Abstract

The emergence of quantum computers constitutes a challenge for cryptography; we need to develop new computational problems that are presumably hard for quantum computers, together with new schemes relying on them. *Post-quantum cryptography* is the domain of research investigating schemes resistant to quantum adversaries.

In this thesis, we explore the foundations of the *isogeny-based* candidate for post-quantum cryptography. The security of isogeny-based schemes initially relies on the difficulty of the ISOGENY problem, which asks one to compute a “nice” map, called an isogeny, between two given elliptic curves. During the development of this field, several other hard problems have arisen, such as variants of the ISOGENY problem or of the problem of computing endomorphism rings (an endomorphism being an isogeny from a curve to itself). It has been proven, first under heuristic assumptions, then under the generalised Riemann hypothesis, that these problems are equivalent. We provide an *unconditional* equivalence result. In addition, we prove worst-case to average-case reductions, showing that assuming the existence of a single hard instance of the ISOGENY problem is enough to ensure that random instances of the other problems are hard. These advances greatly simplify the set of necessary assumptions to provide a secure setup for isogeny-based cryptography.

We also focus our attention on schemes adding data to elliptic curves, called orientations, to perform group actions. Having access to a practical group action framework for post-quantum cryptography is an important challenge for cryptographers as it could lead to many advanced protocols. We provide a rigorous complexity analysis of the underlying hard problems, substituting the previous heuristic assumptions in the literature with the generalised Riemann hypothesis. Then, we present the first practical and scalable post-quantum framework for effective group action: `PEGASIS`.

A central ingredient for these results is the higher dimensional machinery developed from the `SIDH` attack. This is crucial both for our theoretical and practical contributions. In this thesis, we participate in the development of these tools by generalising an efficient division algorithm for isogenies from a specific case in the literature.

Résumé

L'émergence des ordinateurs quantiques est un défi pour la cryptographie moderne. Les cryptologues doivent non seulement considérer de nouveaux problèmes calculatoires, supposés difficiles même pour l'algorithmique quantique, mais aussi construire des protocoles dont la sécurité repose sur ces derniers. Le domaine de recherche ayant pour objectif la conception de cryptosystèmes résistants à des adversaires quantiques est appelé *cryptographie post-quantique.*

Dans cette thèse, nous explorons les fondements d'un des candidats post-quantiques majeurs : *la cryptographie à base d'isogénies.* Sa sécurité repose initialement sur la difficulté du problème ISOGENY, consistant à trouver une "bonne" application, appelée isogénie, entre deux courbes elliptiques données. Le développement de ce domaine a mené à l'étude de nombreux autres problèmes : plusieurs versions du problème ISOGENY mais aussi du problème du calcul de l'anneau d'endomorphismes d'une courbe (un endomorphisme étant une isogénie d'une courbe vers elle-même). Il a été prouvé, d'abord sous des hypothèses heuristiques puis sous l'hypothèse de Riemann généralisée, que ces problèmes étaient équivalents. Dans ce travail, nous démontrons leur équivalence *inconditionnelle.* De plus, nous prouvons des réductions du pire cas au cas moyen qui montrent que l'existence d'une instance difficile du problème ISOGENY suffit à assurer la difficulté des instances aléatoires de n'importe quel autre problème. Ces avancées simplifient grandement l'ensemble des hypothèses nécessaires pour fournir un cadre sécurisé à la cryptographie à base d'isogénies.

Par ailleurs, nous nous intéressons aux protocoles qui considèrent des informations additionnelles sur les courbes elliptiques, appelées orientations, permettant de calculer des actions de groupe. Développer des algorithmes calculant des actions de groupe adaptées à la cryptographie post-quantique est un défi important puisque de nombreux protocoles avancés peuvent être construits à partir de ceux-ci. Nous fournissons une analyse rigoureuse de la complexité des problèmes difficiles sous-jacents, remplaçant les précédentes hypothèses heuristiques de la littérature par l'hypothèse de Riemann généralisée. Puis, nous présentons le premier algorithme pratique d'actions de groupe post-quantique adaptable à des niveaux de sécurité élevés : `PEGASIS`.

Un ingrédient central de ces résultats originaux est l'utilisation d'isogénies en dimension supérieure, développée suite aux attaques de `SIDH`. Ces nouveaux outils sont cruciaux à la fois pour nos contributions théoriques et pratiques. Dans cette thèse, nous participons à leur développement en généralisant un algorithme de division d'isogénies à partir d'un cas particulier de la littérature.

Contents

Introduction

Historically and etymologically, cryptology is the science of secrecy. Nowadays, cryptography is used daily by a large proportion of the population — often without realising it — to secure their communications in many different ways. For most problems that might be encountered on a communication channel, cryptographers have developed specific protocols to address them.

A glimpse into the landscape of cryptography

The three central properties often required for a safe transfer of information are the following:

- **Confidentiality**: the information is only accessible to the authorised parties.
- **Authenticity**: the information originates from the correct sender.
- **Integrity**: the information has not been altered since it was sent.

Many protocols deployed to guarantee these properties use **keys**. For instance, to ensure confidentiality, we use **encryption schemes**. In these protocols, using an *encryption key*, one can make a message comprehensible only to people who know the *decryption key*. When the encryption key and the decryption key are different, we say that the scheme is **asymmetric**, otherwise we say that it is **symmetric**. In the latter case, parties first need to agree on a shared key via a **key exchange** scheme. In the asymmetric case, the encryption key is **public**, so anyone can encrypt messages. Then, the decryption key remains a **secret** shared by all the people allowed to read the messages.

To ensure authenticity, we typically use **digital signature** schemes, which are part of asymmetric cryptography. A secret key is used to sign files and a public key is used to verify the authenticity of the signature.

Finally, to ensure integrity, we rely on **hash functions**.

Proving security. In most cases, it is impossible to prove that someone without the secret key cannot read or sign a message. Indeed, one can only ensure **perfect security** in very few exceptional cases which are not the most suitable for today's real-world cryptography. By perfect security, we mean that no adversary, even with unbounded computational power, can break the security. For instance, the Vernam cipher, or one-time pad, offers perfect confidentiality but it requires, for each new message, sharing a new key which is as long as the message.

The solution of modern cryptography is to develop schemes ensuring a certain **level of security**, or bits of security. We say that a cryptosystem achieves a security level of λ if breaking it requires at least 2^λ operations. Then it is important to answer the following question:

- How can we ensure that there is no algorithm breaking the scheme in less than 2^λ operations?

Security reductions. The trust cryptographers have in the security of asymmetric cryptosystems, and certain hash functions, relies on their belief in the difficulty of specific problems. Instead of directly analysing the security of a given scheme — which is often difficult to extend to other protocols — we study standard hard computational problems related to the scheme. We say that a problem is **hard** if there is no efficient algorithm to solve it, i.e. no algorithm running in polynomial time in the length of the input. It is **easy** otherwise. The relation between computational problems and the security of schemes is established through **reductions**.

An **oracle** solving a given problem is a black box that instantly returns a solution to this problem. For two computational problems P and Q, we say that P **reduces** to Q (in polynomial time), denoted as

$$P \longrightarrow Q,$$

if one can solve the problem P (in polynomial time) given access to an oracle that solves the problem Q.

Let us denote by S the computational problem corresponding to breaking the security of some cryptosystem C — for instance, recovering the private key of an asymmetric scheme C. To prove its security, we consider sequences of reductions such as

$$P \longrightarrow S \longrightarrow Q,$$

where P and Q are two computational problems. In such a sequence, the problem Q offers an attack avenue for the security of the scheme C: if Q is easy then C is broken. While the problem P provides a lower bound on the difficulty to break the scheme C: if P is hard then C is secure.

The best situation is when the problems P and Q are the same. In this case, the security of the scheme C truly corresponds to the difficulty of the problem Q. We say that S is **equivalent** to the problem P.

The quantum threat. In his article [Sho94], Peter Shor demonstrated that quantum algorithms can theoretically solve the two most widespread problems of modern cryptography: the discrete logarithm problem in finite abelian groups, or DLP, and the factorisation problem for large integers. Together, they underlie most modern public key cryptography; for instance, elliptic curve cryptography for the former and RSA (Rivest–Shamir–Adleman) cryptosystem for the latter. The DLP has a central place in this thesis as we shall consider a generalisation of this problem where the groups are replaced by group actions. Fortunately, real-world quantum computers were not powerful enough at that time — and this is still the case — to break the widely deployed cryptosystems. This has provided cryptographers with a crucial window of opportunity to design quantum-resistant cryptography; this area of research is called **post-quantum cryptography**.

The isogeny-based candidate

In this thesis, we explore the foundations of the post-quantum **isogeny-based** candidate. An isogeny is a map that preserves both the algebraic and geometric structures of elliptic curves. In particular, they are given by rational maps. Elliptic curves themselves are well studied objects in both cryptography and mathematics. They are the simplest examples of abelian varieties, which are projective varieties characterised by their additional abelian group structure.

In cryptography, elliptic curves have proven themselves to be a highly suitable framework for many protocols. Indeed, they behave almost like generic groups — groups with no extra structure, in which only group operations can be performed. In addition, they have a compact description and efficient group operations. This offers a powerful setup for cryptography based on the discrete logarithm problem. Unfortunately, they are not immune to attacks based on Shor's algorithm. This is where isogenies come into play.

The idea of using isogenies constructively emerged around 2006, and took two different directions. One relies on *supersingular* elliptic curves, the other on *ordinary* elliptic curves. This distinction between supersingular and ordinary elliptic curves comes down to their endomorphism ring, an endomorphism being an isogeny from a curve to itself. The endomorphism rings of ordinary elliptic curves are commutative, while they are not for supersingular elliptic curves.

The Charles–Goren–Lauter hash function. In [CGL06, CLG09], Charles, Goren and Lauter introduced a hash function, known as the `CGL` hash function, that exploits the properties of *ℓ-isogeny graphs* for a small prime ℓ. In these graphs, vertices correspond to supersingular elliptic curves defined over the finite field $\mathbb{F}_{p^2}$ with p^2 elements, for p a large prime, while edges correspond to isogenies of degree ℓ, also known as ℓ-isogeny. The degree of an ℓ-isogeny is defined as the cardinality of its kernel. This is an important data since the complexity to compute isogenies increases linearly with their degree, including the space complexity.

The key feature of ℓ-isogeny graphs is their rapid mixing: these graphs are *Ramanujan*, which corresponds to *optimal expander graphs.* This implies that, despite the large number of vertices (around p vertices) a short random walk in the graph (at most $O(\log p)$ steps) lands on any vertex with approximately the same probability.

The `CGL` hash function converts a bit string into an elliptic curve by taking a path corresponding to this bit string in the 2-isogeny graph. The hard problem, on which the security of this hash relies, consists in finding a path between vertices. In other words, finding a chain of ℓ-isogenies between two given supersingular elliptic curves. This problem is called the ℓ-IsogenyPath problem. Since its introduction, all known algorithms solving it run in exponential time in the length of the input.

The Couveignes–Rostovtsev–Stolbunov key exchange. In 2006, two independent papers introducing a key exchange protocol based on isogenies were published. Today, they are collectively referred to as the `CRS` key exchange. The first one, [Cou06], was originally written in 1997 by Couveignes but remained unpublished until 2006. The second one was written by Rostovtsev and Stolbunov [RS06].

The core idea of the `CRS` key exchange is to exploit well-known group actions from ideal class groups on sets of ordinary elliptic curves. This group action satisfies suitable properties to serve, in a way, as a proxy for abelian groups. Hence, it allows us to translate cryptography based on the discrete logarithm problem into a group action setting. In particular, one can perform a Diffie-Hellman-like key exchange — this is what the `CRS` key exchange is about.

In his paper, Couveignes formalises this notion of group action for cryptography as *Hard Homogeneous Spaces.* In this thesis, we adopt the more recent *Effective Group Action* point of view, stated in [ADMP20], fulfilling the same purpose. We call the analogous of the discrete logarithm problem in these frameworks the Vectorisation problem, following Couveignes' terminology. In contrast to the DLP, it is not broken by quantum computers. Nevertheless, the best known quantum algorithms to solve this problem run in subexponential time, hence this problem is asymptotically easier to solve than the ℓ-IsogenyPath problem. Actually, Vectorisation formally reduces to the general Isogeny problem, which asks for any isogeny between two elliptic curves. Note that Isogeny itself reduces to ℓ-IsogenyPath.

The main issue with the CRS cryptosystem is its impractical running time. In [CLM+18], a practical version is developed by moving from ordinary elliptic curves to supersingular ones defined over $\mathbb{F}_p$: the CSIDH key exchange scheme. It is worth mentioning that acting with ideal class groups on supersingular elliptic curves is not as well-studied as for ordinary elliptic curves. In the literature of isogeny-based cryptography, we refer to such a framework as **orientations**. It has been formalised relatively recently in the articles [CK20, Onu21].

The foundations of isogeny-based cryptography. Since the introduction of the CRS and CGL cryptosystems, numerous isogeny-based schemes have been developed, from digital signatures [DKL+20] to asymmetric encryption [MOT20] and more advanced protocols [KLLQ23, BKM+21]. Most of them rely on supersingular elliptic curves since they offer better efficiency; this is why this thesis focuses on them. Through the study of their security, other related hard problems have emerged.

One of the articles that played a crucial role in the study of the foundations of isogeny-based cryptography is [KLPT14], where the authors explore the ℓ-IsogenyPath problem from an original point of view. This perspective is based on the Deuring correspondence [Deu41], which establishes that the endomorphism rings of supersingular elliptic curves are isomorphic to maximal orders in certain quaternion algebras. This correspondence extends by associating isogenies to ideals of maximal orders. In [KLPT14], they provided a practical algorithm to solve the analogue of the ℓ-IsogenyPath problem in the quaternion world; this algorithm is now known as the KLPT algorithm. From this result, they identified the EndRing problem, which asks for the endomorphism ring of an elliptic curve, as a hard problem to which the ℓ-IsogenyPath problem reduces.

One can also consider the more abstract problem of finding the quaternion structure of the endomorphism ring. We call this problem the MaxOrder problem, since it requires computing a maximal order isomorphic to the endomorphism ring of a given curve.

Multiple articles have explored the deep connections between the Isogeny, MaxOrder, and EndRing problems. They were shown to be equivalent under heuristic assumptions in [EHL+18], then under the generalised Riemann hypothesis (**GRH**) in [Wes22b].

The introduction in [PW24] of the OneEnd problem — a weaker version of EndRing asking for a single endomorphism — provided the first unconditional reduction from EndRing to Isogeny. An important preliminary result proven in the same article is the unconditional equivalence between OneEnd and EndRing.

The fall of SIDH and the rise of higher dimensions

Another crucial cryptosystem in understanding the development of isogeny-based cryptography is the SIDH key exchange protocol. It was introduced in [JD11], as a Diffie-Hellman-like key exchange based on isogenies between supersingular elliptic curves, in order to provide a secure protocol with good performance relying on the Isogeny problem.

Nevertheless, performing a Diffie-Hellman key exchange with isogenies is not trivial: it requires to make isogenies "commute". In CRS, this is done by using an abelian group action structure, at the cost of security; the best attacks are no longer exponential but subexponential. Here, the idea is to provide additional information, given by the evaluation of isogenies on some points, so one can compute pushforwards of isogenies. Morally, it means that given two isogenies φ and ψ from the same elliptic curve, one can "compose" φ with ψ or ψ with φ and obtain the same result.

Unfortunately for SIDH, a decade after its publication, it was shown in [CD23, MMP^{+}23, Rob23a] that the additional information provided was sufficient to solve the Isogeny problem efficiently, thereby completely breaking the scheme. Fortunately for isogeny-based cryptography, these attacks introduced abelian varieties and their higher dimensional isogenies as powerful new constructive tools for the field. A significant number of new protocols [BDD^{+}24, BM25, Ler25], as well as improvements in our algorithmic tools [PR23, ON24], has arisen from them.

In Chapter 2, we present three such tools, including an original contribution generalising a result from [Rob22b]: an unconditional algorithm to divide isogenies efficiently. It is unconditional in the sense that its complexity is proven without assuming any heuristics. In addition, this algorithm applies to arbitrary divisors. Prior to this result, it was only possible to divide by integers with small prime-power factors. This result is part of the publication:

[HW25a] Arthur Herlédan Le Merdy and Benjamin Wesolowski. The supersingular endomorphism ring problem given one endomorphism. *IACR Communications in Cryptology*, 2(1), 2025.

Together with the two other applications — an efficient "Interpolation" algorithm to compute an isogeny from its image on a large enough subgroup [Rob24] and the CLAPOTI algorithm to compute any group actions given by orientations efficiently [PR23] — these results play a crucial role in the rest of our work. In particular, they relax the previous dependence on **GRH** or on heuristic assumptions.

Contributions and overview

Unconditional foundations. Understanding the foundations of isogeny-based cryptography is crucial for ensuring the security of the cryptosystems. The methods deployed for this aim might also provide new directions and ideas for practical constructions or attacks.

As mentioned before, several articles explore the foundations of isogeny-based cryptography. In the state of the art preceding this thesis, the hard problems at the heart of the field — EndRing, Isogeny, MaxOrder and OneEnd — were proven to be equivalent under the generalised Riemann hypothesis. In Chapter 3, we prove their unconditional equivalence. We also add the HomModule problem — which asks to find the homomorphism module between two supersingular elliptic curves — to this list. This new hard problem is considered for the sake of completeness, but also because it is related to the security of some schemes, such as the SQIsign signature scheme.

This material comes from the unpublished paper under review:

[HW25b] Arthur Herlédan Le Merdy and Benjamin Wesolowski. Unconditional foundations for supersingular isogeny-based cryptography. Cryptology ePrint Archive, Paper 2025/271, 2025.

In order to establish these equivalences without relying on **GRH**, original techniques had to be developed. For instance, we do not have access to the KLPT algorithm — which is only proven under **GRH** [Wes22b] — to perform the reduction from Isogeny to EndRing. Working without **GRH** has also denied us access to certain simple models of quaternion algebra, leading us to generalise the MaxOrder problem to more general quaternion algebras.

Unifying the set of hard problems on which relies the security of isogeny-based cryptography is not sufficient to truly understand the foundations of the field. What if the Isogeny problem is hard in general, but the specific instances encountered in the schemes are easy to solve? For example, solving the ℓ-IsogenyPath is difficult in general, however, finding an ℓ-isogeny between two neighbors in the ℓ-isogeny graph is trivial.

The instances found in cryptosystems are usually coming from random walks in ℓ-isogeny graphs. To demonstrate that assuming the difficulty of the ℓ-IsogenyPath problem in the **worst-case** is sufficient to ensure the difficulty of such instances — corresponding to the ℓ-IsogenyPath problem in the **average-case** — we rely on a worst-case to average-case reductions. For the (ℓ-)Isogeny(Path) and the EndRing problems, such self-reductions are folklore results. In Section 3.4, we prove a more general statement: any hard problem in the worst-case reduces to any other hard problem in the average-case. This implies that assuming the existence of a single hard instance of the Isogeny problem, for instance, is enough to guarantee the hardness of the random instances of any other problem. This result is also coming from [HW25b].

Effective group action from orientations. As mentioned earlier, the `CSIDH` key exchange protocol relies on group actions that can be studied via the orientation framework. In Chapter 4, we explore both the theoretical and practical aspects of "oriented" isogeny-based cryptography.

Our main theoretical contribution starts from the following observation: knowing an orientation of a supersingular elliptic curve is equivalent to knowing a non-trivial endomorphism. For `CSIDH` this endomorphism is the well-known Frobenius endomorphism. Since the security of `CSIDH`-like protocols (numerous cryptosystems have been developed from `CSIDH` and orientations) reduces to the EndRing problem, we need to ask ourselves the following question:

- How does the EndRing problem become easier when a non-trivial endomorphism is given?

We answer this question in Section 4.2, by providing rigorous complexity analyses of the EndRing problem when a non-trivial endomorphism is given. In particular, we demonstrate that the classical security of protocols based on oriented elliptic curves is exponential, while their quantum security is subexponential. Previous general results were only heuristic, see [Wes22a]. Nevertheless, not every endomorphism provides an orientation suitable to construct group actions we are interested in for cryptographic purposes. This orientation needs to be **primitive**. We show that obtaining a primitive orientation from a non-trivial endomorphism can be done mostly at the cost of factoring an integer. This problem, called the Primitivisation problem, was introduced in [ACL+23] as a difficult problem even for quantum computers. These results have been published in:

[HW25a] Arthur Herlédan Le Merdy and Benjamin Wesolowski. The supersingular endomorphism ring problem given one endomorphism. *IACR Communications in Cryptology*, 2(1), 2025.

On the other hand, we contribute to the construction of one of the first practical effective group actions. Indeed, the framework offered by `CSIDH` is not an *effective group action*, since it is only possible to compute the action of a *restricted* set of group elements. This is, unfortunately, a concrete limitation when it comes to building advanced protocols on top of the `CSIDH` framework. Since its introduction, many research efforts [BKV19, DFK$^+$23, CLP24] have been conducted to provide a practical effective group action without any limitation. The main issue with these constructions is the lack of scalability, due to a very costly precomputation step. From the recent `CLAPOTI` algorithm [PR23], which computes the action of any ideal in polynomial time, we develop the `PEGASIS` algorithm. It is currently the most practical effective group action framework, achieving the highest level of security and the best performance. This promising construction should be suitable for more advanced protocols in isogeny-based cryptography. This work is part of the paper:

[DEF$^+$25a] Pierrick Dartois, Jonathan Komada Eriksen, Tako Boris Fouotsa, Arthur Herlédan Le Merdy, Riccardo Invernizzi, Damien Robert, Ryan Rueger, Frederik Vercauteren, and Benjamin Wesolowski. PEGASIS: Practical effective class group action using 4-dimensional isogenies. Cryptology ePrint Archive, Paper 2025/401, 2025. Accepted in the conference Advances in Cryptology – CRYPTO 2025 of the IACR.

Unconditional, rigorous and heuristic algorithms

As mentioned earlier, in this work, we improve some heuristic algorithms by making them rigorous. Sometimes, we also provide original unconditional proofs for results previously proven under **GRH**. This terminology used to qualify algorithms and their proof is not standardised in the literature. Let us clarify the meaning behind each word.

It is not uncommon in cryptography, and more generally in computational number theory, to make certain *ad hoc* assumptions about the behaviour of algorithms in order to analyse their correctness or complexity. For instance, one might assume that an integer output by a subprocedure behaves like a random integer of the same size. An algorithm whose correctness or complexity is proven under such assumptions is referred to as a **heuristic algorithm** and the corresponding assumptions are called **heuristic** assumptions.

