality of the representation (getting exponentially worse with each application), it is not the case here.
- In Section 5 and 6, we give three applications of this division algorithm:
  - We adapt Robert's algorithm for computing in polynomial time the endomorphism ring of ordinary elliptic curves [Rob22b] to one of its supersingular counterparts: the resolution of the PRIMITIVISATION problem. We prove that, when the factorisation of $\mathrm{disc}(\mathfrak{O})$ is known, there is a classical polynomial algorithm solving PRIMITIVISATION.
  - We provide a polynomial algorithm computing the action of smooth ideals. Previous polynomial-time algorithms for this task required the norm of the input ideal to be powersmooth. We expect these techniques to improve the practicality of cryptosystems exploiting this group action, such as [FFK+23].
  - We give an algorithm, with an rigorous analysis under the generalised Riemann hypothesis, to compute the action of any ideal in supexponential time. The previous subexponential algorithms were based on heuristic assumptions.

We now use l to denote the length of the input, and use the standard L-notation (Definition 2.3).

- In Section 7, under the generalised Riemann hypothesis, we prove

– a classical algorithm solving $\mathfrak{O}$-VECTORISATION in expected time $l^{O(1)}|\operatorname{disc}(\mathfrak{O})|^{1/4}$.
– a quantum algorithm solving $\mathfrak{O}$-VECTORISATION in expected time $l^{O(1)}L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$.

This directly leads to

– a classical algorithm solving $\mathfrak{O}$-ENDRING in expected time $l^{O(1)}|\operatorname{disc}(\mathfrak{O})|^{1/4}$.
– a quantum algorithm solving $\mathfrak{O}$-ENDRING in expected time $l^{O(1)}L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$.

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

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

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

In addition, we detail how the algorithmic improvements of Section 5 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 [Wes22, 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 $\operatorname{disc}(\mathfrak{O})$ and in the largest prime factor of c. This mainly removes the heuristics of [Wes22, 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. The authors were supported by the CHARM ANR-NSF grant (ANR-21-CE94-0003) 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 I and II.

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 $\mathcal{O}$, $\operatorname{disc}(\mathcal{O})$ 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.

Definition 2.1 ((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 2.2 (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 2.3 (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 2.4 (Cayley graph)**.** *Let G be a finite group and let $S \subseteq G$ be a generating subset of G. The **Cayley graph** $\mathbf{Cay}(G, S)$ 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 2.5 (**(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 S_x be the set*

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

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 to introduce the recent notions of orientations [CK20]. For more details about elliptic curves theory, we refer the reader to Silverman's book [Sil13].

An **elliptic curve** is an abelian variety of dimension 1. **Isogenies** of elliptic curves are non-trivial homomorphisms between them. Isogenies 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 over 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 $E[p] \simeq \{0\}$. The **degree** of an isogeny φ from an elliptic curve E to an elliptic curve E' is the smallest integer n such that there exists an isogeny ψ from E' to E verifying $\varphi \circ \psi = [n]$. We then call such isogeny ψ the **dual isogeny** of φ and denote it $\hat{\varphi}$. An isogeny is said to be **separable** when its degree is equal to the cardinality of its kernel. In this paper, we only work with supersingular elliptic curves defined over a field of characteristic p, where p is a fixed prime number. For any prime $\ell \neq p$, we call **ℓ-isogenies** the separable isogenies 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 in 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 recent notion of orientations [CK20] describes such embeddings.

Let K be a quadratic number field.

Definition 2.6 (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 **$\mathcal{O}$-orientation** and (E, ι) is an **$\mathcal{O}$-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 $\iota(\mathcal{O}) = \mathrm{End}(E) \cap \iota(K)$, then (E, ι) is said to be **primitively $\mathcal{O}$-oriented**.*

Definition 2.7 (Oriented isogeny)**.** *Let (E, ι) and (E', ι') be two K-oriented elliptic curves, and let $\varphi : E \to E'$ be an isogeny. We say that φ is **K-oriented** if the K-orientation on E' induced by φ, denoted by $\varphi_*(\iota)$, is equal to ι'. Explicitly, this orientation is given by*

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

*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. 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 with their endomorphism ring 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 an 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 $\mathfrak{O}$-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 2.8 ([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 $\mathfrak{O}$-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. **Encoding and representation of isogenies.**

Definition 2.9 (Algorithm of evaluation of isogenies)**.** *An **algorithm of evaluation of isogenies** is an algorithm which takes as input an encoding of an isogeny and an encoding of a point and returns the evaluation of the isogeny at this point. We shall refer to those algorithms simply by **algorithm of evaluation**.*

Definition 2.10 (Representation of an isogeny)**.** *A **representation of an isogeny** φ is a specific encoding of φ associated to an algorithm of evaluation.*

Definition 2.11 (Efficient representation of an isogeny)**.** *Let E be an elliptic curve over a finite field of characteristic p. A representation of an isogeny φ of E is **efficient** if its associated algorithm of evaluation returns $\varphi(P)$ in time polynomial in $k \log p$ and in the length of the encoding of φ for any $P \in E(\mathbb{F}_{p^k})$. Moreover, it is assumed that an efficient representation of an isogeny φ has length $\Omega(\log(\deg(\varphi)))$.*

Definition 2.12 (Efficient representation of orientation)**.** *Let E be an elliptic curve. A representation of an orientation $\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$ is given by a generator ω of the order $\mathfrak{O}$ together with a representation of the isogeny $\iota(\omega)$. It is said to be an **efficient representation** if the representation of $\iota(\omega)$ is efficient. By extension, the oriented elliptic curve (E, ι) is said to be **efficiently represented** if ι is.*

We now define a function `enc`, introduced in [Wes22], 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 form for elliptic curves given by their j-invariant,
  For instance, $E : y^2+xy = x^3-(36x+1)/(j(E)-1728)$ if $j(E) \notin \{0, 1728\}$, see [Sil13, page 47],
- A generator ω of $\mathfrak{O}$, typically one with the smallest possible norm,
- A deterministic procedure that takes as input an elliptic curve $E/\mathbb{F}(q)$ in canonical form and returns a point $P \in E(\mathbb{F}_q)$ of order greater than $4N(\omega)$.

Then the map `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 [Wes22], this encoding is an unique encoding of the K-isomorphism class of (E, ι). Moreover, when ι is efficiently represented, the encoding `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 [Sil13, page 47], for instance

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

and

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

In this 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 `enc`(S).

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 isogeny-based cryptography using supersingular elliptic curves is the following ℓ-IsogenyPath problem, where ℓ is a prime number. Notice that in this paper,

we shall not talk about ordinary elliptic curves since, nowadays, almost all isogeny-based schemes use supersingular elliptic curves.

Problem 3.1 (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 the isogeny-based cryptography. This problem has been shown to be equivalent, under GRH, to the problem of finding the structure of the endomorphism ring of a supersingular elliptic curve [Wes21]. This second problem is called the endomorphism ring problem or 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 [Wes21].

Definition 3.2 (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 3.3 (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], and the best quantum algorithms have complexity in $\tilde{O}(p^{1/4})$, see for example [BJS14].

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

Problem 3.4 ($\mathfrak{O}$-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 [Wes22].

Problem 3.5 ($\mathfrak{O}$-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, CS21, FFK+23], reduces to $\mathfrak{O}$-Vectorisation.

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

Proposition 3.6 (**GRH**, Proposition 7 in [Wes22])**.** *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 [Wes22, 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 [Wes22, Lemma 1] and deduce a quadratic number α such that $\mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$ is an orientation.

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

Problem 3.7 (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+22b] as a hard problem. It reduces the α-EndRing problem to the $\mathfrak{O}$-EndRing problem.

Problem 3.8 (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+22b] 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. We prove in Section 5 that, using tools involving higher dimensional isogenies, see Section 4, 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 3.6, reductions of α-EndRing to $\mathfrak{O}$-Vectorisation. This reasoning is formalized in Section 7 together with a rigorous analysis of the complexity of $\mathfrak{O}$-Vectorisation.

4. Efficient division of endomorphisms

In this section, we discuss higher dimensional isogenies and how they can be used to efficiently divide endomorphisms by integers. The goal of this section is to prove Theorem 4.1 below (and its more precise formulation Theorem 4.16).

Theorem 4.1. *Algorithm 1 takes as input*

- *A supersingular elliptic curve $E/\mathbb{F}_{p^2}$ such that $p > 3$,*
- *An endomorphism φ of E,*
- *An integer $n < \deg \varphi$,*

and returns in time polynomial in $\log p$ and $\log \deg(\varphi)$ a representation of φ/n if it is an endomorphism; otherwise it returns `False`*.*

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

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 4.1 and its proof originate from [Rob22b]. Theorem 4.1 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 Castrick 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, see e.g. [Mil86], the suitable generalisation of elliptic curves to higher dimensions.

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 4.2 ((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 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 4.3. 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 cuvres 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

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

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

Given a unique isogeny $\varphi : E \to E'$, one can construct an isogeny $\boldsymbol{\varphi} : E^n \to E^n$ by setting $\boldsymbol{\varphi}(P) = (\varphi(P_1), \ldots, \varphi(P_n)), \forall P = (P_1, \ldots, P_n) \in E^n$. The matrix form of $\boldsymbol{\varphi}$ is then the identity matrix of dimension n multiplied by φ.

Any isogeny $F : E_1 \times \cdots \times E_n \to E'_1 \cdots \times \ldots E'_n$ between two products of elliptic curves can be written using a matrix form with the following injection map

$$\begin{array}{rccc} \tau_j : & E_j & \to & E_1 \times \cdots \times E_n \\ & P & \mapsto & (\underbrace{0, \ldots, 0}_{i-1}, P, \underbrace{0, \ldots, 0}_{n-i}) \end{array}$$

and projection map

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

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 $M(F) = (F_{i,j})_{i,j \in [\![1,n]\!]}$.

Thanks to the previous notations, we can provide a formal definition of what we mean by embedding an isogeny in higher dimensions.

Definition 4.4 (Embedding representation)**.** *Let $\varphi : E \to E'$ be an isogeny of elliptic curves and n be an integer. An* ***embedding representation*** *of φ in dimension n is a triplet (F, i, j) associated*

to an algorithm to evaluate F, where $F : E^n \to E'^n$, $i, j \in [\![1, n]\!]$ 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 4.5 (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.

We extend the notions of **algorithms of evaluation of isogenies** and **representations of an isogeny** 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 can be identified to their kernel, up to isomorphisms, and then be handled 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 can be identified to 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 4.6 (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 4.7 (Maximal isotropic subgroup)**.** *With the same notations as in Definition 4.6. 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 4.8 (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 from its kernel. Thus, it is important to have a description of the group of automorphisms of E^n as, by Lemma 4.8, endomorphisms with the same kernel differ only by an automorphism.

Lemma 4.9. *Let E be an elliptic curve and n be an integer. Let $\mathrm{Aut}(E^n, \lambda_{E^n})$ be the group of automorphisms of the principally polarised abelian variety (E^n, λ_{E^n}). Then the elements of $\mathrm{Aut}(E^n, \lambda_{E^n})$ in matrix form are the matrices of dimension n with entries in $\mathrm{Aut}(E) \cup \{0\}$ containing only one non-zero entry per column and per row.*

Proof. Let $\psi \in \mathrm{Aut}(E^n, \lambda_{E^n})$. 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 $M(\psi)\tilde{M}(\psi) = \mathrm{I}_n$. Then for any

$i \in [\![1, n]\!]$,

$$\sum_{j=1}^{n} \psi_{i,j} \circ \hat{\psi}_{i,j} = [1].$$

Thus

$$\sum_{j=1}^{n} [\text{degree}(\psi_{i,j})] = [1],$$

which implies that exactly one $\psi_{i,j}$ is non-zero and has degree one, hence it is an automorphism. The identity $\tilde{\psi}\psi = [1]$ yields the same results for columns. □

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 isomorphisms. This result is a generalisation of a construction in dimension 1 given by Kani in [Kan97].

Lemma 4.10 (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 $d_1 d_2$-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 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 $2g$.

We state Lemma 4.11 which describes how an endomorphism of degree N between elliptic curves can be embedded into an N'-endomorphism in dimension 8 knowing its image over the N'-torsion group when $N' > N$. In particular, it allows to get algorithms to efficiently divide endomorphisms by integers as their image over torsion subgroup can easily be computed.

Lemma 4.11. *Let E be an elliptic curve over a finite field $\mathbb{F}_{p^k}$ and φ be an endomorphism of degree N of E. 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 be the endomorphism over E^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 $H := \{(\tilde{\alpha}(P), \boldsymbol{\varphi}(P)) | P \in E^4[N']\}$; then there exists an N'-isogeny of E^8 of kernel H. Furthermore, the following holds for any N'-isogeny G of E^8 of kernel H.

- *The image of G is isomorphic to E^8 as principally polarised abelian varieties.*
- *For any isomorphism $\gamma : G(E^8, \lambda_{E^8}) \to (E^8, \lambda_{E^8})$, there exist an integer $i \in [\![1, 8]\!]$ and an automorphism ψ of E such that the following diagram commutes*

$$\begin{array}{ccc} E & \xrightarrow{G \circ \tau_1} & G(E^8) \\ {\scriptstyle \varphi}\downarrow & & \downarrow{\scriptstyle \pi_i \circ \gamma} \\ E & \xrightarrow{\psi} & E \end{array}$$

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

Proof. We use the same notations as above. By Lemma 4.10, there exists an N'-endomorphism F of E^8 with kernel $\{(\tilde{\alpha}(P), \boldsymbol{\varphi}(P)) | P \in E^4[N']\}$ given by the matrix

$$\begin{pmatrix} M(\alpha) & M(\tilde{\boldsymbol{\varphi}}) \\ -M(\boldsymbol{\varphi}) & M(\tilde{\alpha}) \end{pmatrix}.$$

For any $P \in E$, $F(\tau_1(P)) = (m_1P, m_2P, m_3P, m_4P, -\varphi(P), 0, 0, 0)$.

Let G be an N'-isogeny of E^8 with $\ker G = \ker F$. Then by Lemma 4.8, $G(E^8, \lambda_{E^8})$ and (E^8, λ_{E^8}) are isomorphic and for any isomorphism γ from $G(E^8, \lambda_{E^8})$ to (E^8, λ_{E^8}) there exists an automorphism ψ of E^8 such that $\gamma \circ G = \psi \circ F$. Moreover by Lemma 4.9, there exist 8 automorphisms $\psi_1, \dots \psi_8$ of E and a map σ permuting coordinates of the points of E^8 such that for any point $(P_1, \dots, P_8)$ of E^8, $\psi(P_1, \dots, P_8)$ is equal to $\sigma(\psi_1(P_1), \dots, \psi_8(P_8))$. Let i be the integer such that $\pi_i(\sigma(P_1, \dots, P_8)) = P_5$ for any $(P_1, \dots, P_8) \in E^8$. Then, for any $P \in E$,

$$\pi_i(\gamma(G(\tau_1(P)))) = \pi_i(\psi(F(\tau_1(P)))) = \pi_i(\sigma((\psi_1(m_1P), \dots, \psi_4(m_4P), \psi_5(-\varphi(P)), 0, 0, 0))) = \psi'(\varphi(P)),$$

where ψ' is the automorphism $[-1] \circ \psi_5$. □

This embedding can be evaluated efficiently using the analogue of Vélu's formulae in higher dimension introduced by Lubicz and Robert, [LR12]. It is even possible to compute embeddings of endomorphisms of degree N knowing only their image on the N'-torsion group when ${N'}^2 > N$.