Heuristic assumptions should not be confused with conjectures. Both are unproven claims; however, they do not serve the same purposes. The main objective of heuristics is to simplify the analysis of an algorithm. They are designed for a specific context and typically assert only that an algorithmic step behaves “well enough”. Hence proving them is often not an objective in itself; unlike for conjectures. In fact, heuristic assumptions are not necessarily believed to be strictly true; they might only be an approximation of reality. In contrast, conjectures are widely believed to be true — at least by part of the community. Returning to the earlier example, we may not genuinely expect the subprocedure to output uniformly random integers, nevertheless, we assume its output is sufficiently random to ensure that the algorithm performs well in practice.

A key part of the contributions in this thesis is to improve our trust in certain proofs by reducing the reliance on assumptions. In some cases, we replace heuristic algorithms with ones having complexity and correctness proven under the well-established generalised Riemann hypothesis. This yields what we call **rigorous** algorithms. In other cases, we show that some rigorous algorithms in the literature can, in fact, be made **unconditional**, in the sense that they do not rely on conjectures like **GRH**.

Chapter 1

Background

This chapter is dedicated to providing the necessary background on elliptic curves, isogenies and cryptography essential for the work presented in this thesis.

1.1 Definitions and notation

We start by introducing general definitions and notation that will be used throughout this work. We write

- $\mathbb{Z}$ for the ring of integers, $\mathbb{Z}_{>0}$ (resp. $\mathbb{Z}_{\geq 0}$) for the set of positive (resp. non-negative) integers,
- $\mathbb{Z}/n\mathbb{Z}$ for the ring of integers modulo the integer n,
- $\mathbb{Q}$ for the field of rational numbers, $\mathbb{R}$ for the field of real numbers,
- $\mathbb{F}_q$ for the finite field with q elements, for any prime power q,
- $[F' : F]$ for the degree of the field extension F' of F,
- $\bar{K}$ for the algebraic closure of a field K,
- $A \otimes_R B$ for the tensor product of two R-modules A and B,
- $\#S$ the cardinality of a set S,
- $\langle a_1, \ldots, a_n \rangle$ for the set generated by the $\mathbb{Z}$-linear combinations, or group operation combinations depending on the context, of the elements $a_1, \ldots, a_n$,
- $\mathrm{N}(I) := \#(R/I)$ for the norm of the ideal I in a ring R,
- $R^\times$ for the group of invertible elements of a ring R,
- $M_2(R)$ for the ring of 2×2 matrices with coefficients in a ring R,
- log for the logarithmic function in base 2,
- $| \cdot |$ for the absolute value,
- $\lfloor x \rfloor$ for the greatest integer less than or equal to x,
- $\lceil x \rceil$ for the smallest integer greater than or equal to x,
- $[\![a, b]\!]$ for the set of integers between a and b, inclusive of both endpoints, for $a, b \in \mathbb{Z}$,
- (a, b) for the greatest common divisor of a and b, or for the pair consisting of a and b.

We use $x \leftarrow \mathrm{Unif}\, S$ in algorithms to denote that x is sampled uniformly at random from the set S. We use the standard O-notation together with the $\tilde{O}$ notation $\tilde{O}(g) = (\log(g))^{O(1)} \cdot O(g)$. Moreover $\mathrm{poly}(f_1, \dots, f_n)$ denotes a polynomial function in $f_1, \dots, f_n$, i.e. $(f_1 + \dots + f_n)^{O(1)}$ and $\mathrm{polylog}(f_1, \dots, f_n)$ denotes a polynomial function in the length of $f_1, \dots, f_n$, i.e. $(\log(f_1) + \dots + \log(f_n))^{O(1)}$. For subexponential complexities, we use the standard L-notation

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

where x, a, b are three positive real numbers with $a < 1$. We shall also use it when the constant b is unknown

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

By default, we express time complexity in terms of binary operations. It will be specified when this is not the case. We say that an algorithm is **efficient** if it runs in probabilistic polynomial time in the length of the input. A problem is considered to be **hard** if every efficient algorithm solving it has at most a negligible probability of success. Otherwise, we say it is an **easy** problem. When a problem P is said to **reduce** in polynomial time to another problem Q, it means that there exists a probabilistic algorithm solving P given access to an oracle solving Q such that the algorithm runs in time polynomial in the length of the input and the length of the oracle's output. On a more informal level, we call an algorithm **practical** if it is efficient and its concrete running time is suitable for real-world applications.

In this work, the **generalised Riemann hypothesis** (**GRH**) refers to the assertion that the Riemann hypothesis holds for Hecke L-functions. In the literature, the term "generalised Riemann hypothesis" something refers only to the case of Dirichlet L-functions, while the "extended Riemann hypothesis" typically refers to the Dedekind zeta function. The version of **GRH** assumed here, encompasses, in particular, both of these cases. The results proven under **GRH** do not require the full strength of the assumption, but only certain consequences of it.

Smooth and powersmooth integers are central in this work. We recall their definition and fix some notations for the rest of the thesis.

Definition 1.1.1 ((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 $\max_{i \in [\![1,r]\!]} \ell_i^{e_i}$.*

Finally, for any set $\mathfrak{B}$ of prime numbers, we say that n is $\mathfrak{B}$-smooth if all its prime factors are in $\mathfrak{B}$.

We now introduce precise notation for extension degrees.

Definition 1.1.2 (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 $E[m]$ are defined.

We sometimes refer to security levels using those defined by the National Institute of Standards and Technology (NIST) during its post-quantum cryptography standardisation [NIS16]. In particular, we refer to the lowest level — Level 1 (NIST-I) — which corresponds to a cryptosystem that is at least as hard to break as AES-128. This implies a security level requiring at least 2^{128} operations for a classical adversary and 2^{64} for a quantum adversary.

1.2 Cryptography based on group actions

One promising direction to achieve post-quantum security is to generalize cryptography based on groups to group actions. By considering a commutative group acting on a set that itself has no structure, we can translate the hard problems of cryptography based on groups into new ones that are conjectured to be quantum resistant. Before introducing group action cryptography, let us first recall the two main problems of group-based cryptography. The first one is the **D**iscrete **L**ogarithm **P**roblem (DLP) which asks to invert the exponentiation in a cyclic group.

Problem 1.2.1 (Discrete Logarithm Problem)**.** *Let G be a cyclic group of order n generated by an element g. Given the group elements g,g^a, with $a \in \mathbb{Z}/n\mathbb{Z}$, find the exponent a.*

Then the **D**iffie–**H**ellman **P**roblem (DHP) asks to recover the secret key of a Diffie–Hellman key exchange [DH76]; the steps of this key exchange are summarised in Figure 1.1.

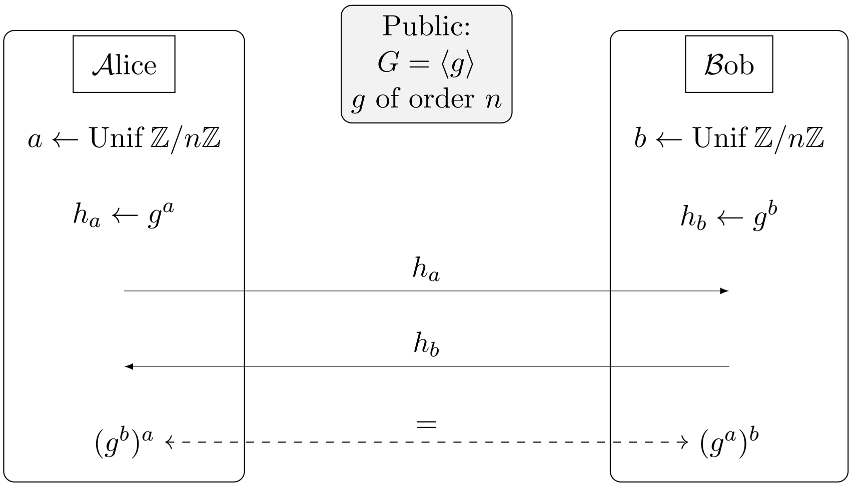


Figure 1.1: Classical Diffie–Hellman key exchange.

Problem 1.2.2 (Diffie–Hellman Problem)**.** *Let G be a cyclic group of order n generated by an element g. Given the group elements g,g^a,g^b, with a and b in $\mathbb{Z}/n\mathbb{Z}$, compute the element g^{ab}.*

There is a direct reduction from the DHP to the DLP. Since Shor's quantum algorithm [Sho94] efficiently solves the DLP, classical Diffie–Hellman key exchange based on groups are not secure against quantum adversaries. Nevertheless, moving to a framework based on group actions, (see Figure 1.2), prevents this attack.

We now define group actions and outline useful properties they can satisfy.

Definition 1.2.3 (Group action)**.** *We say that a tuple $(G, X, \star)$ is a* ***group action*** *when G is a group (denoted multiplicatively with the identity element e_G), X is a set and $\star : G \times X \to X$ is a map such that*

- *for any $x \in X$, we have $e_G \star x = x$,*
- *for any $g_1, g_2 \in G$ and $x \in X$, we have $g_1 \star (g_2 \star x) = (g_1 g_2) \star x$.*

We say that G acts on X. Additionally,

- *if for any $g \in G$, the existence of an element $x \in X$ such that $g \star x = x$ implies that $g = e_g$, we say that $(G, X, \star)$ is* ***free****,*
- *if for any $x_1, x_2 \in X$, there exists some $g \in G$ such that $g \star x_1 = x_2$, we say that $(G, X, \star)$ is* ***transitive****,*
- *if for any $x_1, x_2 \in X$, there exists a* unique *$g \in G$ such that $g \star x_1 = x_2$, we say that $(G, X, \star)$ is* ***regular****.*

In particular, a free and transitive group action is regular.

For cryptographic applications, we consider regular commutative group actions. Then the only difference with cryptography based group is that the group structure is, in a sense, separated from the set. Notice that free group actions that possess multiple orbits are also suitable for cryptography. Indeed, by restricting the action to a single orbit, one obtains a regular group action.

In the context of group actions, the DLP problem becomes the so-called Vectorisation problem, where instead of inverting a group operation, we ask for the inversion of a group action. This problem is is also known as the **G**roup **A**ction **I**nverse **P**roblem (**GAIP**), or as the group-action DLP. On the other hand, the DHP becomes the Parallelisation problem, or group-action Diffie–Hellman problem. We adopt the Vectorisation and Parallelisation names according to Couveignes' terminology in [Cou06].

Problem 1.2.4 (Vectorisation)**.** *Let $(G, X, \star)$ be a regular group action. Given two set elements $x_1, x_2 \in X$, compute the unique group element g such that $g \star x_1 = x_2$.*

Problem 1.2.5 (Parallelisation)**.** *Let $(G, X, \star)$ be an abelian regular group action. Given three set elements $x_0, x_1, x_2 \in X$, compute the unique set element $x_3 \in X$ such that $x_3 = (g_1 g_2) \star x_0$, where $g_1, g_2 \in G$ are the group elements such that $x_1 = g_1 \star x_0$ and $x_2 = g_2 \star x_0$.*

Shor's algorithm does not apply to the Vectorisation problem since it requires a group structure on the set elements. Nevertheless, quantum algorithms still solve this problem more efficiently than classical algorithms. As recalled in [ADMP20], the best known classical algorithm is a meet-in-the-middle approach which runs in $O(\sqrt{\#G})$ due to the birthday paradox, while it is possible to solve it quantumly in time subexponential in $\#G$ using a Kuperberg-like algorithm, see [CJS14]. In Section 4.2, we detail both results by providing rigorous proofs in the context of isogeny-based group action cryptography.

Another important advantage of group action cryptography over group cryptography comes from the relation between Vectorisation and Parallelisation. While, the DLP does not reduces to DHP, it has been proven that the Parallelisation problem quantumly reduces to the Vectorisation problem in several different settings, including in the context of isogeny-based cryptography [GPSV21, MZ22, Wes22a, GLM24]. As the other direction is trivial, these problems are morally equivalent.

This result offers strong foundations for group action cryptography. In particular, it implies that the security of Diffie–Hellman key exchange based on group actions is quantumly equivalent to the hardness of the Vectorisation problem.

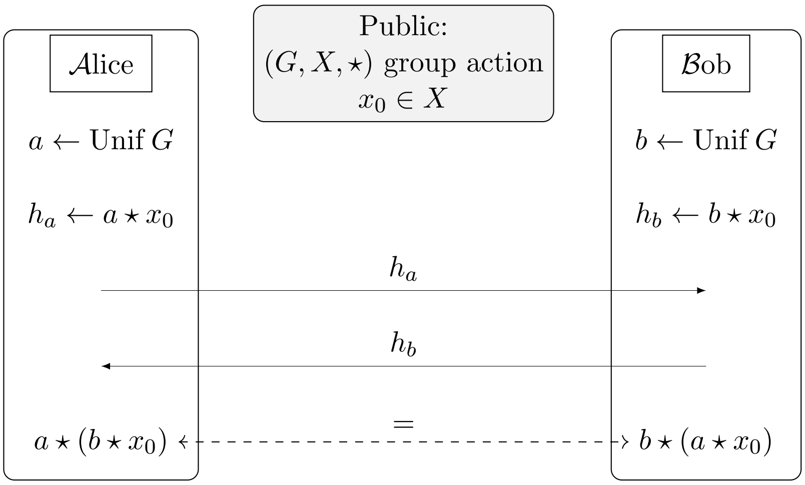

$\mathcal{A}$lice
Public:
$(G, X, \star)$ group action
$x_0 \in X$
$\mathcal{B}$ob
$a \leftarrow \mathrm{Unif}\, G$
$b \leftarrow \mathrm{Unif}\, G$
$h_a \leftarrow a \star x_0$
$h_b \leftarrow b \star x_0$
h_a
h_b
$=$
$a \star (b \star x_0)$
$b \star (a \star x_0)$

Figure 1.2: Diffie–Hellman key exchange based on group actions. The group action $(G, X, \star)$ is regular.

Different formalisms have emerged in the literature for group action based cryptography. One of the earliest articles exploring this algebraic structure for cryptographic purposes is [BY91]. The authors define "one-way group action" in order to construct bit commitment schemes. Another important reference is Couveignes' article [Cou06] which introduces **H**ard **H**omogeneous **S**pace (**HHS**) to propose authentication schemes and key exchange protocols which do not rely on the DLP or Factorisation problems. Notice, that none of these constructions was initially motivated by building post-quantum schemes. Here, we adopt the **E**fficient **G**roup **A**ctions (**EGA**) perspective from [ADMP20], which is itself based on the previously mentioned constructions.

Definition 1.2.6 (Efficient Group Action)**.** *Let $(G, X, \star)$ be a group action with G a finite group and X a finite set. We say that $(G, X, \star)$ is an* ***effective group action (EGA)*** *if there exist efficient algorithms taking bit string as input solving the following problems:*

- ***Set (resp. Group) membership testing****: decide if the input is a valid representation of a set element (resp. a group element),*
- ***Equality testing****: decide if two inputs represent the same group element,*
- ***Sampling****: sample a random group element from a distribution indistinguishable from the uniform distribution over G,*
- ***Operating****: compute the product of two given group elements,*
- ***Inverting****: compute the inverse of a given group element,*

- ***Representing**: compute a* unique *representative of a given set element,*

- ***Acting**: given $g \in G, x \in X$, compute the action $g \star x$.*

In addition, the set X needs to have a public element x_0.

Remark 1.2.7. *Notice that the definition of an EGA might differ depending on the literature. For instance, we have chosen to define the* sampling *problem using the uniform distribution, while some authors define it for a parameter distribution, for instance [ADMP20].*

From an EGA, it is possible to derive several advanced cryptographic protocols such as public key encryption [ElG84], digital signature [Sch90], threshold or updatable encryption [DF90, BDGJ20] schemes and more. Unfortunately, it is not an easy task to construct an EGA framework relying on a quantum resistant VECTORISATION problem. A simpler task is to build up a *Restricted* Efficient Group Action. In this weaker setup, the *acting* problem is assumed to be easy only for a small set of generators of the group.

Definition 1.2.8 (Restricted Effective Group Action)**.** *Let $(G, X, \star)$ be a group action with G a finite group and X a finite set. Let $\overrightarrow{g} := \{g_1, \ldots, g_n\}$ be a generating set of G. We say that $((G, X, \star), \overrightarrow{g})$ is a **restricted efficient group action (REGA)** if there exist efficient algorithms taking bit string as input solving the following problems:*

- ***Set membership testing**: decide if the input is a valid representation of a set element,*

- ***Representing**: compute a* unique *representative of a given set element,*

- ***Acting with $\overrightarrow{g}$**: for any $i \in [\![1, n]\!]$, compute the action $g_i \star x$ or $g_i^{-1} \star x$.*

In addition, we require n to be polynomial in $\log(\#G)$ and the set X to have a public element x_0.

We say that a (R)EGA is suitable for post-quantum cryptography, or just post-quantum secure, if the associated VECTORISATION problem is hard even for quantum computers.

The REGA framework is significantly limited by the lack of an algorithm to uniformly sample elements from the group and to compute efficiently their action. While this suffices for Diffie–Hellman-like key-exchanges built from group actions, more advanced protocols require the use of sophisticated techniques often resulting in impractical schemes. In isogeny-based cryptography, much effort has been made to obtain a digital signature scheme from a REGA framework; see for instance [DG19, BKV19].

1.3 Elliptic curves and isogenies

We now introduce the basics of elliptic curves and their isogenies. For more details on these concepts, we refer the reader to the standard reference [Sil86].

Definition 1.3.1 (Elliptic curve)**.** *An **elliptic curve** E over a field $\mathbb{F}$ is a smooth projective curve of genus 1 defined over $\mathbb{F}$ with a distinguished point over $\mathbb{F}$. This point is either denoted by ∞_E or 0_E. Equivalently, an elliptic curve is an abelian variety of dimension 1; see [Mil86].*

We note $E/\mathbb{F}$ to indicate that the elliptic curve E is defined over the field $\mathbb{F}$.

Let $\mathbb{F}$ be a field of characteristic p, hence p is a prime or 0. When $p \notin \{2, 3\}$, an elliptic curve $E/\mathbb{F}$ can always be given using a **short Weierstrass equation** as

$$E : y^2 = x^3 + Ax + B, \text{ with } A, B \in \mathbb{F} \text{ such that } \Delta(E) := 4A^3 + 27B^2 \neq 0.$$

The quantity $\Delta(E)$ is the **discriminant** of the curve.

Note that this condition on p is not restrictive for cryptography as finite fields need to have large characteristic for security. In this work, we use different models for the curves depending on our goals. Some models are more compact or ensure uniqueness.

Let $\mathbb{F}'$ be a field extension of $\mathbb{F}$, the set of **$\mathbb{F}'$-rational points** of the elliptic curve E is defined as

$$E(\mathbb{F}') := \{(x, y) \in \mathbb{F}' \text{ such that } y^2 = x^3 + Ax + B\} \cup \{0_E\}.$$

Hence, $E(\bar{\mathbb{F}})$ denotes the set of all the points of the curve.

As with all abelian varieties, elliptic curves are both geometric and algebraic objects. Indeed, they can be endowed with an abelian group structure.

Proposition 1.3.2. *Let $E/\mathbb{F}$ be an elliptic curve and $\mathbb{F}'$ be an extension field of $\mathbb{F}$. The points of the curve E, respectively the $\mathbb{F}'$-rational points, form a commutative group with neutral element 0_E. This group law is denoted by $+$.*

Elliptic curves are central objects in modern cryptography thanks to their efficient group law and compact representation. In this thesis, we focus on their application in post-quantum cryptography, an area that primarily relies on isogenies between elliptic curves — hence, the name isogeny-based cryptography.

For the next definitions, we assume that E and E' are elliptic curves defined over $\mathbb{F}$.

Definition 1.3.3 (Isogeny)**.** *An **isogeny** $\varphi : E \to E'$ is a non-constant rational map sending 0_E to $0_{E'}$. In particular, it is a surjective map with a finite kernel denoted* $\ker \varphi$*. An isogeny is said to be **cyclic** if its kernel is a cyclic group.*

Note that the zero map, being constant, is not an isogeny according to Definition 1.3.3. Nevertheless, it is often added to the set of isogenies to serve as a neutral element.

There are several ways to represent isogenies, which vary in terms of compactness and evaluation complexity. The complexity of evaluating an isogeny depends on its field of definition and of its degree.

Definition 1.3.4 (Degree)**.** *An isogeny $\varphi : E \to E'$ defines an embedding function*

$$\begin{aligned}\varphi^* : \mathbb{F}(E') &\to \mathbb{F}(E)\\ f &\mapsto f \circ \varphi,\end{aligned}$$

*where $\mathbb{F}(E')$ (resp. $\mathbb{F}(E)$) is the function field of E (resp. E'). The **degree** of φ, denoted* $\deg(\varphi)$*, is defined as the degree of the finite field extension* $\mathbb{F}(E)/\varphi^*(\mathbb{F}(E'))$*. When* $\deg(\varphi) = n$*, we say that φ is an n-isogeny.*

We say that an isogeny $\varphi : E \to E'$ is an **endomorphism** if $E = E'$ and an **isomorphism** if $\deg(\varphi) = 1$. If both conditions are satisfied, this is an **automorphism**. We denote the set of isogenies from E to E' by $\mathrm{Hom}(E, E')$, the set of endomorphisms of E by $\mathrm{End}(E)$ and its automorphisms by $\mathrm{Aut}(E)$. We shall detail the structure of these three sets later.

There exists a natural embedding of the integers into the endomorphisms of a given elliptic curve.

Definition 1.3.5 (Scalar endomorphisms)**.** *For any positive integer $n \in \mathbb{Z}_{>0}$, we denote by $[n]$ the map*

$$\begin{aligned} [n] : E &\to E \\ P &\mapsto \underbrace{P + \cdots + P}_{n \text{ times}}. \end{aligned}$$

For any negative integer n, we set $[n] : E \to E, P \mapsto -[-n]P$. Moreover, $[0]$ denotes the the zero map. For any integer $n \in \mathbb{Z}$, the map $[n]$ is an endomorphism of degree n^2.

The map $[n]$ is called the ***multiplication-by-n map****. We refer to these endomorphisms as* ***scalar*** *or* ***trivial*** *endomorphisms.*

The kernels of scalar endomorphism forms subgroups that are central to the theory of elliptic curves. They are called torsion subgroups.

Definition 1.3.6 (Torsion subgroup)**.** *For any $n \in \mathbb{Z}_{>0}$, the* ***group of n-torsion points*** *$E[n]$ is the set*

$$E[n] := \ker[n] = \{P \in E \text{ such that } nP = 0_E\}.$$

If n is coprime with p or if $p = 0$, we have

$$E[n] \simeq \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}.$$

Dividing isogenies by scalar endomorphisms or by arbitrary isogenies, is an important ingredient of this work. Let us provide a formal definition of what it means to divide an isogeny, as well as a definition of `IsogenyDivision` algorithms. An efficient `IsogenyDivision` algorithm is presented in Section 2.2.