Remark 4.12. A crucial ingredient used by Lubicz and Robert to get the generalisation of Vélu's formulae to principally polarised abelian varieties is the algebraic theta functions. In this paper, we do not introduce this notion. Instead, we consider the algorithm presented in [LR12] as a black-box taking as input a principally polarised abelian variety A together with one of its maximal isotropic subgroups H and returning the principally polarised quotient abelian variety A/H and a representation of the isogeny from A to A/H. Nevertheless, we emphasize that this algorithm returns in fact the isomorphism class of A/H by providing the theta null point associated to A/H. Since theta null points are coordinates on the moduli space of the principally polarised abelian varieties, they play a similar role to j-invariants for elliptic curves. Meaning that two principally polarised abelian varieties that share the same theta null point are isomorphic. For more information about theta functions, theta null points and moduli spaces, we refer the reader to [Rob21].

Lemma 4.13. *With the same notations and conditions as in Lemma 4.11, if there is an integer* N'' *such that* ${N''}^2 = N'$ *then any* N'*-endomorphism* F *of* E^8 *of kernel*

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

can be decomposed as $F = F_2 \circ F_1$ *where*

- F_1 *is an* N''*-isogeny of kernel given by the points* $(\tilde{\alpha}(P_i), \boldsymbol{\varphi}(P_i))$ *for* $i \in [\![1, 8]\!]$,
- $\widetilde{F_2}$ *is an* N''*-isogeny of kernel given by the points* $(\alpha(P_i), -\boldsymbol{\varphi}(P_i))$ *for* $i \in [\![1, 8]\!]$,

with $(P_1, \dots, P_8)$ *a basis of* $E^4[N'']$.

In addition, given bounds $B \geq P^*(N''), M \geq \Delta_E(N''), D \geq \Delta_{E,2}(N'')$ *and the prime factorisation* $\prod_{i=1} \ell_i^{e_i}$ *of* N'', *one can compute a representation of* F_1 *or* $\widetilde{F_2}$ *as a product of* $O(\log N'')$ $\ell_i^{e_i}$*-isogenies such that*

- *it takes* $O(B^8 D \log^2(N'') \log(B))$ *arithmetic operations over* $\mathbb{F}_{p^k}$ *plus* $O(\log N'')$ *evaluations of* φ *on the bases of* $E[\ell_i^{e_i}]$ *to get the representation,*
- *the representation has size* $O(kM \log(N'') \log(p))$ *bits,*
- *the representation allows to evaluate the isogeny on a point in* $O(B^8 M \log(N'') \log(B))$ *operations over its field of definition.*

Proof. The decomposition of F as two N''-isogenies F_1, F_2 and the description of the kernels come from [Rob22b, Lemma 2.4]. The proof of the different complexities is totally analogous to the algorithm described in [Rob22a, 4. The Algorithm]. □

Remark 4.14. In this section, we always assume that the isogenies are embedded into dimension 8. Lemma 4.13, 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.

Finally, Lemma 4.15 assures that one can also embed an endomorphism divided by an integer using Lemma 4.11 and 4.13.

Lemma 4.15. *Let E be an elliptic curve over a finite field $\mathbb{F}_{p^k}$ and φ be an endomorphism of E. 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 be an m-endomorphism of E^4 with $m = N' - N$.*
Then $H := \{(\tilde{\alpha}(P), \boldsymbol{s\varphi}(P)) | P \in E^4[N']\}$ is a maximal isotropic subgroup of $E^8[N']$.

Moreover, let us assume there exists an integer N'' such that $N''^2 = N'$ and denote $r = n^{-1} \mod N''$, then $H_1 := \{(\tilde{\alpha}(P), \boldsymbol{r\varphi}(P)) | P \in E^4[N'']\}$ and $H_2 := \{(\alpha(P), -\boldsymbol{r\varphi}(P)) | P \in E^4[N'']\}$ are maximal isotropic subgroups of $E^8[N'']$.

Proof. The subgroup structure of H comes immediately by construction. We claim that H is maximal isotropic. Indeed, let λ be the polarisation over E. Let $(\tilde{\alpha}(P), \boldsymbol{s\varphi}(P))$ and $(\tilde{\alpha}(Q), \boldsymbol{s\varphi}(Q))$ with $P = (P_1, P_2, P_3, P_4)$ and $Q = (Q_1, Q_2, Q_3, Q_4)$ in $E^4[N']$. Then,

$$\begin{aligned} e_{N',\lambda^8}((\tilde{\alpha}(P), \boldsymbol{s\varphi}(P)), (\tilde{\alpha}(Q), \boldsymbol{s\varphi}(Q))) &= e_{N',\lambda^4}(\tilde{\alpha}(P), \tilde{\alpha}(Q)) \cdot e_{N',\lambda^4}(\boldsymbol{s\varphi}(P), \boldsymbol{s\varphi}(Q)), \\ &= e_{N',\lambda^4}(P,Q)^m \cdot e_{N',\lambda^4}(P,Q)^{s^2 \deg(\boldsymbol{\varphi})} = e_{N',\lambda^4}(P,Q)^{(m+s^2n^2N)}, \\ &= e_{N',\lambda^4}(P,Q)^0 = 1, \text{ as } m + s^2n^2N \equiv N' \equiv 0 \mod N'. \end{aligned}$$

Thus H is isotropic with respect to the product polarisation. Finally, it is also maximal since it has order N'^8. To prove that H_1 and H_2 are maximal isotropic subgroups of $E^8[N'']$ we use similar computations and the fact that N'' divides N' thus $m + r^2n^2N \equiv m + N \equiv N' \equiv 0 \mod N''$. □

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

Theorem 4.16. *Algorithm 1 is correct and runs in*

- $O(\max(M^2, D)B^8 \log^2(N'')\log(B))$ *operations over $\mathbb{F}_{p^2}$,*
- *plus the cost of the factorisation of N'',*
- *plus the cost of the computation of the bases of $E[\ell^e]$ for each powerprime divisor ℓ^e of N'',*
- *plus the cost of $O(\log N'')$ evaluations of φ over these bases,*

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 endomorphism, the output representation of φ/n has the following properties:

- *It has size $O(M \log(N'')\log(p))$.*
- *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 4.15, $\ker F_1$ and $\ker \widehat{F_2}$ are always maximal isotropic subgroups of $E^8[N'']$ and thus the isogenies $F_1 : E^8 \to E^8/\ker F_1$ and $\widetilde{F_2} : E^8 \to E^8/\ker \widetilde{F_2}$ are well defined.

Algorithm 1 `Endomorphism division`

Input : $E/\mathbb{F}_{p^2}$ a supersingular elliptic curve such that $p > 3$, $\varphi \in \mathrm{End}(E)$, two integers n and $N''^2 > 4\deg(\varphi)$ such that $(N'', p\deg(\varphi)) = 1$.

Output : A representation of φ/n if it is an endomorphism, `False` otherwise.

1: Set $N \leftarrow \deg(\varphi)/n^2$.
2: **if** $N \notin \mathbb{N}$ **then**
3: **return** `False`
4: Set $m \leftarrow N''^2 - N$.
5: Decompose m as $m_1^2 + m_2^2 + m_3^2 + m_4^2$.
6: Set $M \leftarrow \begin{pmatrix} m_1 & -m_2 & -m_3 & -m_4 \\ m_2 & m_1 & m_4 & -m_3 \\ m_3 & -m_4 & m_1 & m_2 \\ m_4 & m_3 & -m_2 & m_1 \end{pmatrix}$.
7: Let α be the m-endomorphism over E^4 given by the matrix M.
8: Let $\tilde{\alpha}$ be the dual isogeny of α with respect to the product polarisation.
9: $s \leftarrow n^{-1} \mod N''$.
10: Compute a factorisation $\ell_1^{e_1} \dots \ell_r^{e_r}$ of N''. .
11: Compute bases $(P_{1,i}, P_{2,i})$ of $E[\ell_i^{e_i}]$ for $i \in [\![1, r]\!]$.
12: Set formally $(P_1, P_2) \leftarrow (\sum_{i=1}^r P_{1,i}, \sum_{i=1}^r P_{2,i})$ a basis of $E[N'']$.
13: Compute a representation of an N''-isogeny F_1 of E^8 of kernel

$$\ker F_1 = \{(\tilde{\alpha}(\tau_i(P_j)), \boldsymbol{s\varphi}(\tau_i(P_j)))|\forall i \in [\![1, 4]\!], \forall j \in \{1, 2\}\}.$$

14: Compute a representation of an N''-isogeny F_2 of E^8 such that

$$\ker \widetilde{F_2} = \{(\alpha(\tau_i(P_j)), -\boldsymbol{s\varphi}(\tau_i(P_j)))|\forall i \in [\![1, 4]\!], \forall j \in \{1, 2\}\}.$$

15: **if** $E^8/\ker F_1 \not\simeq E^8/\ker \widetilde{F_2}$ **then**
16: **return** `False`.
17: Compute a representation of F_2 the dual isogeny of $\widetilde{F_2}$.
18: Set the isogeny $F := F_2 \circ F_1$ represented by the composition of representations.
19: Set $\gamma : F(E^8, \lambda_{E^8}) \to (E^8, \lambda_{E^8})$ be an isomorphism of principally polarised abelian varieties.
20: Compute the group $\mathrm{Aut}(E)$.
21: **for** $t \in [\![1, 8]\!]$ **do**
22: **for** $\psi \in \mathrm{Aut}(E)$ **do**
23: **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**
24: **return** The representation of φ/n induced by $\psi^{-1} \circ \pi_t \circ \gamma \circ F \circ \tau_1$.
25: **return** `False`

When φ/n is an endomorphism of E, we have that $(\varphi/n)|_{E[N''^2]} = (s\varphi)|_{E[N''^2]}$. Hence, by Lemma 4.13 and Lemma 4.11, $F := F_2 \circ F_1$ is isomorphic to an N''^2-isogeny that embeds φ/n. More precisely, $F(E^8, \lambda_{E^8}) \simeq (E^8, \lambda_{E^8})$ as principally polarised abelian varieties and for any isomorphism $\psi : F(E^8, \lambda_{E^8}) \to (E^8, \lambda_{E^8})$, the N''^2-isogeny $\psi \circ F$ is an endomorphism of (E^8, λ_{E^8}) such that there exist an automorphism γ of E and an integer $t \in [\![1, 8]\!]$ such that

$$\pi_t(\psi(F(\tau_1(P)))) = \gamma(\varphi/n)(P), \forall P \in E, \tag{1}$$

where $\pi_t : E^8 \to E, (P_1, ..., P_8) \mapsto P_t$.

We check at line 16 if F is an endomorphism by checking if $E^8/\ker F_1 \simeq E^8/\ker \widetilde{F_2}$. If not, neither is φ/n by Lemma 4.11. Otherwise, we look for an automorphism ψ and an integer t verifying Equality (1).

In the for loop, we search for a solution (ψ, t) of Equality (1) over the bases of $E[\ell_i^{e_i}], \forall i \in [\![1, r]\!]$. It is equivalent to checking Equality (1) over $E[N'']$ as $(\ell_i, \ell_j) = 1, \forall i \neq j \in [\![1, r]\!]$. Moreover as $(N'')^2 > 4\deg\varphi$, a solution of (1) over $E[N'']$ is a solution over the entire elliptic curve E. Indeed,

as noticed in [Rob22b, Section 4.], when two endomorphisms of degree d are equal over $E[M]$ with $M^2 > 4d$, they are equal everywhere.

Since we are doing an exhaustive search at line 21, if φ/n is an endomorphism, the algorithm will find an embedding representation $(F, 1, t)$ of φ/n up to an isomorphim and an automorphism. If no such coefficient of F is found, Lemma 4.11 implies that φ/n is not an endomorphism.
The output representation of φ/n is then given by the composition of the representation of γ^{-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 $B \geq P^*(N''), M \geq \Delta_E(N''), D \geq \Delta_{E,2}(N'')$.

[1-9] The computational cost of these lines is negligible compared to the rest of the algorithm. In particular, the decomposition of m at the line 5 can be done in $O(\log^2 N'')$, see [PT18].

[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 $E[N'']$ by (P_1, P_2) only formally to get simple notations. The computation are always done with the $(P_{1,i}, P_{2,i})$, with $i \in [\![1, r]\!]$.

By Lemma 4.13, getting a representation of the N''-isogenies F_1 and $\widetilde{F_2}$ and checking if $E^8/\ker F_1 \simeq E^8/\ker \widetilde{F_2}$, see Remark 4.12 for more precision about this verification, takes :

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

[17-18] The representation of F_2 is computed from the dual isogenies of every isogeny composing the representation of $\widetilde{F_2}$. Thanks to Lemma 4.13 used to get the representation of $\widetilde{F_2}$, we know that $\widetilde{F_2} : E^8 \to E^8/\ker \widetilde{F_2}$ is given as

$$\widetilde{F_2} = F_{2,r} \circ \cdots \circ F_{2,1}$$

where $F_{2,i}$ is a isogeny of degree $\ell_i^{e_i}$, for $i \in [\![1, r]\!]$. We have

$$F_2 = \widetilde{F_{2,1}} \circ \cdots \circ \widetilde{F_{2,r}}.$$

The kernels of each dual isogeny are computable, since we have

$$\begin{aligned}
\ker \widetilde{F_{2,1}} &= F_{2,1}(E^8[\ell_1^{e_1}]),\\
\ker \widetilde{F_{2,2}} &= F_{2,2}(F_{2,1}(E^8[\ell_2^{e_2}])),\\
&\cdots\\
\ker \widetilde{F_{2,r}} &= F_{2,r} \circ F_{2,r-1} \circ \cdots \circ F_{2,1}(E^8[\ell_t^{e_r}]),
\end{aligned}$$

and we already know bases $E^8[\ell_i^{e_i}]$ and representations of $F_{2,i}$, for all $i \in [\![1, r]\!]$. Then computing these kernels takes $O(\log N'')$ evaluations of the $\ell_i^{e_i}$-isogenies $F_{2,i}$ over the groups $E[\ell_j^{e_j}]$, for $i, j \in [\![1, r]\!]$. Using the Lubicz-Robert algorithm generalising Vélu's formulae [LR23], each composition takes $O(B^8 \log(B))$ operations over extension fields of degree at most D. Thus it takes $O(B^8 D \log^2(N'') \log(B))$ operations over $\mathbb{F}_{p^2}$. They constitute a representation of the isogeny F_2, associated to the algorithm of evaluation given by the generalisation of Vélu's formulae in higher dimensions. Moreover, the representation of F_2 has the same properties as F_1 and $\widetilde{F_2}$, i.e. this representation of F_2 has size $O(kM \log(N'') \log(p))$ and allows to evaluate a point in $O(B^8 M \log(N'') \log(B))$ operations over its field of definition. For more information about this type of computations, the reader is referred, once again, to [Rob22a, 4. The algorithm].

[19] The isomorphism γ at line 19 is directly given by the product polarisation.

[20 - 25] The group of automorphisms of E, for $p > 3$, is easy to compute since it is generated by

$$\begin{cases} \{(x,y) \mapsto (x,-y), (x,y) \mapsto (-x,iy)\} & \text{if } j(E) = 0, \text{ with } i \text{ a primitive 2-nd root of unity in } \bar{\mathbb{F}}_p, \\ \{(x,y) \mapsto (x,-y), (x,y) \mapsto (\zeta_3 x, y)\} & \text{if } j(E) = 1728, \text{ with } \zeta_3 \text{ a primitive 3-rd root of unity in } \bar{\mathbb{F}}_p, \\ \{(x,y) \mapsto (x,-y)\} & \text{otherwise.} \end{cases}$$

The loop at line 21 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 $E[\ell_i^{e_i}]$, $\forall i \in [\![1, r]\!]$ plus $O(B^8 M \log^2(N'') \log(B))$ operations over extension of degree at most M.

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 kernels given F_1 and F_2 thus it has size $O(M \log(N'') \log p)$. 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 4.1 which concludes this section about efficient division of endomorphisms.