Definition 1.3.7 (Isogeny division)**.** *Let $\varphi : E_1 \to E_2$ and $\psi : E_1 \to E_3$ be two isogenies between elliptic curves defined over $\mathbb{F}$. We say that ψ* ***divides*** *φ if there exists an isogeny $\eta : E_3 \to E_2$ such that $\varphi = \eta \circ \psi$.*

Definition 1.3.8 (`IsogenyDivision` algorithm)**.** *An `IsogenyDivision` algorithm takes as input three elliptic curves E_1, E_2, E_3 and two isogenies $\varphi : E_1 \to E_2, \psi : E_1 \to E_3$ and returns an isogeny $\eta : E_3 \to E_2$ such that $\varphi = \eta \circ \psi$ if it exists otherwise, it returns `False`.*

When there exists an isogeny $\varphi : E \to E'$ defined over an extension field $\mathbb{F}' \supseteq \mathbb{F}$, we say that the elliptic curve E is **$\mathbb{F}'$-isogenous** to E'. In fact, being isogenous is an equivalence relation, since for any given isogeny between two curves, there exists an isogeny in the opposite direction.[1] Thus, we simply say that E and E' are **$\mathbb{F}'$-isogenous**.

Definition 1.3.9 (Dual isogeny)**.** *For any isogeny $\varphi : E \to E'$, there exists a unique isogeny from E' to E, denoted $\hat{\varphi}$, such that $\varphi \circ \hat{\varphi} = [\deg(\varphi)] \in \mathrm{End}(E')$ and $\hat{\varphi} \circ \varphi = [\deg(\varphi)] \in \mathrm{End}(E)$. The isogeny $\hat{\varphi} : E' \to E$ is called the* ***dual*** *isogeny of φ.*

Composing a given isogeny with its dual is the same as multiplication by its degree. For endomorphisms, it is also possible to sum them with their dual. In this case, we obtain the multiplication map by its *trace*.

Definition 1.3.10 (Trace and characteristic polynomial)**.** *Let α be an endomorphism of E. The trace of α is*

$$\mathrm{tr}(\alpha) = \alpha + \hat{\alpha} \in \mathbb{Z}.$$

[1] In addition, the relation of being isogenous is reflexive because any elliptic curve is isomorphic to itself and transitive since the composition of isogenies is an isogeny.

The characteristic polynomial of a non-scalar endomorphism α *is*

$$X^2 - \mathrm{tr}(\alpha)X + \deg(\alpha),$$

which has discriminant $\mathrm{tr}(\alpha)^2 - 4\deg(\alpha) < 0$.

We now introduce an important classification for isogenies.

Definition 1.3.11 (Separability)**.** *An isogeny* $\varphi : E \to E'$ *is said to be* ***separable, inseparable*** *or* ***purely inseparable*** *if the extension* $\mathbb{F}(E)/\varphi^*(\mathbb{F}(E'))$ *has the corresponding property. If the isogeny* φ *is separable, then the cardinality of its kernel is equal to its degree.*

Separable isogenies have the advantage of being almost uniquely determined by their kernel.

Proposition 1.3.12. *Let* G *be a finite subgroup of an elliptic curve* E*. There exists a* separable *isogeny* $\varphi : E \to E/G$ *such that* $\ker\varphi = G$*. The isogeny* φ *is unique up to post-composition by an isomorphism of* E/G*. This construction offers a one-by-one correspondence from finite subgroups of* E *to separable isogenies of* E*, up to post-composition by isomorphism.*

This property is often used in cryptography to represent an isogeny by its kernel. This motivates the consideration of elliptic curves up to isomorphism. To determine if two elliptic curves are isomorphic, we rely on the j-invariant.

Proposition 1.3.13 (j-invariant and isomorphisms)**.** *The* ***j-invariant*** *of an elliptic curve* E *of short Weierstrass form*

$$E : y^2 = x^3 + Ax + B,$$

is

$$j(E) := 1728\frac{4A^3}{4A^3 + 27B^2}.$$

Two elliptic curves E *and* E' *are isomorphic over* $\mathbb{F}$ *if and only if* $j(E) = j(E') \in \mathbb{F}$.

In addition, an isomorphism over $\mathbb{F}$ *between* $E : y^2 = x^3 + Ax + B$ *and* $E' : y^2 = x^3 + A'x + B'$ *is given by the map* $(x, y) \mapsto (\mu^{-2}x, \mu^{-3}y)$ *where* $\mu \in \mathbb{F}$ *such that* $A = \mu^4 A'$ *and* $B = \mu^6 A'$.

From now on, we restrict ourselves to elliptic curves defined over the finite field $\mathbb{F}_{p^k}$, where p is a prime and k a positive integer. In particular the field of definition has characteristic $p > 3$.

Let us define the Frobenius isogenies and endomorphisms which are important isogenies of supersingular elliptic curves.

Definition 1.3.14 (Frobenius isogenies)**.** *Let* $E/\mathbb{F}_{p^k}$ *be an elliptic curve. For any positive integer* n*, the* ***p^n-Frobenius isogeny*** *of* E *is*

$$\phi_{p^n}^E : E \to E^{(p^n)} : (x, y) \to (x^{p^n}, y^{p^n}).$$

For $n = k$*, the* p^k*-Frobenius isogeny* $\phi_{p^k}^E$ *becomes an endomorphism. It is called the* ***Frobenius endomorphism*** *and is denoted* π_E*. When there is no ambiguity, we write* $\phi_{p^n}^E$ *as* ϕ_{p^n} *and* π_E *as* π.

Proposition 1.3.15. *Every isogeny φ of elliptic curves defined over $\mathbb{F}_{p^k}$ can be written as*

$$\varphi = \varphi_{sep} \circ \pi_{p^n},$$

where φ_{sep} is a separable isogeny and $n \geq 0$ an integer. Moreover, we have

$$\deg(\varphi) = p^n \deg(\varphi_{sep}).$$

Elliptic curves defined over finite fields can be classified into two categories: ordinary and supersingular elliptic curves, according to the following definition.

Definition 1.3.16 (Ordinary and supersingular elliptic curves)**.** *An elliptic curve $E/\mathbb{F}_{p^k}$ is **supersingular** if and only if $E[p] \simeq \{0\}$. It is **ordinary** otherwise.*

We denote by SS_p the set of $\bar{\mathbb{F}}_p$-isomorphism class of supersingular elliptic curve defined over $\bar{\mathbb{F}}_p$. There are $\frac{p}{12} + \varepsilon$ classes of supersingular elliptic curves in SS_p, with $\varepsilon \in \{0, 1, 2\}$. Every class admits a representative defined over $\mathbb{F}_{p^2}$.

Supersingular elliptic curves differ from ordinary ones by numerous points. Here is a brief list of properties of supersingular elliptic curves that are particularly useful in cryptography:

1. Every supersingular elliptic curve defined over $\mathbb{F}_{p^k}$ is isomorphic to an elliptic curve defined over $\mathbb{F}_{p^2}$. Equivalently, its j-invariant lies in $\mathbb{F}_{p^2}$. As a result, isomorphism classes of supersingular elliptic curves can be represented in a very compact manner.

2. An elliptic curve $E/\mathbb{F}_p$ has $p+1$ $\mathbb{F}_p$-rational points if and only if it is a supersingular elliptic curve. Hence, supersingularity allows exact control on the order of $\mathbb{F}_p$-rational points.

3. Every pair of supersingular elliptic curves are isogenous. This property guarantees that computing an isogeny between two supersingular elliptic curves is always a well-defined problem.[2]

Let us add some details about the second point. The interest in controlling the number of rational points is that it ensures the existence of isogenies of prescribed degree defined over small extension fields, hence minimising the complexity of computing them. Indeed, since the converse of Lagrange's theorem holds for abelian groups, we have that for any integer N dividing the cardinality of $E(\mathbb{F}_q)$, there exists a subgroup $G \subseteq E(\mathbb{F}_q)$ of order N, for E an elliptic curve defined over $\mathbb{F}_{p^k} \subseteq \mathbb{F}_q$. Then by Proposition 1.3.12, there exists an isogeny $\varphi : E \to E/G$ of degree N. Moreover, by Vélu's formulas [Vé71], the isogeny φ will be defined over $\mathbb{F}_q$.

Computational aspects of isogenies.

Let us now state results on elliptic curves and isogenies that are more computational in nature. For these results, we provide precise references or proofs, as they are not usually part of the standard foundational material. Since our goal is to use isogenies to perform cryptography, we need to be able to represent them. However, it is not clear how this should be done. Should we use rational maps, kernels or something else? This concern is actually of little relevance; what truly matters is the ability to evaluate the isogeny efficiently. Hence, instead of fixing a specific representation, we define the properties that

[2] In the general case, Tate's theorem states that two elliptic curves are $\mathbb{F}$-isogenous if and only if they have the same number of $\mathbb{F}$-rational points. Hence, this last item is actually a corollary of the previous one.

an **efficient** isogeny representation must satisfy. We rely on the recent definition from [Wes24].

Definition 1.3.17 (Efficient representation, following [Wes24, Definition 1.3])**.** *Let $\mathcal{A}$ be a polynomial time algorithm. It is an* efficient isogeny evaluator *if for any $D \in \{0,1\}^*$ such that $\mathcal{A}(\texttt{validity}, D)$ outputs $\top$, there exists an isogeny $\varphi : E \to E'$ (defined over some finite field $\mathbb{F}_q$) such that:*

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

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

From now on, and for the remainder of this thesis, we assume that all isogenies are given in efficient representation.

For the rest of the section, let us also assume that every elliptic curves are defined over a finite field $\mathbb{F}_{p^k}$.

One of the earliest ways to efficiently represent isogenies is due to Vélu. Vélu's formulas ensure efficient representations of isogenies of smooth degree.

Proposition 1.3.18 (Vélu's formulas [Vé71])**.** *Given a subgroup $G \subset E(\mathbb{F}_q)$, one can compute a separable isogeny $\varphi : E \to E/G$ and evaluate it in time $\tilde{O}(\#G \cdot \log(q))$.*

In several algorithms presented in this work, torsion subgroups play an important role. For instance, they are used to represent isogenies in higher dimensions and to compute isogenies of prescribed smooth degrees. Therefore, computing a basis for torsion subgroups efficiently is an important problem in isogeny-based cryptography.

Proposition 1.3.19 (Computing torsion basis)**.** *Let ℓ be a prime number and n a positive integer. One can compute a basis of the torsion $E[\ell^n]$ in $\tilde{O}(\ell^{3n}(k \log p) + \ell^{2n}(k \log p)^2)$. Moreover $E[\ell^n]$ lives in an extension of degree $O(\ell^{2n})$.*

Proof. The complexity of computing torsion bases is analysed in different contexts in the literature. For instance, to compute only a basis for $E[\ell]$ one can refer to [BCNE+19, Lemma 6.9]. A proof for the statement provided here can be found in the second paragraph of [Rob23b, Algorithm 4]. □

A direct consequence of Proposition 1.3.19 and Proposition 1.3.18 is a polynomial algorithm to compute random walks in isogeny-graphs.

Corollary 1.3.20 (Isogeny path)**.** *Given a supersingular elliptic curve E and a B-powersmooth integer n, one can compute an isogeny $\varphi : E \to E'$ of degree n in time polynomial in B and in the length of the input. This isogeny follows the uniform distribution among the set of isogenies of degree n.*

Proof. First, one factorises $n = \prod_{i=1}^{r} \ell_i^{e_i}$ in time polynomial in B and $\log n$. Then, for each prime factor ℓ_i, we repeat e_i times the following steps:

1. compute a ℓ_i-torsion basis of the current elliptic curve,
2. choose uniformly at random a point P of order ℓ_i,

3. compute an isogeny φ of kernel P and degree ℓ_i, using Vélu's formulas,

4. consider the codomain of φ as the current curve.

The isogeny given by the composition of every computed isogeny has degree n and has been uniformly sampled from all possible isogeny of degree n. By Proposition 1.3.19 and Proposition 1.3.18, the whole process takes time polynomial in the length of the input and in B. □

Since the `SIDH` attacks, it is also possible to interpolate an isogeny from some image points to obtain an efficient representation. We discuss `SIDH` attacks and present an isogeny interpolation algorithm, denoted `IsogenyInterpolation`, in Chapter 2.

Definition 1.3.21 (`IsogenyInterpolation` algorithm)**.** *An `IsogenyInterpolation` algorithm takes as input two elliptic curves E and E', an integer N and two set of points $\{P_1, \ldots, P_n\} \subset E$ and $\{P'_1, \ldots, P'_n\} \subset E'$ such that an isogeny, if it exists, sending P_i to P'_i for all $i \in [\![1, n]\!]$ is unique. It returns an efficient representation of the isogeny $\varphi : E \to E'$ of degree N such that $\varphi(P_i) = P'_i$, for all $i \in [\![1, n]\!]$, if it exists, otherwise, it returns `False`.*

Remark 1.3.22. *It is well-known that isogenies of degree N are uniquely determined by the images of $4N + 1$ distinct points; this is an application of the Cauchy-Schwarz inequality as in the proof of Algorithm 1. Usually, to represent an isogeny of degree N by interpolation, we choose cyclic or torsion subgroups of order greater than $4N$.*

Lattices and endomorphisms.

We now turn to describe the structures of the sets of homomorphisms, endomorphisms and automorphisms for supersingular elliptic curves. The latter is, in fact, similar for ordinary and supersingular elliptic curves. The results on elliptic curves and their isogenies presented here are, one again, detailed in [Sil86].

Proposition 1.3.23 (Automorphism group)**.** *Let E be an elliptic curve defined over $\mathbb{F}_{p^k}$, with $p > 3$. The set of automorphisms of E forms a group denoted $\mathrm{Aut}(E)$ generated by*

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

where i is a primitive 2-nd root of unity and ζ_3 is a primitive 3-rd root of unity, both in $\bar{\mathbb{F}}_p$.

Lattices provide a key algebraic framework for describing these structures. We now provide important results on lattices for this work; for more details on lattices, we refer the reader to [Cas96].

Definition 1.3.24 (Quadratic spaces and lattices)**.** *We say that the pair (V, f) is a **quadratic space** of dimension d if V is a $\mathbb{Q}$-vector space of finite dimension d and $f : V \to \mathbb{Q}$ is a positive definite quadratic form. Its associated bilinear form is*

$$\langle x, y \rangle_f = \frac{1}{2}(f(x + y) - f(x) - f(y)).$$

*Then a **lattice** Λ in V is a subgroup $\Lambda \subset V$ of rank d such that $V = \mathbb{Q}\Lambda$.*

Remark 1.3.25. *Our definition of a lattice corresponds to what is commonly referred to in the literature as a full-rank lattice. Throughout this thesis, lattices always have rank equal to their dimension. Therefore, we use the terms "rank" and "dimension" interchangeably.*

Definition 1.3.26 (Gram matrix, discriminant and volume)**.** *Let Λ be a lattice of $\mathbb{Z}$-basis $(b_i)_{i=1}^d$ in a quadratic space (V, f) of dimension d. We define its **Gram matrix** to be*

$$G = (\langle b_i, b_j \rangle_f)_{i,j=1}^d.$$

*The **volume** of Λ is*

$$\operatorname{Vol}(\Lambda) := \sqrt{|\det(G)|}$$

*and its **discriminant** is*

$$\operatorname{disc}(\Lambda) = 16 \operatorname{Vol}(\Lambda)^2.$$

Definition 1.3.27 (Successive minima)**.** *Let Λ be a lattice in a quadratic space (V, f) of dimension d. For any $i \in [\![1, d]\!]$, the **i-th successive minimum** λ_i is the minimal rational number such that there exist i linearly independent vectors $x_1, ..., x_i$ in Λ satisfying $\sqrt{f(x_j)} \leq \lambda_i$ for $j \in [\![1, i]\!]$.*

Definition 1.3.28 (Minkowski's second theorem)**.** *Let Λ be a lattice in a quadratic space (V, f) of dimension d. The product of successive minima satisfies*

$$\prod_{i=1}^{d} \lambda_i \leq \gamma_d^{d/2} \operatorname{Vol}(\Lambda)$$

where γ_d is the Hermite constant.

Definition 1.3.29 (Minkowski-reduced basis)**.** *Let Λ be a lattice in a quadratic space (V, f) of dimension d. A basis $(b_1, \ldots, b_d)$ is **Minkowski-reduced** if, for all $i \in [\![1, d]\!]$, we have*

$$f(b_i) = \min\{f(x) \text{ such that } x \in \Lambda \setminus \operatorname{span}_{\mathbb{Z}}(b_1, \ldots, b_{i-1})\}.$$

We can now properly state the structure of isogeny sets via lattices.

Proposition 1.3.30 (Homomorphism module)**.** *Let E and E' be two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$. The set of isogenies from E to E' is a $\mathbb{Z}$-module of rank 4 called the homomorphism module from E to E'. We denote it $\operatorname{Hom}(E, E')$. It has a structure of lattice with associated bilinear form*

$$\langle \varphi, \psi \rangle = \frac{1}{2}(\hat{\varphi} \circ \psi + \hat{\psi} \circ \varphi).$$

The discriminant of this lattice is p^2.

Proposition 1.3.31 ([EHL+18, Lemma 4])**.** *Given $\varphi : E \to E'$ and $\psi : E \to E'$ two efficiently represented isogenies, one can compute $\langle \varphi, \psi \rangle$ in polynomial time in $\log p$ and in the length of the representation of φ and ψ.*

Remark 1.3.32. *From Proposition 1.3.31, one can compute the trace of an endomorphism α in polynomial time since we have*

$$\operatorname{tr}(\alpha) = \langle \alpha, 2 \rangle.$$

We recall that the degree is directly given by the efficient representation of the isogeny. Hence, the degrees and traces of endomorphisms are accessible when efficiently represented.

Note that the structure of the set $\mathrm{End}(E)$ is directly obtained by specialising Proposition 1.3.30 to the case $E = E'$. As a consequence, it is also a lattice of rank 4 and discriminant p^2. An advantage of lattices of rank at most 4 is that their Minkowski-reduced bases achieve all successive minima, see [vdW56]. Together with Minkowski's second theorem, this provides bounds on the degrees of isogenies generating bases of $\mathrm{Hom}(E, E')$ or $\mathrm{End}(E)$.

Proposition 1.3.33. *Let E and E' be two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$. Let $(\beta_i)_{i=1}^4$ be a Minkowski-reduced basis of $\mathrm{Hom}(E, E')$. Then, we have*

$$\deg(\beta_i) = O(p), \forall i \in [\![1, 4]\!].$$

Moreover $\deg(\beta_1) \leq \frac{2\sqrt{2p}}{\pi}$.

When $E' = E$, $(\beta_i)_{i=1}^4$ is a basis of $\mathrm{End}(E)$. In this case, we have $\deg(\beta_1) = 1$ and $\deg(\beta_2) \leq 2p^{2/3}$.

Proof. This is a well-known application of Minkowski's second theorem. See for instance [DLRW24, Lemma 12] for the result on $\mathrm{Hom}(E, E')$ and [EL24, Section 2.1] for $\mathrm{End}(E)$. Both references adopt the quaternion point of view given by the Deuring correspondence, we introduce it in the next subsection. Note that, in the $E' \neq E$ case, classically the degrees are $O(p^2)$. To obtain the $O(p)$, there is more work to do than apply the Minkowski's second theorem, see [DLRW24, Lemma 48]. □

The Deuring correspondence.

In the following subsection, we highlight the connection between endomorphism rings of supersingular elliptic curves and quaternion algebras provided by **the Deuring correspondence** [Deu41]. A standard reference for this topic is [Voi21]. This subsection is also based on Leroux's thesis [Ler22].

The starting point of this connection is the fact that endomorphism algebras of supersingular elliptic curves are quaternion algebras. Let us define these notions formally.

Definition 1.3.34 (Quaternion algebra)**.** *An algebra B over a field $\mathbb{F}$ of characteristic different from 2 is a **quaternion algebra** over $\mathbb{F}$ if there exists an $\mathbb{F}$-basis $1, i, j, ij$ for B satisfying*

$$i^2 = a, j^2 = b, ij = -ji,$$

for some $a, b \in \mathbb{F}^\times$. We denote by $(\frac{a,b}{\mathbb{Q}})$ the quaternion algebra generated as above.

For any quaternion algebra $B = (\frac{a,b}{\mathbb{Q}})$ with basis $(1, i, j, ij)$, there exists a **conjugation** involution defined as

$$\beta = x + yi + zj + tij \mapsto \bar{\beta} := x - yi - zj - tij.$$

We define the **reduced norm** of a quaternion β as $\mathrm{Nrd}(\beta) := \beta\bar{\beta}$ and its **reduced trace** as $\mathrm{Trd}(\beta) := \beta + \bar{\beta}$. The former is a quadratic form on B, with the associated bilinear form

$$\langle \beta_1, \beta_2 \rangle = \frac{1}{2}\,\mathrm{Trd}(\beta_1\bar{\beta}_2).$$

When B is ramified at infinity, the reduced norm is positive definite. Hence (B, Nrd) is a quadratic space of dimension 4.

Definition 1.3.35 (Ramification)**.** *For a prime p, we say that a quaternion algebra B is **ramified at** p if $B \otimes_{\mathbb{Q}} \mathbb{Q}_p \not\simeq M_2(\mathbb{Q}_p)$, where $\mathbb{Q}_p$ is the field of p-adic numbers. It is **ramified at infinity** ∞ if $B \otimes_{\mathbb{Q}} \mathbb{R} \not\simeq M_2(\mathbb{R})$. We denote by $B_{p,\infty}$ the quaternion algebra ramified only at p and ∞.*

The quaternion algebra $B_{p,\infty}$ is unique up to isomorphism.

Definition 1.3.36 (Endomorphisms algebra)**.** *Let $E/\mathbb{F}_{p^k}$ be a supersingular elliptic curve. The **endomorphism algebra** of E is $\operatorname{End}^0(E) := \operatorname{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$. It is a quaternion algebra ramified at p and ∞.*

We denote the elements of $\operatorname{End}^0(E)$ either by $\alpha \otimes \frac{r}{s}$ or $\frac{r}{s}\alpha$, for $\alpha \in \operatorname{End}(E)$ and $\frac{r}{s} \in \mathbb{Q}$.

For a supersingular elliptic curve $E/\mathbb{F}_{p^k}$, the conjugation involution on the endomorphism algebra $\operatorname{End}^0(E)$ can be obtained by extending the duality map $\operatorname{End}(E) \to \operatorname{End}(E), \alpha \mapsto \hat{\alpha}$ as follows

$$\widehat{r\alpha} = r\hat{\alpha}$$

for any $r \in \mathbb{Q}, \alpha \in \operatorname{End}(E)$. This involution is named the Rosati involution. Then the reduced norm and reduced trace in the endomorphism algebra are defined with respect to the Rosati involution. In particular, the reduced norm (resp. reduced trace) of an endomorphism is equal to its degree (resp. trace). Hence, the bilinear form associated with the reduced norm corresponds to the one presented in Proposition 1.3.30.

The endomorphism ring of an elliptic curve is a maximal order in the endomorphism algebra.

Definition 1.3.37 (Maximal orders)**.** *Let B be a quaternion algebra. A lattice Λ in B that is also a subring of B is called an **order**. It is a **maximal order** if, for every order Λ' in B with $\Lambda \subseteq \Lambda'$, we have $\Lambda = \Lambda'$.*

Maximal orders in quaternion algebra ramified in p and at infinity have volume $p/4$.

For any lattice $\Lambda \subset B$, we define its **left order** to be

$$\mathcal{O}_L(\Lambda) := \{\alpha \in B \text{ such that } \alpha\Lambda \subseteq \Lambda\},$$

and its **right order** to be

$$\mathcal{O}_R(\Lambda) := \{\alpha \in B \text{ such that } \Lambda\alpha \subseteq \Lambda\}.$$

When working with quaternion algebra isomorphic to $B_{p,\infty}$, the knowledge of maximal orders is sufficient to provide isomorphisms between the different quaternion algebras.

Proposition 1.3.38 ([CKMZ22, Proposition 4.1])**.** *Let A, B be quaternion algebras isomorphic to $B_{p,\infty}$. Given $\mathcal{O}_A$ a maximal order in A and $\mathcal{O}_B$ a maximal order in B, one can compute an isomorphism between A and B in polynomial time.*

Moreover, it is possible, under **GRH**, to efficiently compute a model for the quaternion algebra $B_{p,\infty}$, i.e. determining a basis, together with a maximal order in this quaternion algebra. We state this result following [Wes22b, Lemma 2.2 and 2.3], which is in turn based on [KLPT14, Section 2.3] and [EHL+18, Proposition 1].

Proposition 1.3.39. *Let $p > 2$ be a prime, then $B_{p,\infty} = (\frac{-p,-q_p}{\mathbb{Q}})$ with basis $(1, i, j, ij)$ and there is a maximal order $\mathcal{O}$ such that*

$$q_p = \begin{cases} 1 & \text{if } p \equiv 3 \mod 4, \\ 2 & \text{if } p \equiv 5 \mod 8, \end{cases} \quad \text{and } \mathcal{O}_0 = \begin{cases} \langle 1, i, \frac{i+ij}{2}, \frac{1+j}{2} \rangle & \text{if } p \equiv 3 \mod 4, \\ \langle 1, i, \frac{2-i+ij}{4}, \frac{-1+i+j}{2} \rangle & \text{if } p \equiv 5 \mod 8. \end{cases}$$

*When $p \equiv 1 \mod 8$, q_p is the smallest prime such that $q_p \equiv 3 \mod 4$ and $(\frac{p}{q_p}) = -1$ and the maximal order $\mathcal{O}_0$ is generated by $\langle \frac{1+i}{2}, \frac{j+ij}{2}, \frac{i+cij}{q_p}, ij \rangle$, where c is an integer such that q_p divides $c^2p + 1$. Under **GRH**, $q_p = O(\log(p)^2)$ and $\mathcal{O}_0$ contains the suborder $R + Rj$ with index $O(\log(p)^2)$, where R is the ring of integers of $Q(i)$.*

One can also compute in polynomial time a supersingular elliptic curve defined over a prescribed finite field together with an explicit isomorphism between its endomorphism ring and a maximal order in a quaternion algebra.

Proposition 1.3.40 ([EHL$^+$18, Proposition 3])**.** *There is an algorithm that, for any prime $p > 2$, computes a supersingular elliptic curve E_0 defined over $\mathbb{F}_p$ and an isomorphism $\varepsilon : \mathcal{O}_0 \to \mathrm{End}(E_0)$, where $\mathcal{O}_0$ is the maximal order described in Proposition 1.3.39, in time polynomial in $\log p$. When $p \equiv 1 \mod 8$, we assume **GRH**.*

Without **GRH**, we do not know, in general, a maximal order in $B_{p,\infty}$, or even a basis of the quaternion algebra. In particular, we cannot guarantee that a supersingular elliptic curve with a known endomorphism is accessible.

The Deuring correspondence extends to a relation between isogenies and ideals of maximal orders. Let us introduce these ideals and their main properties. We fix two maximal orders $\mathcal{O}_1$ and $\mathcal{O}_2$ in a quaternion algebra B.

Fractional ideals are lattice of rank 4 in B. The **reduced norm** of a fractional ideal I is

$$\mathrm{Nrd}(I) := \gcd(\mathrm{Nrd}(\alpha) \text{ such that } \alpha \in I).$$

By default, an ideal I is assumed to be **integral**, i.e. $I \subseteq \mathcal{O}_L(I)$ or equivalently $I \subseteq \mathcal{O}_R(I)$. Then its reduced norm is simply the square root of its norm, i.e. it is equal to $\sqrt{\#(\mathcal{O}_L(I)/I)}$. Its **normalised quadratic form** is

$$q_I : I \to \mathbb{Z}, \alpha \mapsto \frac{\mathrm{Nrd}(\alpha)}{\mathrm{Nrd}(I)}.$$

For two ideals I, J such that $\mathcal{O}_R(I) = \mathcal{O}_L(I)$, we define the product $I \cdot J$ as the products of pairs in $I \times J$. We say that an ideal I is **invertible** if there exists an ideal I^{-1} such that $II^{-1} = \mathrm{Nrd}(I)\mathcal{O}_L(I) = \mathrm{Nrd}(I)\mathcal{O}_R(I^{-1})$ and $I^{(-1)}I = \mathrm{Nrd}(I)\mathcal{O}_R(I) = \mathrm{Nrd}(I)\mathcal{O}_L(I^{-1})$. Most of the time, the ideals considered in this work are invertible.

Let I be a left $\mathcal{O}_1$-ideal. Then $\mathcal{O}_L(I) = \mathcal{O}_1$ and $\mathcal{O}_R(I)$ is another maximal order in B. This maximal order can be computed efficiently using [Ró92, Theorem 3.2]. We say that an ideal **connects** the maximal orders $\mathcal{O}_1$ and $\mathcal{O}_2$ if $\mathcal{O}_L(I) = \mathcal{O}_1$ and $\mathcal{O}_R(I) = \mathcal{O}_2$. Connecting ideals can be computed in polynomial time using [KV10, Algorithm 3.5]. We define **the connecting ideal** of $\mathcal{O}_1$ and $\mathcal{O}_2$ to be

$$I(\mathcal{O}_1, \mathcal{O}_2) = \{\alpha \in B \text{ such that } \alpha\mathcal{O}_2\bar{\alpha} \subseteq [\mathcal{O}_2 : \mathcal{O}_1 \cap \mathcal{O}_2]\mathcal{O}_1\},$$

where $[\mathcal{O}' : \mathcal{O}]$ denotes the index of a suborder $\mathcal{O}$ in an order $\mathcal{O}$. In particular, the ideal $I(\mathcal{O}_1, \mathcal{O}_2)$ connects $\mathcal{O}_1$ and $\mathcal{O}_2$.

Two left $\mathcal{O}$-ideals I and J are **equivalent**, denoted $I \sim J$, if there exists a quaternion $\alpha \in B^\times$ such that $I = \alpha J$. The map

$$\chi_I(\alpha) = I\frac{\bar{\alpha}}{\mathrm{Nrd}(I)}, \alpha \in I$$

provides exactly the equivalent ideals to I. In addition, the reduced norm of $\chi_I(\alpha)$ is $q_I(\alpha)$.

The connection between ideals and isogenies is done via the kernel ideals.

From ideals to isogenies:

Let I be a left $\mathrm{End}(E)$-ideal, for E a supersingular elliptic curve. We define the kernel of the ideal I, or the ***I*-torsion**, as

$$E[I] := \{P \in E \text{ such that } \alpha(P) = 0, \forall \alpha \in I\},$$

and we denote by $\varphi_I : E \to E/E[I]$ an isogeny of kernel $E[I]$. The isogeny φ_I is called the **I-multiplication** and the elliptic curve $E^I := E/E[I]$ the **I-transform**. When there is ambiguity on the domain of the isogeny φ_I, we write it $\varphi_{E,I}$.

From isogenies to ideals:

Let $\varphi : E \to E'$ be an isogeny between two supersingular elliptic curves. We define its kernel ideal to be

$$I_\varphi := \{\alpha \in \mathrm{End}(E) \text{ such that } \alpha(P) = 0, \forall P \in \ker(\varphi)\} \subset \mathrm{End}(E).$$

Equivalently, $I_\varphi = \mathrm{Hom}(E', E) \circ \varphi$. This ideal is a right $\mathrm{End}(E')$-ideal.

These two maps are inverse of each other. They provide the following bijection:

$$\begin{aligned} \{\varphi : E \to E'\}/\sim &\longrightarrow \{I \text{ left } \mathrm{End}(E)\text{-ideal}\} \\ \varphi &\longmapsto I_\varphi \\ \varphi_I &\longleftarrow\!\shortmid I, \end{aligned}$$

where $\varphi \sim \psi$ if $\psi = \lambda \circ \varphi$ for some isomorphism α of E'. Through this bijection, endomorphisms correspond to principal ideal. Moreover, the composition of isogenies translates to multiplication of ideals the other way around, i.e.

$$I_{\varphi \circ I_\psi} = I_\varphi \cdot I_\psi.$$

We use the same notation as above when working with a maximal order $\mathcal{O} \subset B$ isomorphic to $\mathrm{End}(E)$, instead of working directly with $\mathrm{End}(E)$. For instance, given a left $\mathcal{O}$-ideal I, we denote

$$E[I] := \{P \in E \text{ such that } \alpha(P) = 0, \forall \alpha \in \varepsilon(I)\},$$

where $\varepsilon : \mathcal{O} \xrightarrow{\sim} \mathrm{End}(E)$ is an isomorphism. It is important to keep in mind that knowing the isomorphism explicitly is required to be able to compute kernel ideals.

In several algorithms presented in this work, we need to compute isogenies corresponding to ideals, we call such a procedure an `IdealToIsogeny` algorithm. In Chapter 2, we introduce the efficient `IdealToIsogeny` algorithm provided by the `CLAPOTI` algorithm [PR23].

Definition 1.3.41 (`IdealToIsogeny` algorithm)**.** *An `IdealToIsogeny` algorithm takes a supersingular elliptic curve E and an $\mathcal{O}$-ideal I, for $\mathcal{O}$ a maximal order isomorphic to* $\mathrm{End}(E)$ *in a quaternion algebra $B \overset{\varepsilon}{\simeq} End^0(E)$, such that the isomorphism ε is explicit,· and returns the corresponding isogeny $\varphi_I : E \to E/E[I]$.*

Let us summarise every connection between quaternions and endomorphisms useful for this work in Figure 1.3.

Supersingular j-invariants over $\mathbb{F}_{p^2}$	Maximal orders in $B \simeq B_{p,\infty}$
$j(E)$ up to Galois conjugacy	$\mathcal{O} \simeq \mathrm{End}(E)$ up to isomorphism
$\varphi : E \to E_1$	I_φ left $\mathcal{O}$-ideal and right $\mathcal{O}_1$-ideal
$\theta \in \mathrm{End}(E)$	Principal ideal $\mathcal{O}\theta$
$\hat{\varphi}$	$\bar{I}$
$\deg(\varphi)$	$\mathrm{Nrd}(I_\varphi)$
$\varphi : E \to E_1$, $\psi : E \to E_1$	$I_\varphi \sim I_\psi$
$\psi \circ \varphi : E \to E_1 \to E_2$	$I_{\psi\circ\varphi} = I_\varphi \cdot I_\psi$

Figure 1.3: A summary of the Deuring correspondence. It is a simplified version of [Ler22, Table 2.1]

Note that, for a given isomorphism class of a maximal order $\mathcal{O}$, called the **type** of $\mathcal{O}$, there exists at most two elliptic curves, up to isomorphism, with their endomorphism ring isomorphic to $\mathcal{O}$.

1.4 Orientations of elliptic curves

It is well known that the ideal class group $\mathrm{Cl}(\mathfrak{O})$ of an imaginary quadratic order $\mathfrak{O}$ acts freely and transitively on the set of ordinary elliptic curves having an endomorphism ring isomorphic to $\mathfrak{O}$, see [Wat69, Theorem 4.5]. It is possible to obtain a similar action on supersingular elliptic curves by restricting the Deuring correspondence to an imaginary quadratic order. The map providing the embedding of an imaginary quadratic order to a quaternion algebra is called an orientation. This theory has been introduced in isogeny-based cryptography by [CK20]. In this section, we also rely on results from [Onu21].

We fix a prime p and a supersingular elliptic curve $E/\mathbb{F}_{p^2}$. Let K be an imaginary quadratic number field.

Definition 1.4.1 (Orientation)**.** *An embedding $\iota : K \hookrightarrow \mathrm{End}^0(E)$ is called a* ***K-orientation*** *on the elliptic curve E. If a K-orientation on E exists, we say that E is* ***K-orientable****. Then the pair (E, ι) is a* ***K-oriented*** *elliptic curve.*

For an order $\mathfrak{O}$ in K, we say that ι is an ***$\mathfrak{O}$-orientation****, E is* ***$\mathfrak{O}$-orientable****, and (E, ι) is an* ***$\mathfrak{O}$-oriented*** *elliptic curve if $\iota(\mathfrak{O}) \subseteq \mathrm{End}(E)$. By abuse of notation, ι is often viewed as its restriction to $\mathfrak{O} \hookrightarrow \mathrm{End}(E)$.*

When an orientation cannot be extended to a greater order, we say it is primitive. Morally, this property guarantees that the orientation provides the maximum information on the endomorphism ring it can.

Definition 1.4.2 (Primitive orientation)**.** *An $\mathfrak{O}$-orientation ι is* ***primitive*** *if*

$$\iota(\mathfrak{O}) = \mathrm{End}(E) \cap \iota(K).$$

Equivalently, for ω a generator of $\mathfrak{O}$, ι is primitive if there is no integer m such that $\iota(\omega/m)$ is an endomorphism of E. In this case, the oriented elliptic curve (E, ι) is said to be ***primitively $\mathfrak{O}$-oriented****.*

Let $\mathfrak{O}$ be a quadratic order in K and $\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$ be a primitive $\mathfrak{O}$-orientation on E.

Any isogeny $\varphi : E \to E'$ induces an orientation $\varphi_*(\iota)$ on E' from the knowledge of the orientation ι on E:

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

We see that the map $\varphi_*(\iota)$ is indeed an orientation as it maps elements of K to elements in $\mathrm{End}^0(E)$ and satisfies the ring homomorphism properties. In particular, the division by $\deg(\varphi)$ guarantees that $\varphi_*(\iota)(1) = 1$.

We define K-oriented isogenies between K-oriented elliptic curves to be isogeny compatible with the orientations, i.e. they map the orientation on the domain to the orientation of the codomain.

Definition 1.4.3 (Oriented isogeny)**.** *Let (E, ι) and (E', ι') be two K-oriented elliptic curves. An isogeny $\varphi : E \to E'$ is **K-oriented** from (E, ι) to (E', ι') if $\varphi_*(\iota) = \iota'$. When φ has degree 1, the isogeny φ is a **K-isomorphism**.*

For any order $\mathfrak{O}$ in K, we define $\mathrm{SS}_p(\mathfrak{O})$ to be the set of primitively $\mathfrak{O}$-oriented supersingular elliptic over $\bar{\mathbb{F}}_p$, up to K-oriented isomorphism. As it has been proven by [Onu21, Proposition 3.2], when p does not divide the conductor of $\mathfrak{O}$, this set is not empty.

From now on, we always assume that p does not divide the conductor of the considered quadratic order.

Proposition 1.4.4 ([Onu21])**.** *For any order $\mathfrak{O}$ in K, the tuple $(\mathrm{Cl}(\mathfrak{O}, \mathrm{SS}_p(\mathfrak{O}), \star)$ is a free group action with at most two orbits. The action is given by*

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

where $\varphi_{\mathfrak{a}} : E \to E^{\mathfrak{a}}$ is the $\mathfrak{a}$-multiplication map as defined in Section 1.3. In particular, its kernel is the $\mathfrak{a}$-torsion

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

Proof. This result combines Theorem 3.4 and Proposition 3.3 from [Onu21]. □

One can perform group actions with non primitive orientations. However, only primitive orientation ensures that the obtained action is regular, hence, we shall only consider this case for cryptographic purposes. See [ACL+23, Proposition 3.3] for the non-primitive case.

It is possible to move from one orbit to the other using the $\mathfrak{O}$-twist.

Proposition 1.4.5 (Twist)**.** *Let $(E, \iota) \in \mathrm{SS}_p(\mathfrak{O})$ and O be an orbit of the action $\mathrm{Cl}(\mathfrak{O})$ on $\mathrm{SS}_p(\mathfrak{O})$. Then either (E, ι) or its **$\mathfrak{O}$-twist** $(E, \bar{\iota})$ is in O. The orientation of the $\mathfrak{O}$-twist is defined as*

$$\bar{\iota}(\alpha) = \iota(\bar{\alpha}), \forall \alpha \in \mathfrak{O}.$$

Proof. This is shown in the proof of [Onu21, Proposition 3.3]. □

The cardinality of the group is crucial for the security of cryptography based on this action. To estimate it, we use the following proposition.

Proposition 1.4.6. *Let Δ be the discriminant of an imaginary quadratic order, then we have*

$$\# \mathrm{Cl}(\Delta) = O(\log(|\Delta|)\sqrt{|\Delta|}).$$

Proof. This estimate is discussed in [DDF+21, Section 5.3]. □

Let us isolate two properties of this group action which are interesting for the rest of this work.

Proposition 1.4.7. *Let $(E,\iota) \in \mathrm{SS}_p(\mathfrak{O})$ be an oriented elliptic curve and $\mathfrak{a}, \mathfrak{b}$ two $\mathfrak{O}$-ideals. Let (E',ι') be the codomain of the isogeny $\varphi_{E,\mathfrak{a}}$. Then the isogeny $\varphi_{E',\mathfrak{b}} \circ \varphi_{E,\mathfrak{a}}$ has kernel $E[\mathfrak{a}\mathfrak{b}]$.*

Proof. Direct application of [Wat69, Proposition 3.12]. □

Proposition 1.4.8. *Let $(E,\iota) \in \mathrm{SS}_p(\mathfrak{O})$ be an oriented elliptic curve and $\mathfrak{a}$ be a principal $\mathfrak{O}$-ideal generated by $\alpha \in \mathfrak{O}$. Then the isogeny $\varphi_{\mathfrak{a}}$ is equal to the endomorphism $\iota(\alpha)$ of E up to post-composition by an isomorphism and $\mathfrak{a}$ acts trivially on (E,ι).*

Proof. The kernel of $\varphi_{\mathfrak{a}}$ is by definition $\ker \iota(\alpha)$. By Proposition 1.3.12, there is an isomorphism ε of E such that the isogeny $\varphi_{\mathfrak{a}} = \varepsilon \circ \iota(\alpha)$.

Let us now determine the orientation on E induced by $\varphi_{\mathfrak{a}}$ by computing its evaluation on a generator ω of $\mathfrak{O}$. We have

$$\begin{aligned}(\varphi_{\mathfrak{a}})_*(\omega) &= (\varphi_{\mathfrak{a}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{a}})/\deg \varphi_{\mathfrak{a}} \\ &= (\varepsilon \circ \iota(\alpha) \circ \iota(\omega) \circ \hat{\iota}(\alpha) \circ \hat{\varepsilon})/\deg \varphi_{\mathfrak{a}} \\ &= \varepsilon \circ \iota(\alpha\omega\bar{\alpha}) \circ \hat{\varepsilon})/\deg \varphi_{\mathfrak{a}} \\ &= \varepsilon \circ \iota(\omega) \circ \hat{\varepsilon}.\end{aligned}$$

Hence the isogeny $\varphi_{\mathfrak{a}} : (E,\iota) \to (E^{\mathfrak{a}}, \varepsilon \circ \iota(\omega) \circ \hat{\varepsilon})$ is a K-isogeny, with $K = \mathfrak{O} \otimes \mathbb{Q}$. Since $\varepsilon : (E,\iota) \to (E^{\mathfrak{a}}, \varepsilon \circ \iota(\omega) \circ \hat{\varepsilon})$ is a K-isomorphism, we have that

$$\mathfrak{a} \star (E,\iota) = (E,\iota) \in \mathrm{SS}_p(\mathfrak{O}).$$

□

We shall see that the complexity of applying an action depends on the norm of the representative, thereby it is interesting to provide a bound on this norm.

Proposition 1.4.9. *Let $\mathfrak{O}$ be a quadratic imaginary order of discriminant Δ and $\mathfrak{a}$ be an invertible $\mathfrak{O}$-ideal. Then, there exists an $\mathfrak{O}$-ideal equivalent to $\mathfrak{a}$ such that $\mathrm{N}(\mathfrak{a}) = O(\sqrt{|\Delta|})$.*

Proof. Every integral $\mathfrak{O}$-ideal $\mathfrak{b}$ equivalent to $\mathfrak{a}$ is given by $\bar{\beta}\mathfrak{a}/\mathrm{N}(\mathfrak{a})$ for some $\beta \in \mathfrak{a}$. Furthermore, the reverse holds as well and the norm of $\mathfrak{b}$ is $\mathrm{N}(\beta)/\mathrm{N}(\mathfrak{a})$. This is a well-known fact, demonstrated in the proof of [PR23, Proposition 2.1] for instance. In addition, for any basis $[\alpha_1, \alpha_2]$ generating $\mathfrak{a}$ as a $\mathbb{Z}$-module, the map

$$\mathbb{Z}^2 \to \mathbb{Z}, (x,y) \mapsto \mathrm{N}(x\alpha_1 + y\alpha_2)/\mathrm{N}(\mathfrak{a}) \tag{1.1}$$

is a positive definite form of discriminant Δ. Then, by Minkowski's second theorem, there is an $\mathfrak{O}$-ideal equivalent to $\mathfrak{a}$ of norm $O(\sqrt{|\Delta|})$. □

We conclude this section by providing a definition of efficiently represented orientations together with a method to uniquely encode them. This unique encoding is crucial for the resolution of the Vectorisation problem for oriented elliptic curves presented in Section 4.2. Both for the classical one, where it allows to check efficiently if two oriented curves are isomorphic during a meet-in-the-middle approach, and for the quantum one, where the Kuperberg's algorithm requires unique encoding to be properly used.

Definition 1.4.10 (Efficient representation of orientation [HW25a, Definition 8])**.** *Let (E,ι) be an $\mathfrak{O}$-orientated elliptic curve. An **efficient representation** of ι is a pair (ω, D) where ω is a generator of $\mathfrak{O}$, and D is an efficient representation of $\iota(\omega) \in \mathrm{End}(E)$.*

Throughout this thesis, we assume that orientations are efficiently represented.