Proof of Theorem 4.1. To get this result, one only need 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 N'':

We compute it by multiplying successive primes, coprime to $p \deg(\varphi)$, until their product is greater than $2\sqrt{(\deg(\varphi))}$. It takes $O(\log \deg(\varphi))$ arithmetic operations and gives an integer N'' such that $N''^2 > 4 \deg(\varphi)$, coprime to $p \deg(\varphi)$, $O(\log \deg(\varphi))$-powersmooth and such that $\log N'' = O(\log \deg(\varphi))$. Then, with the same notations as in Theorem 4.16, 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'', we already know its factorisation 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 very recently in [ACL+22b] 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 quandratic 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. Before the introduction of higher dimensions for computations over elliptic curves, the most efficient approach was subexponential under GRH and mainly based on the theory of complex multiplication [Bis12]. Hence, using the theory of primitive orientations, which is very similar to complex multiplication, to solve Primitivisation in subexponential time is conceivable. The bottleneck of this approach is that we need to compute efficiently action of ideals which required, in the previous literature, considering only powersmooth ideals and compelled the presence of heuristics. We show in Section 6, thank to the algorithm dividing endomorphisms by integers presented in Section 4, that we can now compute efficiently the action of smooth ideals without heuristics. Though it gives us a possible classical subexponential algorithm to solve Primitivisation, higher dimensional isogenies actually provide a more efficient and direct approach. Indeed, in [Rob22b, Section 4], it is shown how efficient division of endomorphisms allows on to efficiently compute the endomorphism ring of ordinary elliptic curves.

In this section, we describe how Robert's method can be adapted to solve Primitivisation.

In the first place, Theorem 5.1 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 5.2 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 5.1 (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 $N''^2 > 4N$ 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 powerprime divisor ℓ^e of N'' plus the cost of the computation of $O(\log N'')$ evaluations of θ over these bases.

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 integer ring of 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 $N''^2 > 4N$ 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 $(E, \theta', \ell_i, N'')$.
    $\alpha' \leftarrow \alpha'/\ell_i$.
    $\theta' \leftarrow \theta'/\ell_i$.
if $(\theta'+1)/2 \in \mathrm{End}(E)$ **then** ▷ using Algorithm 1 with the input $(E, \theta'+1, 2, N'')$.
  $(\alpha', \theta') \leftarrow ((\alpha'+1)/2, (\theta'+1)/2)$.
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 4.16, 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 powerprime divisor ℓ^e of N'' plus the cost of the computation of $O(\log N'')$ evaluations of θ' over these bases.

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.

□

Corollary 5.2 demonstrates that Primitivisation can be solved in polynomial time when the input is efficiently represented by applying Theorem 5.1.

Corollary 5.2. *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 a 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,*
- *The endomorphism $\iota(\omega)$ can be evaluated at a point in $\tilde{O}(\log^{11}(N))$ operations over its field of definition.*

This algorithm runs in time polynomial in $\log N$ *and* $\log p$.

Proof. As for Theorem 4.1, 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. □

6. Computing the actions of ideals

The class group action over $SS_{\mathfrak{O}}(p)$ is a key notion of the orientation's theory and is a crucial ingredient of the algorithms presented in Section 7 to solve the $\mathfrak{O}$-Vectorisation problem.

In this section, we describe how the higher dimensional tools, introduced in Section 4, bring significant improvements in the computation of theses actions. Indeed, using Algorithm 1, it is possible to relax most of the contraints of the previous literature's algorithms computing these actions.

To begin with, we show how we can now compute actions of smooth ideals efficiently together with the isogenies they induce. This is done in polynomial time and can be applied as many times as wished without impairing the quality of the representation. Before, such algorithms were restricted to powersmooth ideals, and they degraded the quality of the representation of the

orientation (so applying the action of several powersmooth ideal iteratively would actually take an exponential time).

To get an algorithm which efficiently compute actions of smooth ideals, we begin by introducing Algorithm 3 which computes actions of prime ideal.

Algorithm 3 `Action of a prime ideal`

Input : $(E, \iota) \in SS_{\mathfrak{O}}(p)$ an oriented supersingular elliptic curve with $\mathfrak{O} = \mathbb{Z}[\omega]$ and $\mathfrak{l} =< \ell, \beta >$ an invertible ideal of $\mathfrak{O}$ of prime norm $\ell \neq p$.

Output : $[\mathfrak{l}] \star (E, \iota)$ and $P \in E$ such that $\langle P \rangle = E[\mathfrak{l}]$ and a representation of $\varphi_{\mathfrak{l}}$.

1: Compute $\{R, Q\}$ a basis of $E[\ell]$.
2: $\theta \leftarrow \iota(\beta)$.
3: Compute $\theta(R)$ and $\theta(Q)$.
4: **if** $\theta(R)$ (resp. $\theta(Q)$) is equal to zero **then**
5: $\quad P \leftarrow R$ (resp. Q).
6: **else**
7: $\quad$ Find $x \in [\![1, \ell[\![$ such that $x\theta(Q) = \theta(R)$.
8: $\quad$ Compute the generating point $P := R - xQ$ of the subgroup $E[\mathfrak{l}]$.
9: Get an efficient representation of the isogeny $\varphi_{\mathfrak{l}} : E \to E^{\mathfrak{l}}$.
10: Get an efficient representation of the composition of isogenies $\varphi := \varphi_{\mathfrak{l}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{l}}$.
11: Divide φ by $\deg(\varphi_{\mathfrak{l}})$ using Theorem 4.1.
12: Set $\iota' : \mathbb{Z}[\omega] \hookrightarrow \mathrm{End}(E), \omega \mapsto \varphi / \deg(\varphi_{\mathfrak{l}})$.
13: **return** $(E^{\mathfrak{l}}, \iota')$ and P.

Lemma 6.1 (Action of prime ideals)**.** *Algorithm 3 is correct and runs in time polynomial in $\ell, \log p$ and in the length of the representation of ι. The output representation of the orientation verifies*

- *It has size $O(\log^3(\ell^2 \deg(\iota(\omega))) \log p)$ bits,*
- *For any $P \in E'$, one can compute $\iota'(\omega)(P)$ in $\tilde{O}(\log^{11}(\ell^2 \deg(\iota(\omega))))$ operations over the field of definition of P.*

The representation of the isogeny $\varphi_{\mathfrak{l}}$ is given by a generating point of its kernel living in an extension of degree at most $O(\ell^2)$.

Proof. Most steps of the following algorithm are standard, see for instance [GPS20], the contribution here is to use Algorithm 1 to avoid degrading the quality of the representation after several action computations.

[1 - 3] The first line takes a runtime polynomial in $\log p$ and ℓ to return a basis living in an extension of degree at most $O(\ell^2)$. Then to evaluate $\theta(R)$ and $\theta(Q)$, we use the fact that ι is given in efficient representation thus line 3 can be computed in time polynomial in the length of the representation of ι, in ℓ and in $\log p$.

[4 - 8] The discrete logarithm problem of line 7 can be solved in $O(\sqrt{\ell})$ operations over $E[l]$. Once a solution x is found, we also get the point $R - xQ$ that generates $E[\mathfrak{l}] = E[\ell] \cap \ker \theta$ since it has order ℓ and it is in the kernel of θ. The same arguments hold to ensure that R or Q is a generator of $E[\mathfrak{l}]$ if $\theta(R) = 0$ or $\theta(Q) = 0$.

[9 - 10] The computation of an efficient representation of $\varphi_{\mathfrak{l}}$ is done using Vélu's formulae in $O(\ell)$ [Vé71]. To have an efficient representation of the orientation ι' it is enough to have an efficient algorithm to evaluate $\iota'(\omega)$, i.e. to have an efficient representation of

$$\frac{\varphi_{\mathfrak{l}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{l}}}{\deg(\varphi_{\mathfrak{l}})}.$$

Thanks to line 9 and the hypothesis on the inputs, an efficient representation of $\varphi := \varphi_{\mathfrak{l}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{l}}$ is given by the composition of the representations. It only remains to divide φ by $\deg(\varphi_{\mathfrak{l}})$ using Theorem 4.1.