To uniquely encode K-isomorphism class in $\mathrm{SS}_p(\mathfrak{O})$ we define the function `enc` introduced in [Wes22a]. First, we need to fix:

- A canonical representative for $\overline{\mathbb{F}}_p$-isomorphism class of elliptic curves over $\overline{\mathbb{F}}_p$; for instance, the curve of equation $E : y^2 + xy = x^3 - (36x + 1)/(j(E) - 1728)$ for any $j(E) \notin \{0, 1728\}$, see [Sil86, page 52],[3]
- A generator ω of $\mathfrak{O}$, typically one with the smallest possible norm,
- A deterministic procedure that takes as input an elliptic curve E in canonical form and returns a point $P \in E$ of order greater than $4\,\mathrm{Nrd}(\omega)$.

Then, we define $\texttt{enc}(E', \iota)$ to be the triplet (E, P, Q) such that

- E is the canonical representative of E',
- P is the point of order greater than $4\,\mathrm{Nrd}(\omega)$ output by the deterministic procedure,
- Q is its image by the endomorphism $\iota(\omega)(P)$.

It was proven in [Wes22a] that this map provides a unique encoding of the K-isomorphism classes. In addition, it can be computed in polynomial time when ι is given in efficient representation; as defined in Definition 1.3.17. Thereby, this ensures that one can verify if two oriented curves are K-isomorphic efficiently.

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

The `CSIDH` key exchange protocol

We present in this chapter the `CSIDH` key exchange protocol as an important example of application of the orientation framework. This key-exchange is central for the one we introduce in Section 4.3.

The authors of [CLM$^+$18] adapted the `CRS` key exchange scheme relying on *ordinary* elliptic curves to *supersingular* elliptic curves to provide a practical key exchange scheme. They named the obtained scheme `CSIDH`, pronounced "sea-side", for **C**ommutative **S**upersingular **I**sogeny **D**iffie–**H**ellman. The early version of `CSIDH` outperforms the state-of-the-art implementation of `CRS` by a factor of 2000.[4] This propelled `CSIDH` to become the first *practical* post-quantum non-interactive key exchange scheme.

Let us provide some insight into the decisive advantages of supersingularity that allowed `CSIDH` to distance itself significantly from `CRS`.

Why does supersingularity make it practical? As detailed in the original paper [CLM$^+$18], translating the `CRS` framework to `CSIDH` offers multiple advantages. Let us summarise important ones here.

First, in both schemes we consider the action of the ideal class group $\mathrm{Cl}(\mathcal{O})$, for some quadratic order $\mathcal{O}$, over a set of elliptic curves. In the ordinary case, the elliptic curves are required to have their endomorphism ring isomorphic to $\mathcal{O}$, while in the supersingular case, we require the elliptic curve to be primitively $\mathcal{O}$-oriented. Then the core idea is to

[3] For j-invariant equal to 0 or 1728, we use another canonical representative, for instance the one given at [Sil86, page 52]. In this case, the point Q needs to be substituted by the set $\{(\sigma_*\iota)(\omega)(P) | \sigma \in \mathrm{Aut}(E)\}$.

[4] We are refering to the state-of-the-art implementation of `CRS` at the time of the publication of `CSIDH`. However, to the best of our knowledge, this is still the state-of-the-art version.

act with small $\mathcal{O}$-ideals of prime norm that generate the whole class group.

A strength of supersingularity is the control on the curve cardinality. A supersingular elliptic curve defined over $\mathbb{F}_p$ always has $p+1$ points. Then by choosing the prime p such that $p-1$ is divisible by many small primes, we ensure that the curve has many points of small order defined over $\mathbb{F}_p$. Thereby, it has many isogenies of small degree defined over $\mathbb{F}_p$, hence easy to compute thanks to Vélu's formulas. In fact, by construction, these isogenies correspond to the prime ideal action we are looking for to generate the whole group.

On the other hand, the discriminant of the characteristic polynomial of the Frobenius achieves its maximum, in absolute value, with supersingular elliptic curves. Asymptotically, this implies that the cardinality of the ideal class group is close to its maximum for a fixed p, optimising the security of the schemes.

For ordinary elliptic curves, it is difficult to satisfy both properties — having many small order points and a large class group order — at the same time.

Nevertheless, a notable advantage of ordinarity is the absence of orientation data, which is costly both in terms of time and memory for supersingular elliptic curves. Luckily, by adopting a `CSIDH`-like setup, where the orientation is always given by the Frobenius endomorphism, it is possible to exploit the advantages of supersingularity while avoiding the drawbacks due to the orientation.

Finally, by combining the previous observations a few more tricks, one can perform all the computations involved in `CSIDH` over $\mathbb{F}_p$ only.

The `CSIDH` settings We now turn to formally define the `CSIDH` framework. Let p be a prime of the form $4 \cdot \ell_1 \cdot \cdots \cdot \ell_r - 1$ where $\ell_1, \dots, \ell_{r-1}$ are successive odd primes and ℓ_r is the smallest next prime such that p is prime. The setup is as follows:

- **The group** is the ideal class group $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$,
- **The set** is the set of supersingular elliptic curves defined over $\mathbb{F}_p$ primitively oriented by $\mathbb{Z}[\sqrt{-p}]$, via the map $\sqrt{-p} \to \pi$, up to $\mathbb{F}_p$ isomorphism. We denote this set $\mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}])$,
- **The action** is $\star : \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}]) \times \mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}]) \to \mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}])$ defined as in Section 1.4,
- **The public element** is given by the elliptic curve $E_0 : y^2 = x^3 + x$ defined over $\mathbb{F}_p$,
- **The generating set of the group** is $\overrightarrow{g} := \{\mathfrak{l}_1, \dots, \mathfrak{l}_r\}$, where $\mathfrak{l}_i$ is the $\mathbb{Z}[\sqrt{-p}]$-ideal generated by $\langle \ell_i, \pi - 1 \rangle$ for any $i \in [\![1, r]\!]$.

We denote by `CSIDH`-512 the specific instantiation where $r = 74$ so $\log p \simeq 512$, which at the time was conjectured to provide NIST level 1 security [NIS16].

It is not obvious to see that the set $\{\mathfrak{l}_1, \dots, \mathfrak{l}_r\}$ generates the whole ideal class group. Even more so because the Bach's bound [Bac90], which guarantees, under **GRH**, that the first $12 \log^2(p)$ prime ideals generates everything, is not reached for the parameter r suggests by the authors. Thankfully, the class group tends to be cyclic. For instance, for `CSIDH`-512, the integer r is equal to 75, which is far from the Bach's bound, but $\mathfrak{l}_3$ by itself generates the entire group. The generic `CSIDH` construction relies on heuristic argument to ensure that almost the complete ideal class group is generated by the set $\overrightarrow{g}$.

To really be qualify as REGA, the CSIDH framework needs to admit polynomial algorithms to perform *set membership testing, representing* and acting with generators. Let us address them one by one.

Representing. First of all, since the orientation is always the same, representing a class in $\mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}])$ reduces to representing an elliptic curve up to $\mathbb{F}_p$-isomorphisms. There are multiple ways to *uniquely* represent isomorphism classes of elliptic curves.

For instance, one could use the j-invariant or coefficients of its curve model. Even when both sets of data are defined over $\mathbb{F}_p$, it might be most costly in terms of memory to store all the model coefficients. In our specific case, by [CLM+18, Proposition 8], an isomorphism class in $\mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}])$ can always be represented by the elliptic curve

$$E_A : y^2 = x^3 + Ax + x,$$

for an $A \in \mathbb{F}_p$ which is uniquely determined. Thus both representations cost $\log p$ bits in memory. In CSIDH, a class $[E]$ in $\mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}])$ is represented by the integer A in $\mathbb{F}_p$ such that $E \simeq E_A$.

Set membership testing. The most straightforward way to test for membership is to rely once more on [CLM+18, Proposition 8]. Given an integer A, the proposition implies that one just has to check if the corresponding elliptic curve E_A is supersingular to verify that it is an element of $\mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}])$. This can be done by checking that the number of points on the curve is $(p+1)$ in polynomial time in $\log p$ with the SEA algorithm [Sch95].

Acting. It remains to prove that the action of the different $\mathfrak{l}_i$ can be computed efficiently. We recall that the action by $\mathfrak{l}$ sends a supersingular elliptic curve E in $\mathrm{SS}_{\mathbb{F}_p}(\mathbb{Z}[\sqrt{-p}])$ to the curve $E/E[\mathfrak{l}]$ where

$$E[\mathfrak{l}] := \cap_{\alpha \in \mathfrak{l}} \ker \alpha.$$

Then for $\mathfrak{l} = \langle \ell, \pi - 1 \rangle$, we have

$$E[\mathfrak{l}] = E[\ell] \cap \ker(\pi - 1).$$

Since, for supersingular elliptic curve defined over $\mathbb{F}_p$, the Frobenius endomorphism is the identity exactly for points defined over $\mathbb{F}_p$. We have that

$$E[\mathfrak{l}] = E[\ell] \cap E(\mathbb{F}_p).$$

This implies that any point P of order ℓ defined over $\mathbb{F}_p$ generates the kernel $E[\mathfrak{l}]$. Since for any $i \in [\![1, r]\!]$, the prime ℓ_i divides the cardinality of $E(\mathbb{F}_p)$, such a point P exists. Vélu-like formulas then provide an efficient way to compute the action.

For CSIDH to be a REGA also requires that the action $[\mathfrak{l}_i]^{-1} = [\bar{\mathfrak{l}}]$ is efficiently computable. Similar computations as above show that, the kernel of this action is generated by any point of order ℓ defined over $\mathbb{F}_{p^2} \setminus \mathbb{F}_p$. In [CLM+18], the authors present tricks to determine the elliptic curve $E/E[\mathfrak{l}]$ performing only computations over the base field $\mathbb{F}_p$.

Sampling. The above discussion addresses all aspects necessary for CSIDH to be a REGA. Now to perform a Diffie–Hellman key exchange, a main issue still remains. How should we sample ideals to ensure that their action sample a random enough element?

Even if the set $\overrightarrow{g}$ generates the ideal class group, without knowing its structure one can not sample ideal classes uniformly at random. Currently, the best algorithm to compute the structure of the ideal class group $\mathrm{Cl}(\Delta)$, for a discriminant $\Delta < 0$, is subexponential in $\log \Delta$ [HM89]. Hence, it seems unfeasible to compute it for the prime p we are considering.

The authors of `CSIDH` suggest to add a parameter integer m — for instance for `CSIDH`-512, we take $m = 5$ — and to restrict ourselves to the map

$$\begin{aligned}
&[\![-m, m]\!]^r \times \mathbb{F}_p \to \mathbb{F}_p \\
&((e_1, \dots, e_r), A) \mapsto A' \text{ such that } E_{A'} \simeq (\prod_{i=1}^{r} \mathfrak{l}_i^{e_i}) \star E_A.
\end{aligned}$$

Heuristically, we expect this map to have very few collisions when $(2m+1)^r \geq \# \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$.[5] Hence it should provide results that are sufficiently well-distributed to avoid brute-force search attacks.

In addition, restring the computation of actions to small exponents is also interesting in terms of efficiency. Indeed, there is no known analog to square-and-multiply (or double-and-add) algorithm for this context. Even if optimisations exist to compute the action of $\mathfrak{l}^e$ more efficiently than performing e successive actions of $\mathfrak{l}$, it remains costly.

The security of the scheme. As any key exchange based on group actions, a meet-in-the-middle key search can be performed in $O(\sqrt{N})$ where N is the size of the group. In `CSIDH`, N is the cardinality of the ideal class group $\mathrm{Cl}(\mathbb{Z}[\sqrt{-p}])$. By Proposition 1.4.6, we have that

$$\# \mathrm{Cl}(\mathbb{Z}[\sqrt{-p}]) = \tilde{O}(\sqrt{p}).$$

Hence, a meet-in-the-middle approach can break the security of `CSIDH` in time $\tilde{O}(\sqrt[4]{p})$. In Section 4.2, we prove Theorem 4.2.10 which rigorously demonstrates this result for oriented elliptic curves in general.

In Section 4.2, we also provide a rigorous asymptotic complexity analysis of the oriented Vectorisation problem regarding quantum computers. It implies that `CSIDH` can be broken in time subexponential in $\log p$ by quantum algorithms. The underlying quantum algorithm is the so-called Kuperberg's algorithm [Kup05]. Nevertheless, there is still ongoing discussion on the concrete complexity of this quantum algorithm, hence on the quantum security of `CSIDH`. While the authors of `CSIDH` suggest to use a prime p of length 512 to achieve NIST-I security in [CLM$^+$18], more recent analyses tend to recommend primes up to 4096 bits, see [CSCJR22].

The `CSURF` variant For a supersingular elliptic curve defined over $\mathbb{F}_p$ with $p \equiv 3 \mod 4$, the Frobenius endomorphism π is always a non trivial endomorphism. However, not all such curves are primitively oriented by

$$\mathbb{Z}[\sqrt{-p}] \hookrightarrow \mathrm{End}(E), \sqrt{-p} \mapsto \pi.$$

For some supersingular elliptic curves $E/\mathbb{F}_p$, we have that $\frac{1+\pi}{2}$ is a well-defined endomorphism. Since the ring of integers of $\mathbb{Q}(\sqrt{-p})$ is $\mathbb{Z}[\frac{1+\sqrt{-p}}{2}]$, these curves are primitively oriented by

$$\mathbb{Z}[\frac{1+\sqrt{-p}}{2}] \hookrightarrow \mathrm{End}(E), \sqrt{-p} \mapsto \pi.$$

We say that they are on the **surface**, while the curves primitively oriented by $\mathbb{Z}[\sqrt{-p}]$ are on the **floor**.

[5] Here, we describe the original `CSIDH` scheme. Since then several ways to choose the exponents have been developed; taking in account the relative costs of the different isogenies in function of their degree for instance.

In the CSURF variant of CSIDH, introduced in [CD20], the curves are on the surface, instead of being on the floor as in CSIDH. In addition, the prime p is assume to be congruent to 7 modulo 8. On the other hand, it is still constructed such that $p+1$ have a lot of small prime factors, as for CSIDH. The main advantage of this new framework is the possible to act with ideals of norm 2. This leads to a speed-up of around 5% for the CSURF-512 variant compared to the CSIDH-512.

An important point is that curves on the floor are not uniquely represented anymore by models like

$$E_A : y^2 = x^3 + Ax + x,$$

for some $A \in \mathbb{F}_p$. Fortunately, by [CD20, Proposition 4], when $p \equiv 7 \mod 8$ a supersingular elliptic curve $E/\mathbb{F}_p$ on the surface can always be given in the model

$$E : y^2 = x^3 + Ax - x,$$

where $A \in \mathbb{F}_p$ is unique.

Hence, the acting in CSURF— expect for ideals of norm 2 which are treated differently — is the same as in CSIDH representing and testing the membership is also analogous, the canonical model is just a different choice.[6] Nevertheless, in CSURF, the discriminant of the class group is as large as it can be for a fixed prime optimizing the asymptotical security.

Note that given a supersingular elliptic curve defined over $\mathbb{F}_p$ on can

1.5 Random walks over elliptic curves

Thought out this work, we shall perform random walks in two kind of graphs: in Cayley graphs given by oriented elliptic curves and in isogeny graphs.

Random walks in Cayley Graph. From any set of generators of a given group, one can define a graph, called a Cayley graph.

Definition 1.5.1 (Cayley graph)**.** *Let G be a finite group and let $S \subseteq G$ be a generating subset of G. The **Cayley graph** $\mathbf{Cay}(\boldsymbol{G}, \boldsymbol{S})$ 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 $g_2 = sg_1$.*

It has been proven by Bach [Bac90] under **GRH** that the ideal class groups $\mathrm{Cl}(\Delta)$ of a number field of discriminant Δ are generated by prime ideals of norm smaller than $12\log(\Delta)^2$. Hence, we can consider the Cayley Graph $\mathrm{Cay}(\mathrm{Cl}(\Delta), S)$ with S a set of small prime generating the whole group.

Childs, Jao and Soukharev have proven, under **GRH**, that the output of short random walks in this kind of graphs follows distribution close to the uniform distribution.

Proposition 1.5.2 ((**GRH**) Theorem 2.1 in [CJS14])**.** *Let $\mathfrak{O}$ be an imaginary quadratic order of discriminant Δ and conductor $f_{\mathfrak{O}}$. Let $\varepsilon > 0$ and x be a real number such that $x \geq (\log|\Delta|)^{2+\varepsilon}$. Let Σ_x be the set*

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

[6] One can also check if an elliptic curve defined over $\mathbb{F}_p$ belongs to the CSURF framework by verifying whether it is supersingular and situated on the surface. The latter can be done efficiently by looking at the 2-torsion; see [CD20, Remark 1].

from which we define the set S_x to be

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

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

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

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

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

Proposition 1.5.2 tells us that the ideal class given by a products of small primes is almost uniform among the class group. Hence, by acting with such ideals on an oriented elliptic curve, we almost obtain a uniform distribution over the oriented elliptic curves. We recall that this action is regular when restricted to a single orbit. In Chapter 4, this result allows us to use meet-in-the-middle approach in the context of oriented elliptic curves.

Random walks in isogeny graphs. Let p be a cryptographically large prime and $\ell \neq p$ a prime. The ℓ-isogeny graph over the finite field $\mathbb{F}_{p^2}$ is the graph having for vertices the set SS_p of supersingular elliptic curves up to isomorphism, where each class is represented by a curve defined over $\mathbb{F}_{p^2}$, and edges corresponding to ℓ-isogenies. This graph is $\ell + 1$-regular and connected. Hence it has around $p/12$ vertices.

The ℓ-isogeny graphs are optimal expander graphs, or Ramanujan graphs, as proven by [Piz90]. This implies that random walks in such graphs rapidly mix. In this subsection, we provide a rigorous proposition on this rapid mixing property for a more general case where random steps are taken in ℓ-isogeny graphs, for multiple different prime ℓ.

This generalisation is useful to perform random walks of powersmooth degree which can be then translated efficiently to its corresponding ideal thought Deuring correspondence. This trick and the more standard random walk in ℓ-isogeny graph are used for the reduction presented in Chapter 3.

The remainder of this content comes from the preliminaries of [HW25b].

Let $\mathbb{C}^{\mathrm{SS}_p}$ be the set of functions $\mathrm{SS}_p \to \mathbb{C}$. We consider two natural distances on $\mathbb{C}^{\mathrm{SS}_p}$. First,

$$d_{\mathrm{TV}}(f, g) = \frac{1}{2}\|f - g\|_1 = \frac{1}{2} \sum_{E \in \mathrm{SS}_p} |f(E) - g(E)|.$$

When f and g are distributions on SS_p, this is known as the *total variation distance.* Second, we have the scalar product

$$\langle f, g \rangle = \sum_{E \in \mathrm{SS}_p} f(E)\overline{g(E)} \# \mathrm{Aut}(E),$$

inducing the norm $\|f\| = \langle f, f \rangle^{1/2}$. Note that by the Cauchy-Schwarz inequality and Eichler's formula, for any f, g, we have

$$d_{\mathrm{TV}}(f, g) \leq \frac{\|f - g\|}{2} \left(\sum_{E \in \mathrm{SS}_p} \frac{1}{\# \mathrm{Aut}(E)} \right)^{1/2} = \frac{\|f - g\|}{2} \left(\frac{p-1}{24} \right)^{1/2}.$$

Given an elliptic curve E, a prime ℓ and an integer k, a random ℓ-walk from E of length k is a random sequence $(\varphi_0, \ldots, \varphi_{k-1})$ sampled as follows:

1. Let $E_0 = E$,
2. For each i, let G_i be a uniformly random subgroup of order ℓ in E_i, and let $E_{i+1} = E_i/G_i$
3. For each i, let $\varphi_i : E_i \to E_{i+1}$ be the quotient isogeny.

The curve E is the *source* of the walk, and the codomain of φ_{k-1} is called the *target* of the walk. Let N be a positive integer with prime factorization $N = \prod_{i=1}^{t} \ell_i^{k_i}$. A random N-walk from E is a sequence $(w_i)_{i=1}^{t}$ where

1. The source of w_1 is E,
2. Each w_i is a random ℓ_i-walk of length k_i, and
3. For each i, the target of w_i is the source of w_{i+1}.

The *target* of the walk is the target of w_t. Note that the walk itself depends on an order of the prime factors of N, but the distribution of the target does not.

The following proposition states that random walks rapidly converge to the so-called *stationary distribution.*

Definition 1.5.3. *The* stationary distribution *on* SS_p *is the probability distribution defined by* $\mu(E) = \frac{24}{(p-1)\#\operatorname{Aut}(E)}$.

Remark 1.5.4. *For any* $p > 3$*, the quantity* $\#\operatorname{Aut}(E)$ *is equal to* 2 *for all curves* E *with two exceptions: if* $j(E) = 1728$*, then* $\#\operatorname{Aut}(E) = 4$*, and if* $j(E) = 0$*, then* $\#\operatorname{Aut}(E) = 6$*. Therefore, the total variation distance between the uniform distribution and the stationary distribution on* SS_p *is* $O(1/p)$*. In particular, the two distributions are statistically and computationally indistinguishable.*

Proposition 1.5.5. *Let* N *be a positive integer with prime factorization* $N = \prod_{i=1}^{t} \ell_i^{k_i}$*. Let* E *be a random supersingular elliptic curve defined over* $\mathbb{F}_{p^2}$*, for some distribution* f*. Let* $W_N(f)$ *be the distribution of the target of a random* N*-walk from* E*. Then,*

$$\|W_N(f) - \mu\| \le \|f - \mu\| \cdot \prod_{i=1}^{t} \left(\frac{2\sqrt{\ell_i}}{\ell_i + 1}\right)^{k_i},$$

where μ *is the stationary distribution.*

Proof. This is a folklore consequence of Pizer's proof that the supersingular ℓ-isogeny graph is Ramanujan [Piz80]. However, previous literature only details the case where N is a prime power, so let us show that it extends to the general case. Following [PW24, Appendix A.1], let $W_\ell = B_\ell/(\ell + 1)$ be the ℓ-walk operator in X: for any distribution f on SS_p, we have that $W_\ell^k(f)$ is the distribution of the target of a random ℓ-walk of length k. From [PW24, Appendix A.1] and [PW24, Theorem 3.10], we have $\|W_\ell^k(f) - \mu\| \le \frac{2\sqrt{\ell}}{\ell+1}\|f - \mu\|$. We deduce that

$$\|W_N(f) - \mu\| = \|(W_{\ell_1}^{k_1} \circ \cdots \circ W_{\ell_t}^{k_t})(f) - \mu\| \le \|f - \mu\| \cdot \prod_{i=1}^{t} \left(\frac{2\sqrt{\ell_i}}{\ell_i + 1}\right)^{k_i},$$

as claimed. □

In our applications, we only need the following corollary, where we introduce a useful notation $\tau(p, \varepsilon)$ for the proofs in Section 3.4.

Corollary 1.5.6. *Let E be a random supersingular elliptic curve defined over $\mathbb{F}_{p^2}$, for some distribution f, and let $\varepsilon > 0$. There exists a bound $\tau(p, \varepsilon) = O(\log(p) - \log(\varepsilon))$ such that for any $N > 2^{\tau(p,\varepsilon)}$, the output distribution of a random N-walk is at total variation distance at most ε to the stationary distribution.*

Proof. By [BCC$^+$23, Theorem 7, Item 5], we have $\|f - \mu\| \leq \sqrt{3}$. Let $\lambda = \log(3/(2\sqrt{2}))$. From Proposition 1.5.5, we have

$$\|W_N(f) - \mu\| \leq \|f - \mu\| \cdot \prod_{i=1}^{t} \left(\frac{2\sqrt{\ell_i}}{\ell_i + 1}\right)^{k_i} \leq \sqrt{3} \cdot \left(\frac{2\sqrt{2}}{3}\right)^{\log(N)} = \sqrt{3} \cdot 2^{-\lambda \log(N)}.$$

Now,

$$d_{\mathrm{TV}}(W_N(f), \mu) \leq \frac{\|W_N(f) - g\|}{2} \left(\frac{p-1}{24}\right)^{1/2} \leq 2^{-\lambda \log(N)} \left(\frac{p-1}{24}\right)^{1/2}.$$

The latter quantity is smaller than ε if and only if

$$\log(N) \geq \frac{\lambda}{2} \left(\log(p-1) - \log(24) - 2\log(\varepsilon)\right) = O(\log(p) - \log(\varepsilon)),$$

which proves the result. □

1.6 Principally polarised abelian varieties

We now provide the necessary background for Chapter 2 and Section 4.3 where higher dimensional isogenies play a key role. By higher dimensional isogenies, we mean isogenies between principally polarised abelian varieties — a natural generalisation of elliptic curves to higher dimensions, see e.g. [Mil86]. These objects have become central in isogeny-based cryptography following the attacks on SIDH [CD23, MMP$^+$23, Rob23a], also discussed in Chapter 2.

This section is divided into two parts. The first part offers a brief high-level overview of abelian varieties and their isogenies, and fixes the notation. In fact, only products of elliptic curves appear directly in this thesis; technical aspects involving more general abelian varieties are abstracted away via results from the literature. We therefore avoid developing the full theory. Standard references on abelian varieties include [Mil86, Mum70], while the basics of algebraic geometry are introduced in [Har13]. The second part is more substantial and focuses specifically on products of elliptic curves; this material comes from [HW25a, Section 4].

Abelian varieties are smooth, projective, irreducible varieties equipped with a compatible abelian group structure given by regular maps. Elliptic curves correspond precisely to the abelian varieties of dimension one.

Every abelian variety A has a **dual** abelian variety $\hat{A}$. An homomorphism $\varphi : A \to B$ between abelian varieties is called an **isogeny** if its kernel is a finite subgroup of A and the dimension of A and B are the same; in particular, it is surjective. The **degree** of an isogeny between abelian varieties is defined from degrees of field extensions, similarly to Definition 1.3.4. Every isogeny $\varphi : A \to B$, has a **dual** isogeny $\hat{\varphi} : \hat{B} \to \hat{A}$ of the same degree. A **polarisation** $\lambda : A \to \hat{A}$ on an abelian variety A is a specific type of isogeny, derived from an ample divisor on A. When the polarisation λ is an isomorphism, it is called a **principal polarisation**, and the pair (A, λ) a **(principally) polarised abelian variety**, shortened as **(P)PAV**.

Once again, while the underlying theory is rich, we have no need for its full generality in this work. Indeed, for elliptic curves, there exists a *canonical* principal polarisation, denoted λ_E for an elliptic curve E. From this, we define the **product polarisation** $\lambda_{E_1\times\cdots\times E_n}$ on products of elliptic curves $E_1 \times \cdots \times E_n$ simply as the product of their individual polarisations. This natural and canonical construction is sufficient for our purposes.

In general, abelian varieties do not have explicit group laws or equations describing them. To enable computation, we embed them into a projective space using principal polisation and theta functions, see [Rob21] for more details on these aspects. Once equipped with a principal polarisation, abelian varieties and their isogenies resemble elliptic curves in many ways, as the following illustrates.

Products of elliptic curves. The work presented in this thesis mostly makes use of isogenies between products of elliptic curves. Let us introduce notation and important preliminary results.

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

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

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

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

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

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

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

and projection maps

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

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

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

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

Remark 1.6.1. *We shall primarily consider isogenies in higher dimensions whose domain is a product of elliptic curves that are principally polarised by the product polarisation. However, to compute such isogenies, one needs to use a coordinate system capable of* *handling any principally polarised abelian varieties of the same dimension. Currently, the most commonly used coordinate system is the theta model. In particular, this is the case in the generalisation of Vélus' formulas we shall use to compute higher dimensional isogenies. With such formulas, one can compute an N-isogeny between PPAVs of dimension g from its kernel in $\tilde{O}(N^g)$ operations over the field of definition [LR23]. Hence with $g = 1$ we recover the complexity of standard Vélu's formulas.*

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

A powerful feature of isogenies in higher dimensions is their ability to embed isogenies between elliptic curves. For convenience, we generalise the notion of efficient representation of isogenies between elliptic curves to isogenies between products of elliptic curves. Mainly, we require an efficient representation of isogenies between elliptic curves to be associated to an algorithm such that one can compute the image of any tuple of points in time polynomial in the length of the representation and in the size of the field over which those points are defined.

Definition 1.6.2 (Embedding representation)**.** *Let n be an integer and E_i, E'_i be elliptic curves for $i \in [\![1, n]\!]$. Let $\varphi : E \to E'$ be an isogeny such that $E \in \{E_1, \ldots, E_n\}$ and $E' \in \{E'_1, \ldots, E'_n\}$. An* ***embedding representation*** *of φ in dimension n is a triplet (F, i, j) associated to a representation of F, where $F : E_1 \times \cdots \times E_n \to E'_1 \times \cdots \times E'_n$, $i, j \in [\![1, n]\!]$ and such that $\varphi(P) = \pi_j \circ F \circ \tau_i$ for any $P \in E$.*

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

Definition 1.6.3 (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 $\tilde{\varphi} \circ \varphi = [N]$. We say that an N-isogeny between PPAV has* ***degree*** *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 matrix $\tilde{M}$.

The notions of **isogeny evaluators** and **representations of an isogeny**, Definition 1.3.17, naturally extend to N-isogenies between PPAV. In particular each N-isogeny is associated to principal polarisations on its domain and codomain. Thus algorithms of evaluation of N-isogenies also return the principal polarisation of the codomains.

Separable isogenies between elliptic curves are determined by their kernel (up to isomorphism of the target curve), see Proposition 1.3.12. Given a kernel, the corresponding isogeny can be evaluated using Vélu's formulas, see Proposition 1.3.18. We have similar results for N-isogenies with N prime to the characteristic of the field of definition. Indeed, such isogenies are determined by their kernel and there exists an analogue to Vélu's formulas for them.

Their kernel must be a **maximal isotropic** subgroup in order for the corresponding isogeny to behave well with the polarisations. Namely, it is essential for the isogeny to be a **polarised isogeny**. Before defining this maximality property, we need to introduce the Weil pairing for principally polarised abelian varieties.

Definition 1.6.4 (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 1.6.5 (Maximal isotropic subgroup)**.** *With the same notations as in Definition 1.6.4. Let H be a proper subgroup of $A[N]$. The subgroup H is* ***maximal isotropic in $\boldsymbol{A[N]}$*** *if the polarised Weil pairing $e_{N,\lambda}$ restricted to H is trivial but is not over any proper supergroup of H. For an isogeny with domain A having a maximal isotropic kernel in $A[N]$ is equivalent to be an N-isogeny.*

Lemma 1.6.6 (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****.*

Efficiently representing isogenies between products of elliptic curves defined by their kernel is a crucial step for the higher dimension tools presented in Chapter 2 — and thus for all of their applications, such as the effective group action presented in Section 4.3. The following lemma states the complexity to compute such a representation. This can be seen as a generalisation of Vélu's formulas for products of elliptic curves. It is worth noting that, to achieve the complexity claimed in the lemma, one needs to use the more general Vélu's formulas between generic abelian varieties mentioned in Remark 1.6.1; such generalisations have been introduced by Lubicz and Robert in [LR12].

Lemma 1.6.7 ([Rob22a])**.** *Let $(E_1 \times \cdots \times E_n, \lambda_{E_1 \times \cdots \times E_n})$ be a principally polarised abelian varity where E_i are elliptic curves defined over $\mathbb{F}_{p^k}$. Let H be a maximal isotropic subgroup of $(E_1 \times \cdots \times E_n)[N']$ where N' is an integer coprime to p of prime factorisation $\prod_{i=1}^r \ell_i^{e_i}$.*

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

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

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

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

Note that Lemma 1.6.7 allows us to efficiently compute isogenies between products of elliptic curves from their kernel *if it has powersmooth cardinality.* It is also possible to be efficient when the kernel has cardinality equal to a power of a small prime. For instance, the case of isogenies of degree a power of 2 in dimension 4 has been studied in [Dar24]. This is the situation encountered in Section 4.3.

As presented by Robert in [Rob22b], isogenies between abelian varieties can be embedded into isogenies of higher dimensions. Namely, given an isogeny φ between abelian varieties, one can construct a higher dimensional isogeny such that one of its matrix form coefficients is equal to f, up to isomorphism. This result is a generalisation of a construction in dimension 1 given by Kani in [Kan97]; we refer to it as Kani's Lemma. The diagrams involved in these results are called **isogeny diamonds**, **Kani diagrams** or **Kani squares**.

Definition 1.6.8. *A **(d_1, d_2)-isogeny diamond** is a commutative diagram*

$$\begin{tikzcd} A \arrow[r, "\varphi_1"] \arrow[d, "\varphi_2"'] & \varphi_1(A) \arrow[d, "\varphi_1'"] \\ \varphi_2(A) \arrow[r, "\varphi_2'"] & B \end{tikzcd}$$

where f_1, f_2' are d_1-isogenies and f_2, f_1' are d_2-isogenies between principally polarised abelian varieties. In particular, $\varphi_1' \circ \varphi_1 = \varphi_2' \circ \varphi_2$.

Lemma 1.6.9 (Lemma 3.6. in [Rob23a])**.** *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 with $(d_1 + d_2, p) = 1$ such that they form the following (d_1, d_2)-isogeny diamond*

$$\begin{tikzcd} A \arrow[r, "\varphi_1"] \arrow[d, "\varphi_2"'] & \varphi_1(A) \arrow[d, "\varphi_1'"] \\ \varphi_2(A) \arrow[r, "\varphi_2'"] & B \end{tikzcd} .$$

Then

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

is the matrix form of a $(d_1 + d_2)$-isogeny $F : A \times B \to \varphi_1(A) \times \varphi_2(A)$. In addition, 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 conclude this section with a final proposition stating that an isogeny of composite degree between principally polarised abelian varieties can be factored into isogenies of smaller degree, generalising the case of elliptic curves.

Proposition 1.6.10 ([DEF$^+$25b, Lemma 3])**.** *Let A and B be two principally polarised abelian varieties over a field of characteristic p and $\varphi : A \to B$ be a n-isogeny between them. For every pair of positive coprime integers n_1, n_2 coprime to p such that $n = n_1 n_2$, there exists a PPAV C, a n_1-isogeny $\varphi_1 : A \to C$ and a n_2-isogeny $\varphi_2 : C \to B$, each uniquely defined up to isomorphism, such that $\varphi = \varphi_2 \circ \varphi_1$.*

Chapter 2

Isogeny Interpolation and applications

In this chapter, we explore important consequences of the breaking of SIDH, which constitute precious tools for the subsequent chapters. In Section 2.2, we describe one of these tools, which is an original contribution: an efficient algorithm to divide isogenies. This result is based on Robert's division algorithm [Rob22b] and has been published in:

[HW25a] Arthur Herlédan Le Merdy and Benjamin Wesolowski. The supersingular endomorphism ring problem given one endomorphism. IACR Communications in Cryptology*, 2(1), 2025*

One of the isogeny-based protocols with the richest history is `SIDH`, for **S**upersingular **I**sogeny **D**iffie–**H**ellman [JD11]. It was the first *practical* key exchange scheme relying on isogenies. As a selected candidate in the NIST standardisation process on post-quantum cryptography [NIS16], `SIDH` remained the standard-bearer of isogeny-based cryptography until its fall in 2022 with the groundbreaking attacks [CD23, MMP$^+$23, Rob23a]. From its ashes, a wide variety of powerful tools have bloomed, leading to the development of numerous original cryptosystems [BDD$^+$24, BM25, Ler25] and a deeper understanding of the field [CP25, ES24].

In Section 2.1, after a brief presentation of `SIDH`, we describe one of the most versatile versions of the `SIDH` attack: an `IsogenyInterpolation` algorithm due to Robert [Rob23a]. This algorithm opens the possibility of efficiently representing isogenies without any constraint on their degree — compared to the previous state of the art where efficiently represented isogenies were typically chains of smooth-degree isogenies. This result deeply relies on higher dimensional isogenies and on the now well-known Kani's Lemma. In Section 2.2, we prove that these new tools can also be used to divide any isogeny in polynomial time — generalising a result from [Rob22b]. Prior to this contribution, it was only feasible to divide isogenies by powersmooth integers, which required assuming heuristics in many situations. Then, in Section 2.3, we describe two last applications of the higher dimension machinery: the `CLAPOTI` algorithm, introduced in [PR23], to compute class group actions efficiently, and its direct generalisation to an unconditional `IdealToIsogeny` algorithm running in polynomial time. In the previous literature, such algorithms were only efficient for ideals with smooth norms.

These powerful results are crucial for the work presented in this thesis, in particular for studying the relations between hard problems in Chapter 3. In addition, in Chapter 4, `CLAPOTI` and the division algorithm are central to the rigorous cryptanalysis of oriented problems. Finally, the practical effective group action `PEGASIS` [DEF$^+$25a], also presented in Chapter 4, is built as a practical instantiation of `CLAPOTI`.

2.1 The fall of SIDH by Interpolation

The history of key exchange in isogeny-based cryptography began with the impractical `CRS` scheme [Cou06, RS06]. Twelve years later, `CSIDH` [CLM$^+$18] addressed its efficiency limitations by leveraging supersingular elliptic curves. However, while the best attacks on the Isogeny problem are exponential, recovering the shared secret from a `CSIDH`-like exchange takes only quantum subexponential time. Hence, isogeny-based cryptography should be able to guarantee better security for a key exchange.

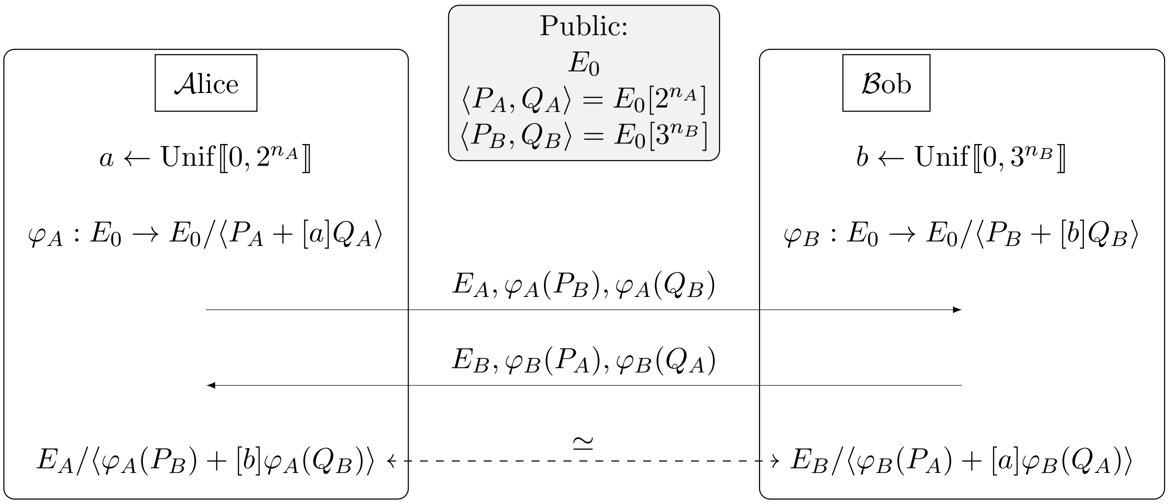
Prior to the introduction of `CSIDH`, a line of research orthogonal to `CRS` led to the `SIDH` scheme [JD11]. This key exchange solved both the efficiency and security issues of `CRS`. Unfortunately, this was only a temporary success. Its main vulnerability lay in the requirement to reveal the images of several points under a secret isogeny in order to perform the key exchange, thereby weakening the Isogeny problem. This concern ultimately proved to be a significant flaw, enabling a polynomial time attack due to the deployment of the higher dimensional machinery [CD23, MMP+23, Rob23a].

A brief description of `SIDH`. Before presenting Robert's attack by interpolation, let us introduce the `SIDH` key exchange scheme. The core idea behind `SIDH`, common to other isogeny-based schemes such as `CRS`, is to translate the Diffie–Hellman key exchange into the world of elliptic curves and isogenies, in such a way that the difficulty of recovering the shared key relies on the hard problem of finding an isogeny between two elliptic curves.

Namely, the process for the two parties wishing to agree on a key is to compute a secret isogeny from a common public elliptic curve and to make its codomain public. Then, each of them "applies" their secret isogeny to the codomain of the other party, in order to recover the same target elliptic curve. Unfortunately, it does not really make sense to "apply" an isogeny to a curve other than its original domain. This issue is addressed in `CRS` by relying on group actions. The adopted solution in `SIDH` is to publish the codomain of the secret isogenies together with their image on a specific subgroup. Then, each party can compute the pushforward of their secret isogeny on the other curve to obtain the same isomorphism class of elliptic curves. Thus, computing the j-invariant of this final curve provides a shared secret.

The public parameters are:

- A public elliptic curve $E_0/\mathbb{F}_{p^2}$ such that $\#E_0(\mathbb{F}_{p^2}) = (p+1)^2$, where p is a prime of the form $2^{n_A}3^{n_B} - 1$ with $2^{n_A} \simeq 3^{n_B} \simeq \sqrt{p}$,
- A pair of points P_A, Q_A generating the torsion subgroup $E_0[2^{n_A}]$,
- A pair of points P_B, Q_B generating the torsion subgroup $E_0[3^{n_B}]$.



Public:
E_0
$\langle P_A, Q_A\rangle = E_0[2^{n_A}]$
$\langle P_B, Q_B\rangle = E_0[3^{n_B}]$
$\mathcal{A}$lice
$\mathcal{B}$ob
$a \leftarrow \mathrm{Unif}[\![0, 2^{n_A}]\!]$
$b \leftarrow \mathrm{Unif}[\![0, 3^{n_B}]\!]$
$\varphi_A : E_0 \to E_0/\langle P_A + [a]Q_A\rangle$
$\varphi_B : E_0 \to E_0/\langle P_B + [b]Q_B\rangle$
$E_A, \varphi_A(P_B), \varphi_A(Q_B)$
$E_B, \varphi_B(P_A), \varphi_B(Q_A)$
$E_A/\langle\varphi_A(P_B) + [b]\varphi_A(Q_B)\rangle$
$\simeq$
$E_B/\langle\varphi_B(P_A) + [a]\varphi_B(Q_A)\rangle$

Figure 2.1: `SIDH` key exchange.

The interpolation attack. We now state a recent formulation of Robert's interpolation algorithm, which is suitable for various applications. The proof of the result will be omitted here, however, in the next section, we prove a generalisation of Robert's idea to divide isogenies. The approach in both proofs is similar.

Proposition 2.1.1 (`IsogenyInterpolation` [Rob24, Theorem 5.19]). *Let $\varphi : E \to E'$ be an n-isogeny between supersingular elliptic curves defined over $\mathbb{F}_{p^2}$. Let $N > n$ be an integer coprime to pn with prime power decomposition $\prod_{i=1}^r \ell_i^{e_i}$. Let $(P_1, Q_1, \ldots, P_r, Q_r)$ be a set of generators of the N-torsion $E[N]$ such that (P_i, Q_i) is a basis of $E[\ell_i^{e_i}]$, for $i = 1, \ldots, r$.*

Then, given $(\deg\varphi, P_1, Q_1, \ldots, P_r, Q_r, \varphi(P_1), \varphi(Q_1), \ldots, \varphi(P_r), \varphi(Q_r))$, one can compute an efficient representation of φ in polynomial time in the length of the input and in the largest prime factor of N.

This powerful proposition has found numerous applications. As a striking example of its groundbreaking nature, we show that the attack on `SIDH` is a direct corollary.

Corollary 2.1.2 ([Rob23a]). *There is an algorithm that recovers the shared key of an `SIDH` key exchange in time polynomial in the length of the input.*

Proof. Let $E_0, P_A, Q_A, P_B, Q_B, n_A, n_B$ be the public parameters of a `SIDH` key exchange. We assume that $3^{n_B} > 2^{n_A}$. Let $E_A, \varphi_A(P_B), \varphi_A(Q_B), E_B, \varphi_B(P_A), \varphi_B(Q_A)$ be the information shared by both parties during the exchange.

Since the isogeny φ_A has degree $2^{n_A} < 3^{n_B}$, one can compute an efficient representation of φ_A from the tuple

$$(\deg\varphi_A, P_B, Q_B, \varphi_A(P_B), \varphi_A(Q_B))$$

using Proposition 2.1.1. With this representation, computing $\varphi_A(P_A)$ and $\varphi_A(Q_A)$ takes polynomial time.

Then finding an integer m such that $\varphi_A(P_A) = [m](\varphi_A(Q_A))$ can be done in polynomial time using the Pohlig-Hellman algorithm in the cyclic group $\varphi_A(E_0[2^{n_A}])$ of order 2^{n_A}. Since

$$\varphi(P_A + [a]Q_A) = 0 = \varphi(P_A - [m]Q_A),$$

we have that

$$-m \equiv a \mod 2^{n_A}.$$

We have recovered Alice's secret. Then by computing the codomain of the isogeny of E_B with kernel $\langle \varphi_B(P_A) - m\varphi_B(Q_A) \rangle$, we recover the shared key. Recovering the key when $3^{n_B} < 2^{n_A}$ is analogous. □

Another powerful direct application of the `IsogenyInterpolation` is the possibility of representing any dual isogeny efficiently.

Lemma 2.1.3 ([Rob24, Proposition 6.4]). *Let $\varphi : E \to E'$ be an isogeny defined over $\mathbb{F}_{p^k}$. Given an efficient representation of the isogeny φ, one can compute an efficient representation of its dual $\hat{\varphi}$ in time polynomial in $\log(\deg(\varphi))$ and $k\log(p)$.*

Proof. In order to obtain an efficient representation of the dual $\hat{\varphi}$, one only needs to find a suitable powersmooth integer N and to take advantage of the efficient representation of φ to apply the `IsogenyInterpolation` algorithm given by Proposition 2.1.1.