[11 - 13] Let N be the degree of φ. By Theorem 4.1, it takes a runtime polynomial in $\log p$ and $\log N$ to give a representation of the endomorphism $\iota'(\omega)$ in $O(\log^3 N \log p)$ bits such that $\iota'(\omega)$ can be evaluated at a point in $\tilde{O}(\log^{11} N)$ operations over the field of definition of the point.

By summing the complexities of the different steps and as $N = \ell^2(\deg(\iota(\omega)))$, this algorithm runs in time polynomial in $\ell, \log p$ and in the length of the representation of ι. □

Theorem 6.2 proves the efficiency of the algorithm which uses Algorithm 3 as a subprocedure to compute action of smooth ideals.

Theorem 6.2 (Action of smooth ideals)**.** *For any imaginary quadratic order $\mathfrak{O} = \mathbb{Z}[\omega]$, there exists an algorithm that takes as input:*

- *An oriented elliptic curve $(E, \iota) \in SS_{\mathfrak{O}}(p)$ with an efficient representation of ι,*
- *An invertible $\mathfrak{O}$-ideal $\mathfrak{a}$ of B-smooth norm coprime to p,*

and returns $(E', \iota') := [\mathfrak{a}]\star(E, \iota)$ together with a representation of $\varphi_{\mathfrak{a}}$ in time polynomial in $B, \log p$ and in the length of the representation of ι. The output orientation representation verifies that

- *It has size $\tilde{O}(\log^3(B^2 \deg(\iota(\omega))) \log p)$ bits,*
- *For any $P \in E$, one can compute $\iota'(\omega)(P)$ in $\tilde{O}(\log^{11}(B^2 \deg(\iota(\omega))))$ operations over the field of definition of P.*

The representation of $\varphi_{\mathfrak{a}}$ is given as a formal composition of $O(\log N(\mathfrak{a}))$ isogenies of prime degree each of them represented by a point, living in an extension of degree at most $O(B^2)$, generating their respective kernel.

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. Then one can deduce 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 Lemma 6.1 and the compatibility of the group action, the action of $\mathfrak{a}$ over (E, ι) can be computed as successive actions of all its prime ideal factors in time polynomial in $B, \log p$ and in the length of the representation of ι. Simultaneously, using the same lemma, one can store each isogeny representation induced by the successive prime ideals to get a representation of $\varphi_{\mathfrak{a}}$. The claimed properties of the orientation representation are again given by Lemma 6.1. □

At last, we present an algorithm to compute the action of any given ideal in subexponential time. In prior literature, achieving such results necessitated to compute a powersmooth representative of the input ideal, and then to apply standard algorithms to compute its action. Yet, it was mandatory to use heuristics about the distribution of powersmooth ideal. Thanks to Theorem 6.2, we can now search for a smooth representative of the input and replace previous heuristics with rigorous results.

Theorem 6.3 (**GRH**, Action of ideals)**.** *Let $(E, \iota) \in SS_{\mathfrak{O}}(p)$ be an oriented elliptic curve and $\mathfrak{a}$ be an invertible $\mathfrak{O}$-ideal of norm coprime to p. There exists an algorithm that computes $[\mathfrak{a}] \star (E, \iota)$ together with an representation of $\varphi_{\mathfrak{a}}$ in time $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$ where l is the length of the input. The representation of $\varphi_{\mathfrak{a}}$ has size $(\log^3 N(\mathfrak{a}) \log p)$ and allows us to evaluate it on a point in $\tilde{O}(\log^{11} N(\mathfrak{a}))$ arithmetic operations over its field of definition.*

Proof. Let $z > 0$ and B be an integer greater than $L_{|\operatorname{disc}(\mathfrak{O})|}[\frac{1}{2}, z]$. First, we compute a representative ideal of $[\mathfrak{a}]$ in $\mathrm{Cl}(\mathfrak{O})$ of B-smooth norm using the standard algorithm 4, see for instance [CJS14, Algorithm 1].

This algorithm returns a representative B-smooth ideal of the class $[\mathfrak{a}]$ in expected time

$$L_{|\operatorname{disc}(\mathfrak{O})|}\left[\frac{1}{2}, z + o(1)\right] + L_{|\operatorname{disc}(\mathfrak{O})|}\left[\frac{1}{2}, \frac{1}{\sqrt{2}} + \frac{1}{4z} + o(1)\right].$$

Indeed, the complexity of the different steps is as following:

Algorithm 4 `Smooth representative ideal`

Input : $\mathfrak{a}$ an $\mathfrak{O}$-ideal, $z > 0$ and $B > L_{|\operatorname{disc}(\mathfrak{O})|}[\frac{1}{2}, z]$.
Output : $\mathfrak{b}$ an $\mathfrak{O}$-ideal of B-smooth norm such that $[\mathfrak{a}] \sim [\mathfrak{b}]$ in $\mathrm{Cl}(\mathfrak{O})$.

1: Set $S_B := \{[\mathfrak{p}] \in \mathrm{Cl}(\mathfrak{O})$ such that $\gcd(f_{\mathfrak{O}}, \mathfrak{p}) = 1$ and $N(\mathfrak{p}) \leq B$ prime, and their inverse$\}$.
2: smooth $\leftarrow$ false.
3: **while** smooth is false **do**
4: $\quad y \leftarrow \mathrm{Unif}\{y \in \mathbb{N}^{\#S_B}$ such that $||y||_1 = \lceil \log |\operatorname{disc}(\mathfrak{O})| \rceil\}$.
5: $\quad$ Compute $\mathfrak{c}$ the reduced ideal in the class of $\mathfrak{a} \cdot \prod_{[\mathfrak{p}] \in S_B} \mathfrak{p}^{y_{\mathfrak{p}}}$.
6: $\quad$ **if** there exists $x \in \mathbb{N}^{\#S_B}$ such that $\mathfrak{c} = \prod_{[\mathfrak{p}] \in S_B} \mathfrak{p}^{x_{\mathfrak{p}}}$ **then**
7: $\quad\quad w \leftarrow x - y$.
8: $\quad\quad \mathfrak{b} \leftarrow \prod_{[\mathfrak{p}] \in S_B} \mathfrak{p}^{(x-y)_{\mathfrak{p}}}$.
9: $\quad\quad$ smooth $\leftarrow$ true.
10: **return** $\mathfrak{b}$.

- At line 1, computing the set S_B takes time $\tilde{O}(B)$ and, at line 5, computing the reduced ideal takes time polynomial in $\log |\operatorname{disc}(\mathfrak{O})|$. For instance, both can be done using the well-known bijection between class group of an imaginary quadratic order of discriminant D and the group of primitive positive definite forms of discriminant D, as described in [BV07].
- At line 6, one can check if $\mathfrak{c}$ factors over S_B by checking the B-smoothness of its norm, hence it has an excepted running time $L_{|\operatorname{disc}(\mathfrak{O})|}[\frac{1}{2}, \frac{1}{\sqrt{2}} + o(1)]$ using Schnorr-Seysen-Lenstra probabilistic algorithm for instance.

Finally, under GRH, thanks to Proposition 2.5 and [Sey87, Proposition 4.4], the probability to draw a vector y such that the reduced ideal of its associated class is B-smooth is at least $L_{|\operatorname{disc}(\mathfrak{O})|}[\frac{1}{2}, -\frac{1}{4z} + o(1)]$. This complexity can be optimised by taking $z = \frac{\sqrt{2}+\sqrt{6}}{4}$, the representative ideal is then computed in expected time $L_{|\operatorname{disc}(\mathfrak{O})|}[\frac{1}{2}, \frac{\sqrt{2}+\sqrt{6}}{4} + o(1)]$.

This algorithm outputs a product of $O(\log |\operatorname{disc}(\mathfrak{O})|)$ prime ideals of norm lower than B. Then, by Theorem 6.2, the action $\mathfrak{a} \star (E, \iota)$ can be computed in time polynomial in $B, \log p$ and in the length of the representation of ι.