We compute N by multiplying successive primes, coprime to $p\deg(\varphi)$, until their product exceeds than $\deg(\varphi)$. It takes $O(\log\deg(\varphi))$ arithmetic operations and provides an integer N which is $O(\log\deg(\varphi))$-powersmooth and such that $\log N = O(\log\deg(\varphi))$. Then, by Proposition 1.3.19, computing bases (P_i, Q_i) of the ℓ_i-torsion subgroup $E[\ell_i]$, for $\ell_1, \ldots, \ell_r$ the prime factors of N, takes time polynomial in $\log \deg \varphi$ and $k \log p$. By coprimality, for any $i \in [\![1, r]\!]$, the pair $(\varphi(P_i), \varphi(Q_i))$ is a basis of the ℓ_i-torsion subgroup of E'. Moreover, the image of these bases by $\hat{\varphi}$ are simply $(\hat{\varphi} \circ \varphi(P_i), \hat{\varphi} \circ \varphi(Q_i)) = ([\deg \varphi]P_i, [\deg \varphi]P_i)$. Thus, by Proposition 2.1.1 and by the efficient representation of φ, the tuple $(\deg \varphi, \varphi(P_i), \varphi(Q_i), [\deg \varphi]P_i, [\deg \varphi]Q_i)$ can be used to interpolate the isogeny $\hat{\varphi}$ in time polynomial in $\log(\deg(\varphi))$ and $k \log(p)$. □

2.2 Division algorithm

In this section, we discuss how higher dimensional isogenies can be used to efficiently divide isogenies. This is an original contribution published in [HW25a] based on ideas developed in [Rob22b]. The goal is to prove Theorem 2.2.1 below (and its more precise formulation Theorem 2.2.7).

Theorem 2.2.1. *Algorithm 1 takes as input*

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

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

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

Before proving it, let us state one of its direct corollaries once combined with Lemma 2.1.3.

Corollary 2.2.2 (General division of isogenies)**.** *There is an algorithm which on input*

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

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

Proof. Apply Theorem 2.2.1 to the isogeny $\tilde{\varphi} = \varphi \circ \hat{\eta}$ and the integer $n = \deg(\eta)$. The isogeny ψ, if it exists, is precisely $\tilde{\varphi}/n$. This computation is possible in polynomial time as φ and η are efficiently represented. In particular, by Lemma 2.1.3, this means we also have access to an efficient representation of $\hat{\eta}$. □

Before proving Theorem 2.2.1, it is important to note that, given an `IsogenyInterpolation` algorithm, one can efficiently divide isogenies, following a method similar to the one used in Lemma 2.1.3 to compute dual isogenies. The core idea is that we can evaluate an isogeny $\varphi : E \to E'$ divided by an integer n over the N-torsion subgroup $E[N]$ when N is coprime to n. Indeed, in this case, we have

$$\frac{\varphi}{n}(P) = m\varphi(P), \forall P \in E[N],$$

with m the inverse of n modulo N. Hence, when N is powersmooth and large enough, the data set

$$(\deg(\varphi), P, Q, m\varphi(P), m\varphi(Q)), \text{ with } \langle P, Q \rangle = E[N]$$

allows us to efficiently interpolate the isogeny $\frac{\varphi}{n}$. However, sometimes we do not know a priori if the isogeny φ is divisible by the integer n. By Definition 1.3.21, a data set that does not represent an isogeny will be detected by an `IsogenyInterpolation` algorithm. Thus, `IsogenyInterpolation` algorithms provide a method to check wether a division is well-defined and to compute it if it is.

Nevertheless, the behaviour of `IsogenyInterpolation` algorithms in such cases was unclear at the time of the writing [HW25a]. Moreover, some necessary steps to obtain a general algorithm were missing. This is why to provide a division algorithm, we had to study in detail the `IsogenyInterpolation` algorithm in cases where the input can be wrong. Our contribution is an `IsogenyDivision` algorithm with a complete proof of its correct behavior together with a detailed complexity analysis. The `IsogenyDivision` algorithm presented in this section can also be used to properly check if a set of data correctly represents an isogeny.

Let us emphasise that the ideas underlying Theorem 2.2.1 and its proof originate from [Rob22b]. Theorem 2.2.1 and its proof are simply expressed in higher generality and greater detail than [Rob22b] provides. Among other ingredients, this attack relies on the generalization of Vélu's formulae by Lubicz and Robert, see [LR12].

In further results of this section, we shall need to recover endomorphisms of a given product of elliptic curves $E^n \times E'^n$ from its kernel. Thus, it is important to have a description of group of automorphisms of $E^n \times E'^n$ as, by Lemma 1.6.6, endomorphisms with the same kernel differ only by an automorphism.

Lemma 2.2.3. *Let E_1 and E_2 be two elliptic curves and n, m be two integers. Let $\mathrm{Aut}(E_1^n \times E_2^m, \lambda_{E_1^n \times E_2^m})$ be the group of automorphisms of the principally polarised abelian variety $(E_1^n \times E_2^m, \lambda_{E_1^n \times E_2^m})$. Then for any element ψ of $\mathrm{Aut}(E_1^n \times E_2^m, \lambda_{E_1^n \times E_2^m})$, we have*

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

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

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

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

This implies that for any i, exactly one of the isogenies $\psi_{i,j}$ is non-zero, and that isogeny has degree one, hence it is an isomorphism. The identity $\tilde{\psi}\psi = [1]$ yields the same results for columns. Moreover, for ψ to be well-defined, the domain and codomain of each $\psi_{i,j}$ must be

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

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

□

Lemma 2.2.4 below describes how an isogeny φ of degree N between elliptic curves can be embedded into an N'-endomorphism in dimension 8, for some $N' > N$. This embedding can then be evaluated efficiently using Lemma 1.6.7.

Lemma 2.2.4. *Let E_1 and E_2 be two elliptic curves over a finite field $\mathbb{F}_{p^k}$ and $\varphi : E_1 \to E_2$ be an isogeny of degree N. Let $N' > N$ be an integer such that $(N', Np) = 1$. Let m_1, m_2, m_3 m_4 be integers such that $m_1^2 + m_2^2 + m_3^2 + m_4^2 = N' - N$ and let α_{E_1} (resp. α_{E_2}) be the endomorphism over E_1^4 (resp. E_2^4) given by the matrix*

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

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

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

$$\begin{tikzcd} E_1 \arrow[r, "G \circ \tau_1"] \arrow[d, "\varphi"'] & G(E_1^4 \times E_2^4) \arrow[d, "\pi_i \circ \gamma"] \\ E_2 \arrow[r, "\psi"] & E' \end{tikzcd}$$

i.e. $\pi_i(\gamma(G(\tau_1(P)))) = \psi(\varphi(P))$, for all $P \in E_1$, and thus $(\gamma \circ G, 1, i)$ is an embedding representation of $\psi \circ \varphi$, see Definition 1.6.2.

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

$$\begin{tikzcd} E_1^4 \arrow[r, "\alpha_{E_1}"] \arrow[d, "\varphi^{\times 4}"'] & E_1^4 \arrow[d, "\varphi^{\times 4}"] \\ E_2^4 \arrow[r, "\alpha_{E_2}"] & E_2^4. \end{tikzcd}$$

By construction of $\varphi^{\times 4}$ is an N-isogeny and α_{E_1} and α_{E_2} are $(N' - N)$-isogenies. Thus, the sum of the degree of $\varphi^{\times 4}$ with the degree of α_{E_1} or α_{E_2} is equal to N'. By assumption, N' is coprime to Np. In particular N' is coprime to p and N is coprime to $N' - N$. Hence, by taking $A := E_1^4, B := E_2^4, \varphi_1 := \alpha_{E_1}, \varphi_2' := \alpha_{E_2}, \varphi_2 := \varphi^{\times 4}$ and $\varphi_1' := \varphi^{\times 4}$, we have $d_1 = N' - N$, $d_2 = N$ and all the assumptions of Lemma 1.6.9 are satisfied. Its application gives us an N'-endomorphism F of $E_1^4 \times E_2^4$ with kernel

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

and matrix form

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

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

$$F(\tau_1(P)) = (m_1P, m_2P, m_3P, m_4P, -\varphi(P), 0, 0, 0) \tag{2.1}$$

Let G be an N'-isogeny of $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ with $\ker G = \ker F$. By Lemma 1.6.6, $G(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ and $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ are isomorphic and for any isomorphism γ from $G(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ to $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ there exists an automorphism ψ of $E_1^4 \times E_2^4$ such that we have

$$\gamma \circ G = \psi \circ F. \tag{2.2}$$

By Lemma 2.2.3, there exist 8 isomorphisms $\psi_1, \dots \psi_8$ such that

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

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

$$\psi(P_1, \dots, P_8) = \sigma(\psi_1(P_1), \dots, \psi_8(P_8)). \tag{2.3}$$

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

$$\begin{aligned} \pi_i(\gamma(G(\tau_1(P)))) &= \pi_i(\psi(F(\tau_1(P)))), \text{ by (2.2)}, \\ &= \pi_i(\sigma((\psi_1(m_1P), \dots, \psi_4(m_4P), \psi_5(-\varphi(P)), 0, 0, 0))), \text{ by (2.1)}, \\ &= -\psi_5(\varphi(P)), \text{ by construction } \pi_i. \end{aligned}$$

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

Remark 2.2.5. *In this section, we always assume that the isogenies are embedded into dimension 8. Lemma 1.6.7, 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 [Zar74] as done in Lemma 2.2.4. For dimension 2 (resp. 4), $N' - N$ needs to be a square (resp. a sum of two squares) to construct easily an endomorphism of E (resp. E^2) and to embed the isogenies into dimension 2 (resp. 4). It is possible to relax these conditions, under some heuristics and when the endomorphism ring is known. Here, we neither want to rely on heuristics, nor presume that we know the endomorphism ring, so we only consider the case of dimension 8.*

By Lemma 2.2.4, if an isogeny $\varphi : E_1 \to E_2$ is divisible by an integer n then it can be embedded into an isogeny in dimension 8 of kernel

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

with N' and α_{E_1} defined as in the lemma. Thanks to Lemma 1.6.7, one can compute in polynomial time this 8-dimensional isogeny from its kernel. It only remains to show how we can compute the kernel H to get a complete division algorithm.

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

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

This trick allows us to compute the kernel H from an efficient representation of φ. Lemma 2.2.6 ensures that computing H this way always provides a maximal isotropic subgroup of $E_1^4 \times E_2^4$ even if φ/n is not a well-defined isogeny. This will be crucial to verify the divisibility of an isogeny by an integer.

Lemma 2.2.6. *Let $\varphi : E_1 \to E_2$ be an isogeny. Let n^2 be a divisor of $\deg(\varphi)$ and $N = \deg(\varphi)/n^2$. Let $N' > N$ such that $(N', p\deg(\varphi)) = 1$ and $s = n^{-1} \mod N'$. Let α_{E_1} be an m-endomorphism of E_1^4 with $m = N' - N$.*
Then $H := \{(\tilde{\alpha}_{E_1}(P), s\varphi^{\times 4}(P)) | P \in E_1^4[N']\}$ is a maximal isotropic subgroup of $(E_1^4 \times E_2^4)[N']$.

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

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

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

□

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

Theorem 2.2.7. *Algorithm 1 is correct and runs in*

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

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

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

Algorithm 1 `IsogenyDivision`

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

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

```
Set $N \leftarrow \deg(\varphi)/n^2$.
if $N \notin \mathbb{N}$ then
    return False
Set $m \leftarrow N' - N$.
Decompose $m$ as $m_1^2 + m_2^2 + m_3^2 + m_4^2$.
Set $M \leftarrow \begin{pmatrix} m_1 & -m_2 & -m_3 & -m_4 \\ m_2 & m_1 & m_4 & -m_3 \\ m_3 & -m_4 & m_1 & m_2 \\ m_4 & m_3 & -m_2 & m_1 \end{pmatrix}$.
Let $\alpha$ be the $m$-endomorphism over $E_1^4$ given by the matrix $M$.
Let $\tilde{\alpha}$ be the dual isogeny of $\alpha$ with respect to the product polarisation.
$s \leftarrow n^{-1} \mod N'$.
Compute a factorisation $\ell_1^{e_1} \dots \ell_r^{e_r}$ of $N'$.
Compute bases $(P_{1,i}, P_{2,i})$ of $E_1[\ell_i^{e_i}]$ for $i \in [\![1, r]\!]$.
Set $(P_1, P_2) \leftarrow (\sum_{i=1}^r P_{1,i}, \sum_{i=1}^r P_{2,i})$ a basis of $E_1[N']$.
Compute a representation of an $N'$-isogeny $F$ of $E_1^4 \times E_2^4$ of kernel
    $$\ker F = \{(\tilde{\alpha}_{E_1}(\tau_i(P_j)), s\varphi^{\times 4}(\tau_i(P_j))) | \forall i \in [\![1,4]\!], \forall j \in \{1,2\}\}.$$
if $F(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4}) \not\simeq (E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ then
    return False.
Set $\gamma : F(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4}) \to (E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ be an isomorphism of principally
    polarised abelian varieties.
if $E_1 \simeq E_2$ then
    Set $\psi_0 : E_2 \to E_1$ to be an isomorphism.
else
    Set $\psi_0 : E_2 \to E_1$ to be the zero map.
Compute the sets of maps $S_{E_1} := \mathrm{Aut}(E_1)\psi_0$ and $S_{E_2} := \mathrm{Aut}(E_2)$.
for $t \in [\![1, 8]\!]$ do
    for $\psi \in S_{E_1}$, if $1 \leq t \leq 8$, or $\psi \in S_{E_2}$, if $5 \leq t \leq 8$, do
        if $n(\psi^{-1} \circ \pi_t \circ \gamma \circ F \circ \tau_1(P_{i,j})) = \varphi(P_{i,j}), \forall i \in \{1,2\}, \forall j \in [\![1, r]\!]$ then
            return The representation of $\varphi/n$ induced by $\psi^{-1} \circ \pi_t \circ \gamma \circ F \circ \tau_1$.
return False
```

Proof. Let us prove the correctness of Algorithm 1.

First, by Lemma 2.2.6, $\ker F$ is always a maximal isotropic subgroup of $(E_1^4 \times E_2^4)[N']$ and thus the isogeny F is well defined.

When $\varphi/n : E_1 \to E_2$ is an isogeny, we have that $(\varphi/n)|_{E[N']} = (s\varphi)|_{E[N']}$. Hence, by Lemma 2.2.4, F is isomorphic to an N'-isogeny that embeds φ/n. More precisely, $F(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4}) \simeq (E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ as principally polarised abelian varieties and for any isomorphism ψ between them, there exist an isomorphism $\psi : E_2 \to E'$, where $E' \in \{E_1, E_2\}$, and an integer $t \in [\![1, 8]\!]$ such that

$$\pi_t(\gamma(F(\tau_1(P)))) = \psi(\varphi/n)(P), \forall P \in E_1. \tag{2.4}$$

We check at line 14 if we can find such isomorphism γ. If this is the case, we fix an isomorphism $\gamma : F(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4}) \to (E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$. Otherwise, by Lemma 2.2.4, φ/n is not an isogeny and the algorithm must return `False`.

We then look for an isomorphism ψ and an integer t verifying Equality (2.4). To satisfy the equality, ψ needs to have the same codomain as $\pi_t \circ \gamma \circ F$ which is equal to E_1 if $1 \leq t \leq 4$ and E_2 if $5 \leq t \leq 8$. In the first case, ψ is an element of $\mathrm{Aut}(E_1)\psi_0$, where $\psi_0 : E_2 \to E_1$ is an isomorphism. In the second case, ψ is an automorphism of E_2.

In the for loop, we search for a solution (ψ, t) of Equality (2.4) over the bases of $E_1[\ell_i^{e_i}], \forall i \in [\![1, r]\!]$. It is equivalent to checking Equality (2.4) over $E_1[N']$ as $(\ell_i, \ell_j) = 1, \forall i \neq j \in [\![1, r]\!]$. Moreover as $N' > \deg \varphi$ and $\deg \varphi \geq 2$, a solution of (2.4) over $E_1[N']$ is a solution over the entire elliptic curve E_1. Indeed, if two isogenies φ and φ' of same degree are equal over $E_1[N']$ with $N' > \deg \varphi \geq 2$, then they are equal everywhere. Let us assume that $\phi := \varphi - \varphi'$ is a non-zero isogeny. We have $\phi(E_1[N']) = 0$, hence

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

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

Since we are doing an exhaustive search at line 22, if φ/n is an isogeny, the algorithm will find an embedding representation $(F, 1, t)$ of φ/n up the two isomorphims ψ and γ. If no such coefficient of F is found, Lemma 2.2.4 implies that φ/n is not an isogeny.
The output representation of φ/n is then given by the composition of the representation of ψ^{-1} with the embedding representation $(\gamma \circ F, 1, t)$.

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

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

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

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

By Lemma 1.6.7, getting a representation of the N'-isogeny F takes :

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

Then, it remains to compute an isomorphism γ between the codomain of F and $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$. As described in [DLRW24, Appendix F.3], one can perform an exhaustive search over the symplectic group $\mathrm{Sp}_{16}(\mathbb{Z}/4\mathbb{Z})$, which consists of the set of 16×16 matrices that preserve symplectic form, to find a matrix that sends the theta coordinates of the codomain of F to the theta coordinates of $(E_1^4 \times E_2^4, \lambda_{E_1^4 \times E_2^4})$ (see Remark 1.6.1 for the construction of the latter). This step does not impact the overall complexity, as the number of possibilities is constant for a given dimension.[1] If no such linear transformation is found, then the two principally polarised abelian varieties are not isomorphic.

[17 - 21]:
These steps are done efficiently using basics of elliptic curves' theory. Checking if two elliptic curves are isomorphic can be done using the j-invariant, while an explicit isomorphism can be obtained by a small computation from the equations of the curves; see Proposition 1.3.13. Then, explicit description of the automorphism groups are easy to compute; see Proposition 1.3.23.

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

We get the claimed complexity by summing all those steps.

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

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

When the input of Algorithm 1 is efficiently represented, it leads to Theorem 2.2.1 which concludes this section about efficient division of isogenies.

Proof of Theorem 2.2.1. As in the proof of Lemma 2.1.3, to get this result, one only needs to find a suitable powersmooth integer N' and to take advantage of the efficient representation of φ. Again the integer N' is computed as the smallest product of successive primes, coprime to $p \deg(\varphi)$, which is greater than $\deg(\varphi)$. The obtained integer N' is then $O(\log \deg(\varphi))$-powersmooth and verifies that $\log N' = O(\log \deg(\varphi))$. Hence, with the same notation as in Theorem 2.2.7, 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.

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

[1] As this constant is very large, it is preferable, for more practical usage, to directly compute the matrix we are looking for, see [DLRW24, Appendix F.3].

2.3 The `CLAPOTI` breakthrough

We conclude this chapter with two additional applications of Kani's Lemma. First, we introduce the *unconditional* and *unrestricted* algorithm `CLAPOTI` [PR23], for **CL**ass group **A**ction in **PO**lynomial **TI**me, which, as the name suggests, computes class group actions in polynomial time. This result concerns the class group actions provided by oriented elliptic curve; see Section 1.4. Second, we present its direct generalisation: an efficient *unconditional* and *unrestricted* algorithm for computing the Deuring correspondence from ideals to isogenies in polynomial time, i.e. an `IdealToIsogeny` algorithm. We emphasise that these algorithms are unrestricted, in the sense that there are no constraints on the input ideal, and unconditionally because their complexities are proven without assuming heuristics or **GRH**.

In the previous state of the art, acting on oriented elliptic curves was only possible for ideals of powersmooth norm. This improvement has important consequences in our work; both from a theoretical perspective — it plays a key role in the complexity analyses in Section 4.2 — and for practical purposes — the effective group action algorithm introduced in Section 4.3 is based on `CLAPOTI`.

On the other hand, the previous `IdealToIsogeny` algorithms were only efficient to translate ideals of powersmooth norm; see for instance [GPS17, Lemma 5] or [EHL$^+$18, Proposition 4]. To fulfill this requirement, one had to use `KLPT`-like algorithms [KLPT14], such as [Wes22b, Theorem 6.4] which is proven under **GRH**, to find equivalent ideals with suitable norms. Hence by removing this smoothness constraint, `CLAPOTI` also removes the need for **GRH** from proofs that rely on such `IdealToIsogeny` translation. This observation is crucial to our proof of unconditional equivalence of hard problems in Section 3.2. Note that there are other `IdealToIsogeny` algorithms which do not require the ideal norm to be powersmooth. For instance in [DLLW23] an algorithm to efficiently translate ideals with norms that are powers of small primes is presented; thereby providing an efficient `IdealToIsogeny` for ideals of smooth norm. Unfortunately, these algorithms are heuristics.

Let us first state the `CLAPOTI` algorithm in a ready-to-use form to compute the isogeny $\varphi_\mathfrak{a} : E \to E^\mathfrak{a}$ for an $\mathfrak{O}$-ideal $\mathfrak{a}$ acting on an $\mathfrak{O}$-oriented elliptic curve (E, ι).

Proposition 2.3.1 ([PR23, Theorem 2.9])**.** *Let (E, ι) be an $\mathfrak{O}$-oriented supersingular elliptic curve defined over $\mathbb{F}_{p^2}$ with $\mathfrak{O}$ a quadratic imaginary order of discriminant Δ. Given an integral invertible $\mathfrak{O}$-ideal $\mathfrak{a}$, one can compute an efficient representation of $\varphi_\mathfrak{a} : E \to E^\mathfrak{a}$ in probabilistic time polynomial in $\log p$, $\log N(\mathfrak{a})$ and in the length of the representation of ι.*

From this isogeny, one can efficiently compute the induced orientation on $E^\mathfrak{a}$ — and thus the complete action $\mathfrak{a} \star (E, \iota)$ — using the higher dimension tools presented earlier in this section.