It only remains to compute a better representation of $\varphi_{\mathfrak{a}}$. Dividing $\varphi_{\mathfrak{a}}$ by 1 with Algorithm 1 provides an efficient representation of the isogeny $\varphi_{\mathfrak{a}}$. To this end, one needs first to evaluate $\varphi_{\mathfrak{a}}$ over the $p_i^{e_i}$-torsion subgroups of E for some powerprimes $p_i^{e_i}, i \in [\![1, s]\!]$, such that $p_1^{e_1} \dots p_s^{e_s} > 4 \deg \varphi_{\mathfrak{a}}$ with $(p_i, p \deg \varphi_{\mathfrak{a}}) = 1$. One can simply multiply the successive prime integers, coprime to $p \deg \varphi_{\mathfrak{a}}$, until their product is greater than $\deg 4\varphi_{\mathfrak{a}}$. Then the evaluation of $\varphi_{\mathfrak{a}}$, using the representation given by Theorem 6.2, over the torsion subgroups, including the computation of the basis of the torsion subgroups, takes a time polynomial in $B, \log p$ and in the length of the representation of $\varphi_{\mathfrak{a}}$, which is here $O(\log p \log^3(\deg \varphi_{\mathfrak{a}}))$ bits. Thus, by Theorem 4.16, applying Algorithm 1 to divide $\varphi_{\mathfrak{a}}$ by 1 using the N'' previously computed takes a time polynomial in $B, \log p$ and in $\log \deg \varphi_{\mathfrak{a}}$ and provide an efficient representation of $\varphi_{\mathfrak{a}}$ with the claimed properties. □

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

In this section, we prove, under GRH only, the complexity of a classical and a quantum resolutions 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. Finally, we present how algorithms of Section 4 can be used to naviguate efficiently in the oriented volcano of isogenies and how it can improve the resolution of $\mathfrak{O}$-Vectorisation.

7.1. Classical algorithm.

Currently, the best complexity we can expect for a classical algorithm solving $\mathfrak{O}$-Vectorisation 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]. Preceding 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 where similar results are in fact proven for smooth ideals. Thanks to Theorem 6.2, 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 [Wes22].

Problem 7.1 (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 5 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 5 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').

$x\leftarrow\lceil\log^{2+\varepsilon}|\operatorname{disc}(\mathfrak{O})|\rceil$.
$S_x\leftarrow\{[\mathfrak{p}]\in\operatorname{Cl}(\mathfrak{O})$, such that $\gcd(f_{\mathfrak{O}},\mathfrak{p})=1$ and $N(\mathfrak{p})\le x$ prime, and their inverse$\}$.
$T[\texttt{enc}((E,\iota))]\leftarrow(1)$.
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 $T[\texttt{enc}(\mathfrak{a}\star(E,\iota))]$ is empty **then**
    $T[\texttt{enc}(\mathfrak{a}\star(E,\iota))]\leftarrow\mathfrak{a}$.
$\mathfrak{a}\leftarrow(1)$.
while $T[\texttt{enc}(\mathfrak{a}\star(E',\iota'))]$ 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 $\bar{\mathfrak{a}}T[\texttt{enc}(\mathfrak{a}\star(E',\iota'))]$.

Lemma 7.2 (**GRH**)**.** *Algorithm 5 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 dictionnary structure for the table T, one can add and search for elements in time $O(\log\#T)$. Using the standard result $\#\operatorname{Cl}(\mathfrak{O})=O(\log(|\operatorname{disc}(\mathfrak{O})|)\sqrt{|\operatorname{disc}(\mathfrak{O})|})$, insertion and search 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.

[1-3] Those steps are polynomial in $\log^{2+\varepsilon}\operatorname{disc}(\mathfrak{O})$.

[4-8] It is expected that this first while loop will end after $O(\sqrt{\#\operatorname{Cl}(\mathfrak{O})})$ iterations. Indeed, by Proposition 2.5, one can expect to add a new element to the table T after at most 2 draws of random smooth ideals.

By Theorem 6.2, computing the action of $\mathfrak{a} \star (E, \iota)$ is done in polynomial time in the length of the representation of ι, in $\log p$ and in $\log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|$. In particular, the length of the representation is in $\tilde{O}((l_1 \log^2 x)^3 \log p)$ bits, where l_1 is the length of the representation of ι. Thus, it is done in time polynomial in l_1, $\log p$ and $\log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|$.

[10-12] This while loop is also expected to end after $O(\sqrt{\#\operatorname{Cl}((\mathfrak{O})})$ iterations, since, thanks again to Proposition 2.5, each iteration has a probability of success greater than $\frac{1}{2\sqrt{\#\operatorname{Cl}(\mathfrak{O})}}$.

Moreover, as in the first loop, using Theorem 6.2, 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 $(l \log p \log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|)^{O(1)} \sqrt{\#\operatorname{Cl}(\mathfrak{O})}$, where $l := \max\{l_1, l_2\}$. 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 7.3. Algorithm 5 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 betwen ordinary elliptic curves in [BS12]. This is not detailed here, since we focus on rigorously proven complexities and it might be difficult to avoid heuristics in the analysis of such algorithms.

Algorithm 5 is a central subprocedure in our classical resolution of $\mathfrak{O}$-Vectorisation and so of α-EndRing too. Furthermore, it is central in the complete resolution of Effective $\mathfrak{O}$-Vectorisation. These applications of Algorithm 5 require to move from one orbit to the other one using the $\mathfrak{O}$-twists.

Theorem 7.4 (**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, notice that p does not split over K otherwise $SS_{\mathfrak{O}}(p)$ would be empty [Onu21, Proposition 3.2].

By [ACL+22a, Theorem 4.4], if p is inert in K, then the action of $\operatorname{Cl}(\mathfrak{O})$ has only one orbit. Thus by running Algorithm 5 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 ramified in K, again by [ACL+22a, Theorem 4.4], the action of $\operatorname{Cl}(\mathfrak{O})$ has two orbits. We then run two instances of Algorithm 5 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 5 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 Theorem 6.2 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 7.5 (**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 2.8. Let O be the orbit of (E', ι'). From the proof of [Onu21, Proposition 3.3], we know that (E, ι) or its $\mathfrak{O}$-twist $(E, \bar{\iota})$ is in O. Thus, by running two instances of Algorithm 5 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 7.2. □

Proof of Theorem I. 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 $\ln |\operatorname{disc}(\mathbb{Z}[\alpha])|$. Then, by Corollary 5.2, 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 3.6. Finally, by Theorem 7.4 and since $|\operatorname{disc}(\mathbb{Z}[\alpha])|$ is greater than $|\operatorname{disc}(\mathfrak{O})|$, the $\mathfrak{O}$-VECTORISATION problem can be solved in $l^{O(1)} |\operatorname{disc}(\mathbb{Z}[\alpha])|^{1/4}$. □

7.2. **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 7.6 (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 $k, n_1, \ldots n_k$. 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 the Kuperber's quantum algorithm to solve it in an subexponential number of queries of the black-box functions f_0 and f_1.

Theorem 7.7 (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 6 which solves $\mathfrak{O}$-VECTORISATION assuming that the two input curves are in the same orbit. This algorithm is analogous to [CJS14, Algorithm 3].

Lemma 7.8 (**GRH**)**.** *The Algorithm 6 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.*

Algorithm 6 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 $\varphi_{\mathfrak{a}}$.

Proof. Let us prove the complexity of Algorithm 6:

[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 Kuperber's algorithm, Theorem 7.7, 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 $l^{O(1)} L_{\operatorname{disc}(\mathfrak{O})}[1/2]$ by Theorem 6.3, where l is the length of the input. Thus this step in done in $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})}[1/2]$.

[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 $\# \mathrm{Cl}(\mathfrak{O}) = O(\ln(|\operatorname{disc}(\mathfrak{O})) \sqrt{\operatorname{disc}(\mathfrak{O})})$, this whole step is done in time polynomial in $\log |\operatorname{disc}(\mathfrak{O})|$.

[5] Finally with Theorem 6.3, we can compute the isogeny $\varphi_{\mathfrak{a}} : (E_0, \iota_0) \to (E_1, \iota_1)$ in $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$.

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 $b = (b_1, \dots, b_k) \in \mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_k\mathbb{Z}$ 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 7.9 (**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 6. 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). Then the complexity in the Theorem 7.9 directly comes from Lemma 7.8. □

This leads us to the following proof of Theorem II.

Proof of Theorem II. By Corollary 5.2 and because the factorisation of $\mathrm{disc}(\mathbb{Z}[\alpha])$ 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 3.6, α-EndRing reduces to $\mathfrak{O}$-Vectorisation in probabilistic time polynomial in the length of the input. Hence, by Theorem 7.9, α-EndRing can be solved in expected time $l^{O(1)}L_{|\operatorname{disc}\mathbb{Z}[\alpha]|}[1/2]$. □

7.3. **Optimising by climbing the volcano.** We fix K to be a quadratic number field and we consider supersingular elliptic curves over the finite field $\mathbb{F}_{p^2}$. Let $\ell \neq p$ be a prime number.

Adding K-orientations to a ℓ-isogeny graph of supersingular elliptic curves gives a structure of volcano to each of its connected component which is analogous to the structure of isogeny graphs of ordinary elliptic curves. We now introduce formally this notion before to show how results of Section 6 can be used to naviguate efficiently in this volcano and to optimise previous results of this section.

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 primite $\mathfrak{O}'$-orientation. For any K-oriented isogeny $\varphi:(E,\iota)\to(E',\iota')$ of degree ℓ, we say that φ is

$$\begin{aligned}\nearrow&\ \textbf{ascending} \text{ if } \mathfrak{O}\subsetneq\mathfrak{O}',\\ \to&\ \textbf{horizontal} \text{ if } \mathfrak{O}=\mathfrak{O}',\\ \searrow&\ \textbf{descending} \text{ if } \mathfrak{O}\supsetneq\mathfrak{O}'.\end{aligned}$$

From [CK20], where $\left(\frac{\mathrm{disc}(\mathfrak{O})}{\ell}\right)$ is the Legendre Symbol, we know that (E,ι) always 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 infite 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 made possible thanks to Section 6.

Lemma 7.10 (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 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 [Wes22, 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. (This computation is similar to the steps from 1 to 9 of Algorithm 3 except that $\theta = \iota(\omega)$.) 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 4.1, 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 7.11 (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, [Wes22, Lemma 12]. Notice that to use directly [Wes22, 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 7.10, 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 7.10, 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 [Wes22, Lemma 12]. □

Corollary 7.12 (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 7.11 together with the complexity result on $\mathfrak{O}$-EndRing given by Theorem I. □

References

[ACL+22a] Sarah Arpin, Mingjie Chen, Kristin E. Lauter, Renate Scheidler, Katherine Stange, and Ha T. N. Tran. Orientations and cycles in supersingular isogeny graphs, 2022. Published: Cryptology ePrint Archive, Paper 2022/562.

[ACL+22b] Sarah Arpin, Mingjie Chen, Kristin E. Lauter, Renate Scheidler, Katherine E. Stange, and Ha T. N. Tran. Orienteering with one endomorphism. *arXiv e-prints*, page arXiv:2201.11079, January 2022.

[Bis12] Gaetan Bisson. Computing endomorphism rings of elliptic curves under the GRH. *Journal of Mathematical Cryptology*, 5(2), January 2012.

[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*, 2014.

[BKV19] Ward Beullens, Thorsten Kleinjung, and Frederik Vercauteren. CSI-FiSh: Efficient Isogeny based Signatures through Class Group Computations, 2019. Published: Cryptology ePrint Archive, Paper 2019/498.

[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, April 2012. arXiv:1101.0564 [cs, math].

[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*, SODA '16, pages 893–902, USA, 2016. Society for Industrial and Applied Mathematics. event-place: Arlington, Virginia.

[BV07] Johannes Buchmann and Ulrich Vollmer. *Binary quadratic forms: an algorithmic approach*. Number v. 20 in Algorithms and computation in mathematics. Springer, Berlin ; New York, 2007.

[CD20] Wouter Castryck and Thomas Decru. CSIDH on the Surface. In Jintai Ding and Jean-Pierre Tillich, editors, *Post-Quantum Cryptography - 11th International Conference, PQCrypto 2020, Paris, France, April 15-17, 2020, Proceedings*, volume 12100 of *Lecture Notes in Computer Science*, pages 111–129. Springer, 2020.

[CD23] Wouter Castryck and Thomas Decru. An Efficient Key Recovery Attack on SIDH. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 423–447. Springer, 2023.

[CGL06] Denis Xavier Charles, Eyal Z. Goren, and Kristin E. Lauter. Cryptographic hash functions from expander graphs. *IACR Cryptol. ePrint Arch.*, page 21, 2006.

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

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

[Cou06] Jean-Marc Couveignes. Hard Homogeneous Spaces, 2006. Published: Cryptology ePrint Archive, Paper 2006/291.

[CPV20] Wouter Castryck, Lorenz Panny, and Frederik Vercauteren. Rational isogenies from irrational endomorphism. *dvances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 12106:523–548, 2020.

[CS21] Mathilde Chenu and Benjamin Smith. Higher-degree supersingular group actions, 2021. _eprint: 2107.08832.

[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 \(\mathbb{F}\)\(_\mboxp\). *Des. Codes Cryptogr.*, 78(2):425–440, 2016.

[EHL+18] Kirsten Eisenträger, Sean Hallgren, Kristin Lauter, Travis Morrison, and Christophe Petit. Supersingular isogeny graphs and endomorphism rings: reductions and solutions. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2018 Proceedings*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pages 329–368, Germany, 2018. Springer Verlag.

[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, 2023. Published: Cryptology ePrint Archive, Paper 2023/058.

[GPS20] Steven D. Galbraith, Christophe Petit, and Javier Silva. Identification Protocols and Signature Schemes Based on Supersingular Isogeny Problems. *J. Cryptol.*, 33(1):130–175, 2020.

[Kan97] Ernst Kani. The number of curves of genus two with elliptic differentials. 1997(485):93–122, 1997.

[Kup05] Greg Kuperberg. A Subexponential-Time Quantum Algorithm for the Dihedral Hidden Subgroup Problem. *SIAM J. Comput.*, 35(1):170–188, 2005.

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

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

[Mil86] J S Milne. Abelian Varieties. 1986.

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

[PT18] Paul Pollack and Enrique Trevino. FINDING THE FOUR SQUARES IN LAGRANGE'S THEOREM. 2018.

[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, 2022. Published: Cryptology ePrint Archive, Paper 2022/1068.

[Rob22b] Damien Robert. Some applications of higher dimensional isogenies to elliptic curves (preliminary version), 2022. Published: Cryptology ePrint Archive, Paper 2022/1704.

[Rob23] Damien Robert. Breaking SIDH in Polynomial Time. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 472–503. Springer, 2023.

[RS06] Alexander Rostovtsev and Anton Stolbunov. PUBLIC-KEY CRYPTOSYSTEM BASED ON ISOGENIES, 2006. Published: Cryptology ePrint Archive, Paper 2006/145.

[Sey87] Martin Seysen. A probabilistic factorization algorithm with quadratic forms of negative discriminant. *Mathematics of computation*, 48(178):757–780, 1987.

[Sil13] Joseph H. Silverman. Elliptic curves. In Gary L. Mullen and Daniel Panario, editors, *Handbook of Finite Fields*, Discrete mathematics and its applications, pages 422–439. CRC Press, 2013.

[Voi21] John Voight. *Quaternion Algebras*, volume 288 of *Graduate Texts in Mathematics*. Springer International Publishing, Cham, 2021.

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

[Wes21] Benjamin Wesolowski. The supersingular isogeny path and endomorphism ring problems are equivalent. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 1100–1111. IEEE, 2021.

[Wes22] Benjamin Wesolowski. Orientations and the Supersingular Endomorphism Ring Problem. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology – EUROCRYPT 2022*, volume 13277, pages 345–371. Springer International Publishing, Cham, 2022.

[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, June 1974.

ENS de Lyon, LIP, UMR 5668 (U. Lyon, ENS de Lyon, Inria, UCBL), France

ENS de Lyon, CNRS, UMPA, UMR 5669, Lyon, France