Corollary 2.3.2. *Let (E, ι) be an $\mathfrak{O}$-oriented supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. One can compute an efficient representation of $\mathfrak{a} \star (E, \iota)$ in time probabilistic polynomial in $\log p$, $\log N(\mathfrak{a})$ and in the length of the representation of ι.*

Proof. By Proposition 2.3.1, one can recover an efficient representation of the isogeny $\varphi_\mathfrak{a} : E \to E^\mathfrak{a}$ in probabilistic time polynomial in $\log p$, $\log N(\mathfrak{a})$ and in the length of the representation of ι. To obtain an efficient representation of $\mathfrak{a}\star(E, \iota)$, it remains to compute $(\varphi_\mathfrak{a})_*(\iota)$ the orientation induced by $\varphi_\mathfrak{a}$ over $E^\mathfrak{a}$. We recall that, as always, oriented elliptic curves given as input are assumed to be efficiently represented. In particular, there is a generator ω of the order $\mathfrak{O}$ associated to the representation of (E,ι). Then to get an efficient representation of $(\varphi_{\mathfrak{a}})_*(\iota)$, it is sufficient to compute an efficient representation of the evaluation of this orientation at ω. In other words, one only needs to compute an efficient representation of the endomorphism

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

As, by assumption and by Lemma 2.1.3, $\iota(\omega)$, $\varphi_{\mathfrak{a}}$ and its dual $\hat{\varphi}_{\mathfrak{a}}$ are efficiently represented, their composition is also efficiently represented. Thus, one can obtain an efficient representation of $(\varphi_{\mathfrak{a}})_*(\iota)(\omega)$ by dividing the composition by $\mathrm{N}(\mathfrak{a})$. By Theorem 2.2.1, this computation takes time polynomial in $\log p$, $\log \mathrm{N}(\mathfrak{a})$ and in the length of the representation of ι. □

As mentioned before, this result has been generalised to perform the generic `IdealToIsogeny` algorithm of the Deuring correspondence. The isogeny corresponding to any given $\mathcal{O}$-ideal, for a maximal order isomorphic to the endomorphism ring of an elliptic curve, can now be computed in polynomial time *unconditionally.* Note that in order to perform such translation, it is mandatory to know explicitly the Deuring isomorphism between an endomorphism ring and a maximal order for at least one elliptic curve.

This generalisation of `CLAPOTI` has been made for instance in [BDD$^+$24] using 2-dimensional isogenies for the sake of efficiency, at the cost of heuristics. Another recent example of such algorithms is given in [ON24]. Considering abelian varieties of dimension 8 ensures a rigorous polynomial time algorithm.

Proposition 2.3.3 (Unconditional `IdealToIsogeny` algorithm [PR23])**.** *Let E be a supersingular elliptic curve defined over $\mathbb{F}_q$ such that an isomorphism $\varepsilon : \mathrm{End}(E) \simeq \mathcal{O}$ is known, where $\mathcal{O}$ is a maximal order in some quaternion algebra isomorphic to $B_{p,\infty}$. Given a left $\mathcal{O}$-ideal I, one can compute an efficient representation of the isogeny $\varphi_I : E \to E/E[I]$ in polynomial time in the length of the input.*

To conclude this section, we provide some insight into the main ideas behind `CLAPOTI` algorithm. Indeed, as we heavily rely on them to build the `PEGASIS` algorithm in Section 4.3, it is important to first explain how `CLAPOTI` works.

Sketch of the proof of `CLAPOTI`. Let $\mathfrak{O}$ be an order in an imaginary quadratic number field. Let (E,ι) be a primitively $\mathfrak{O}$-oriented elliptic curve and $\mathfrak{a}$ an $\mathfrak{O}$-ideal.

In `CLAPOTI`, the first step to compute the action $\mathfrak{a} \star (E,\iota)$ is to find two $\mathfrak{O}$-ideals of coprime norm $\mathfrak{b}$ and $\mathfrak{c}$, both equivalent to $\mathfrak{a}$, such that

$$\mathrm{N}(\mathfrak{b}) + \mathrm{N}(\mathfrak{c}) = N, \tag{2.5}$$

with N a smooth integer coprime to p where the torsion subgroup $E[N]$ is accessible, i.e. easy to compute. The coprimality conditions are needed to apply Kani's Lemma, Lemma 1.6.9, while the smoothness and the accessibility of the torsion is required to process the next steps in polynomial time. From now on, we shall refer to Equation (2.5) as the `CLAPOTI` equation.

Once such a pair $(\mathfrak{b},\mathfrak{c})$ is found, we consider the following diagram

$$\begin{tikzcd} E \arrow[r, "\varphi_{\mathfrak{b}}"] \arrow[d, "\varphi_{\overline{\mathfrak{c}}}"'] & E^{\mathfrak{b}} \arrow[d, "\varphi'_{\overline{\mathfrak{c}}}"] \\ E^{\overline{\mathfrak{c}}} \arrow[r, "\varphi'_{\mathfrak{b}}"] & E \end{tikzcd} .$$

By Proposition 1.4.7 and Proposition 1.3.12, this is a commutative diagram up to isomorphism on the codomain E. In particular, $\varphi'_{\mathfrak{b}} \circ \varphi_{\bar{\mathfrak{c}}}$ and $\varphi'_{\bar{\mathfrak{c}}} \circ \varphi_{\mathfrak{b}}$ both have kernel equal to $E[\mathfrak{b}\bar{\mathfrak{c}}]$.

Since the ideals $\mathfrak{b}$ and $\mathfrak{c}$ are equivalent to the ideal $\mathfrak{a}$, the ideal $\mathfrak{b}\bar{\mathfrak{c}}$ is equivalent to $\mathfrak{a}\bar{\mathfrak{a}} = (\mathrm{N}(\mathfrak{a}))$, thus it is a principal ideal. Let us denote by α a generator of the ideal $\mathfrak{b}\bar{\mathfrak{c}}$.

As the endomorphism $\iota(\alpha)$ has kernel $E[\mathfrak{b}\bar{\mathfrak{c}}]$, by Proposition 1.3.12, there exist two isomorphisms $\varepsilon_0, \varepsilon_1$ of E such that

$$\iota(\alpha) = \varepsilon_0 \circ \varphi'_{\mathfrak{b}} \circ \varphi_{\bar{\mathfrak{c}}} = \varepsilon_1 \circ \varphi'_{\bar{\mathfrak{c}}} \circ \varphi_{\mathfrak{b}}.$$

In particular, post-composing $\varphi'_{\bar{\mathfrak{c}}}$ by ε_1 and $\varphi'_{\mathfrak{b}}$ by ε_0 makes the above diagram commute explicitly. Applying Kani's Lemma on the obtained diagram yields the isogeny $F : E \times E \to E^{\mathfrak{b}} \times E^{\bar{\mathfrak{c}}}$ given by the matrix

$$\begin{pmatrix} \varphi_{\mathfrak{b}} & \widetilde{\varepsilon_1 \circ \varphi'_{\bar{\mathfrak{c}}}} \\ -\varphi_{\bar{\mathfrak{c}}} & \widetilde{\varepsilon_0 \circ \varphi'_{\mathfrak{b}}} \end{pmatrix}.$$

Then, $\varphi_{\mathfrak{b}}$ and $\varphi_{\bar{\mathfrak{c}}}$ can be recovered from the evaluation of F on $E \times \{0\}$. Moreover, since $\mathrm{N}(\mathfrak{b})$ and $\mathrm{N}(\mathfrak{c})$ are coprime, the kernel of F is

$$\ker F = \{(N(\mathfrak{b})(P), \iota(\alpha)(P)) \text{ such that } P \in E[N]\}.$$

By hypothesis, $E[N]$ is an accessible subgroup, and ι is efficiently represented, so the kernel of F is computable in polynomial time.

Thanks to the generalisation of Vélu's formulas, Lemma 1.6.7, one can compute an isogeny G of E^2 with kernel equal to $\ker F$ efficiently. By Lemma 1.6.6, this isogeny G is equal to F up to post-composition with an isomorphism. Hence, we actually recover $\varphi_{\mathfrak{b}}$ and $\varphi_{\bar{\mathfrak{c}}}$ only up to isomorphisms. This is not an issue since we are working with isomorphism classes of $\mathfrak{O}$-oriented elliptic curves. In particular, this is enough to obtain the isomorphism class of $E^{\mathfrak{a}}$. [2]

To completely compute the action of the ideal class $[\mathfrak{b}]$ on (E, ι), one needs to compute the orientation over $E^{\mathfrak{b}}$ induced by $\varphi_{\mathfrak{b}}$. We recall that the definition of the action of $[\mathfrak{b}]$ on (E, ι) is

$$\mathfrak{b} \star (E, \iota) = (E^{\mathfrak{b}}, \varphi_{\mathfrak{b}} \circ \iota \circ \hat{\varphi}_{\mathfrak{b}} / \mathrm{N}(\mathfrak{b})).$$

Thanks to the `IsogenyDivision` algorithm, see Theorem 2.2.1, an efficient representation of the induced orientation can by computed in polynomial time from the representation of $\varphi_{\mathfrak{b}}$ and ι. However, this division algorithm requires to process more higher dimensional isogenies computations, increasing heavily the global complexity. Unfortunately, most of the time, this computation is mandatory if one wants to compute successive group actions.

[2] In [PR23], the authors show that it is actually possible to recover exactly $\varphi_{\mathfrak{a}}$.

Chapter 3

Foundations of isogeny-based cryptography

In Section 3.2, we present the major contribution of the chapter: the unconditional *equivalence between hard problems, together with the inclusion of the* HomModule *problem in this list of equivalent problems. The content comes from the preprint article currently under review:*

[HW25b] Arthur Herlédan Le Merdy and Benjamin Wesolowski. Unconditional foundations for supersingular isogeny-based cryptography. Cryptology ePrint Archive, Paper 2025/271, 2025.

In Section 3.3, we present an original extension of this article: a proof that these problems are also equivalent to "compact" variants, which require small solutions.

We conclude with Section 3.4, where we demonstrate that the existence of a hard instance of Isogeny *implies the average-case hardness of all the other problems. This result comes from the same paper [HW25b].*

The security of isogeny-based cryptography does not rely on a single hard problem, but rather on a variety of problems. Historically, the hard problems underlying the security of the first isogeny-based schemes were the Isogeny problem, which asks for an isogeny between two elliptic curves, and its variant, the ℓ-IsogenyPath problem, in which the isogeny must be a chain of ℓ-isogenies. The security of the `CRS` key exchange scheme [Cou06, RS06] relies on the former, in its specialisation to *ordinary* elliptic curves, while the security of the `CGL` hash function [CLG09] is related to the latter. Together they constitute the earliest isogeny-based constructions.

In [CLG09], the authors also consider the computation of endomorphisms as a hard problem. This led to the introduction of two different problems in the subsequent literature: the EndRing problem, where one needs to find a basis of the endomorphism ring, and the OneEnd problem, where one needs to find a single non-trivial endomorphism.

The EndRing problem itself comes in another flavour, known as the MaxOrder problem, induced by the Deuring correspondence. In this variant, one is looking only for the abstract structure of the endomorphism ring.

Most isogeny-based cryptography relies on one of these problems; for instance [CLM$^+$18, DKL$^+$20, CLG09]. However, the existence of such a diverse set of fundamental problems makes it challenging to establish solid foundations for isogeny-based cryptography. The consequences of breaking one of these problems for the overall security of the field might be unclear. This has motivated strong efforts to explore the connections between the different hard problems. Fortunately, cryptographers have proven their equivalence, first under heuristic assumptions [EHL$^+$18] and later under the generalised Riemann hypothesis only [Wes22b]. As a result, the generalised Riemann hypothesis is currently the last assumption required to unify the foundation of isogeny-based cryptography.

More recently, the unconditional reduction from EndRing to Isogeny, proven in [PW24], and the introduction of higher dimensional isogenies, following SIDH attacks [CD23, MMP$^+$23, Rob23a], have opened the possibility of proving the unconditional equivalence of all the core problems.

This chapter begins with Section 3.1, where we describe the previous state of the art regarding the foundations of isogeny-based cryptography. First, we present the different hard problems on which the security of isogeny-based schemes relies, sorting them into two families. One concerns problems asking for a single isogeny or endomorphism; the other embeds the problems asking for a structured set of isogenies or endomorphisms. Throughout this section, we motivate their study by providing concrete examples of reductions to scheme security. Finally, we summarise the state-of-the-art reductions that have been rigorously proven.

In Section 3.2, we prove their unconditional equivalence (Theorem 3.2.1). In addition, we extend the global picture with the introduction of the HomModule problem — which asks for a basis of the homomorphism module between two given elliptic curves — as a new fundamental problem. This original work takes advantage of the recent higher dimensional machinery, see Chapter 2, via its various applications such as the `IsogenyInterpolation`, `IsogenyDivision` and `IdealToIsogeny` algorithms. This demonstrates that the consequences of the `SIDH` attacks do not only extend to building or improving schemes [BM25, DLRW24], but also to reinforce the foundations of isogeny-based cryptography.

Then in Section 3.3, we prove that every hard problem is equivalent to a variant problem where the solutions are required to have length polynomial in $\log p$. While this has already been studied heuristically in previous literature for some of the problems — for instance, in [EHL+18], regarding EndRing and MaxOrder— proving it rigorously, especially with our more general definition of MaxOrder, is not trivial.

Finally, in Section 3.4, we explore worst-case to average-case reductions between the different hard problems. The average cases are defined to suit the distributions of the problem instances encountered in the isogeny-based cryptosystems. The interest of such reductions lies in the following: they demonstrate that if there exists a single hard instance for a given problem, then solving it on average is hard. Our main result proves that the existence, for example, of a single hard EndRing instance implies the hardness of every fundamental problem on average.

3.1 The fundamental problems

The Isogeny problem(s)

The problem that gives its name to isogeny-based cryptography asks for an isogeny between two given elliptic curves. It was introduced as the computational problem behind the security of the `CRS` key exchange [Cou06, RS06]. While the early isogeny-based schemes used ordinary elliptic curves, now most of the schemes rely on supersingular elliptic curves for the sake of efficiency. We refer to Section 1.4 for a detailed example of the `CRS` framework turned into the practical `CSIDH` algorithm by leveraging supersingular elliptic curves. Hence, we consider this presumably hard problem and all the following ones in the supersingular context.

Problem 3.1.1 (Isogeny). *Given two supersingular elliptic curves E and E' defined over $\mathbb{F}_{p^2}$, compute an isogeny $\varphi : E \to E'$ between them.*

An intrinsic challenge in the study of the Isogeny problem is the representation of isogenies: what does it mean to answer with an isogeny? For us, it means providing an efficient representation of the isogeny as defined in Definition 1.3.17. Thanks to the development of higher dimensional isogenies, in particular the `IsogenyInterpolation` algorithm, a wide variety of isogenies can now be efficiently represented; see Chapter 2.

Prior to the SIDH attacks, most schemes relied on chains of isogenies with small degree, typically ℓ-isogenies for a small prime ℓ, to ensure efficient representations.

One might also want to consider ℓ-isogenies to exploit the rapid mixing property of ℓ-isogeny graphs; see Section 1.5. We define the specialisation of the Isogeny problem in this particular setting as follows.

Problem 3.1.2 (ℓ-IsogenyPath). *Given two supersingular elliptic curves E and E' defined over $\mathbb{F}_{p^2}$ and a prime $\ell \neq p$, compute a chain of ℓ-isogenies between E and E'.*

The state-of-the-art algorithms to solve this problem are exponential in $\log p$.

Proposition 3.1.3. *[PW24, Proposition 8.7] One can compute a solution to the ℓ-IsogenyPath problem in expected time $(\ell, \log p)^{O(1)}\sqrt{p}$ such that the solution is a path of length $O(\log p)$.*

We cite this particular reference as it provides a compact proposition and proof. Nevertheless, it has been studied in several prior articles; see for instance [DG16] for more details. Thanks to Grover's algorithm [Gro96], one can achieve a quadratic speed up using quantum computers.

Proposition 3.1.4. *[BJS14] One can compute a solution to the* Isogeny *problem in expected time $(\ell, \log p)^{O(1)}\sqrt[4]{p}$ such that the solution is a path of length $O(\log p)$.*

In 2006, Charles, Goren and Lauter introduced a hash function [CLG09], now referred to as the CGL hash function, taking advantage of supersingular elliptic curves through the properties of ℓ-isogeny graphs. The security of a cryptographic hash function is primarily characterised by its preimage resistance and collision resistance. The former corresponds to finding a bit string with a given hash value; the latter to finding two bit strings with the same hash. The preimage resistance of CGL is proven to be at least as hard as the ℓ-IsogenyPath problem, [CLG09, Theorem 2]. Its collision resistance is at least as hard as the OneEnd problem [CLG09, Theorem 1], which consists of finding a non-trivial endomorphism, i.e. an endomorphism which is not a multiplication by an integer.

Problem 3.1.5 (OneEnd). *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, compute an endomorphism in $\mathrm{End}(E) \setminus \mathbb{Z}$.*

This problem is also central to the security of the SQIsign digital signature [DKL$^+$20, DLRW24, BDD$^+$24]. In particular, the soundness of SQIsign2D-West [BDD$^+$24] is equivalent to the OneEnd problem. The soundness property ensures that a dishonest prover cannot convince a verifier.

Remark 3.1.6. *The* OneEnd *problem, as defined above, was introduced in [PW24] around fifteen years after the CGL hash function. In [CLG09], the authors consider a more specific problem of finding an endomorphism of degree ℓ^{2n}. We shall see in the next sections that the* OneEnd *problem, as defined here, plays a central role in the network of reductions.*

The Endomorphism Ring problem(s)

In isogeny-based cryptography, besides the problems consisting in finding isogenies, there are problems asking to find the structure of isogeny sets, which gives central information about elliptic curves. First, we consider the problem of computing the endomorphism ring of a given elliptic curve.

Problem 3.1.7 (EndRing). *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find four endomorphisms generating $\mathrm{End}(E)$ as a $\mathbb{Z}$-module.*

This problem is also strongly connected to the security of isogeny-based schemes. For instance, it is equivalent to the problem of recovering the private key in the `SQIsign2D-West` digital signature scheme.

Once again, we restrict ourselves to the case of supersingular elliptic curves. It is worth noting that, thanks to higher dimensional isogenies, this problem is no longer a hard problem for ordinary elliptic curves. It can be solved in quantum polynomial time [Rob22b].

The first computational analysis of EndRing was conducted in the thesis of Kohel [Koh96] outside of any cryptographic purposes. It resulted in a probabilistic algorithm to find a full-rank subring of the endomorphism ring in time $\tilde{O}(p)$ [Koh96, Theorem 75]. This early work has already outlined the strong connection between the Isogeny problem family and the EndRing problem, as ℓ-isogeny graphs play a crucial role in the resolution of EndRing.

A more explicit connection emerges when one takes the quaternion point of view offered by the Deuring correspondence. This correspondence has been summarised in Section 1.3; we recall some useful parts here. First, there exists a quaternion algebra B isomorphic to the endomorphism algebra $\mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$. In particular, there is a maximal order in B which is isomorphic to the endomorphism ring of E. Computing such a maximal order is called the Maximal Order problem (MaxOrder). This problem corresponds to computing the abstract structure of the endomorphism ring.

Problem 3.1.8 (MaxOrder)**.** *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find a quaternion algebra $B = (\frac{-a,-b}{\mathbb{Q}}) \simeq \mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$ and four quaternions generating a maximal order in B isomorphic to $\mathrm{End}(E)$ as a $\mathbb{Z}$-module.*

Adopting this abstract perspective is powerful because some hard problems in the isogeny world become easy in the quaternion world. In particular, while finding an isogeny is (hopefully!) difficult, computing a connecting ideal is an easy task. We recall that isogenies between elliptic curves correspond to connecting ideals between maximal orders via the Deuring correspondence. Moreover, an ideal can efficiently be translated into an isogeny. Thus, solving the Isogeny problem can be done in polynomial time from the knowledge of suitable maximal orders — in the next sections, we give more details about this reduction and the evolution of the assumptions that have been used to prove it. The main point here being that the difficulty of solving the isogeny problem via “quaternionic” tools relies mostly in the computation of maximal orders isomorphic to the endomorphism rings of the given elliptic curves.

The MaxOrder problem, as stated above, differs from its definition in prior work such as [EHL$^+$18, Wes22b]. In the previous literature, the maximal orders were always computed in the quaternion algebra given by the model $(\frac{-p,-q_p}{\mathbb{Q}}) \simeq B_{p,\infty}$ as defined in Proposition 1.3.39. Hence, there was no need to compute a model $(\frac{-a,-b}{\mathbb{Q}})$ for a quaternion algebra isomorphic to the endomorphism algebra. However, without **GRH**, it is no longer possible to consider this model as “canonical”, whence the more general MaxOrder problem defined here. We discuss this point in more depth in Section 3.2, where we prove that the more general problem is actually equivalent to the classical one, which we call MaxOrder$_{\mathcal{Q}}$, when a model for $B_{p,\infty}$ as in [EHL$^+$18, Wes22b] is provided.

Problem 3.1.9 (MaxOrder$_{\mathcal{Q}}$)**.** *Let $\mathcal{Q}$ be an algorithm which for any prime number p, outputs a prime $q = \mathcal{Q}(p)$ such that $B_{p,\infty} \simeq (\frac{-p,-q}{\mathbb{Q}})$. The problem MaxOrder$_{\mathcal{Q}}$ is the following. Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find four quaternions in $(\frac{-p,-\mathcal{Q}(p)}{\mathbb{Q}})$ generating a maximal order isomorphic to $\mathrm{End}(E)$.*

We combine the EndRing problem — which asks for actual endomorphisms generating $\operatorname{End}(E)$ — and the MaxOrder problem — which asks for the "quaternionic" structure of $\operatorname{End}(E)$ — in the following problem.

Problem 3.1.10 (MOER). *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find four endomorphisms $(\alpha_i)_{i=1}^4$ generating* $\operatorname{End}(E)$ *as a $\mathbb{Z}$-module, a quaternion algebra $B = (\frac{-a,-b}{\mathbb{Q}}) \simeq \operatorname{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$, and four quaternions $(\beta_i)_{i=1}^4$ in B such that*

$$\operatorname{End}(E) \otimes \mathbb{Q} \longrightarrow B : \alpha_i \longmapsto \beta_i$$

is an isomorphism.

Solving this problem ensures that one can exploit the structure of quaternions to perform efficient computations before coming back to the isogeny world.

The MOER problem is comparable to the EndRing problem as defined in [Wes22a]. In this article, EndRing asks for an ε-basis of the endomorphism ring, meaning that an $\operatorname{End}(E)$ basis comes together with an explicit isomorphism ε between $\operatorname{End}(E)\otimes\mathbb{Q} \to B_{p,\infty}$. Hence, this is exactly MOER restricted to the case $B = B_{p,\infty}$. The choice of defining MOER as a problem on its own follows our objective of studying relations between hard problems — it greatly simplifies the map of proofs.

Finally, we introduce a new problem which asks for the computation of the $\mathbb{Z}$-module of isogenies between two supersingular elliptic curves.

Problem 3.1.11 (HomModule). *Given two supersingular elliptic curves E and E' defined over $\mathbb{F}_{p^2}$, find four isogenies generating* $\operatorname{Hom}(E, E')$ *as a $\mathbb{Z}$-module.*

This problem plays the same role for the Isogeny problem as EndRing for the OneEnd problem. However, it is not only introduced for the sake of completeness, but also because it appears naturally in isogeny-based cryptography. For example, it corresponds to the space of possible answers in `SQIsign` during an identification.

They are equivalent

The connections between the ℓ-IsogenyPath, EndRing, and MaxOrder problems have been well studied in the literature. They turned out to be equivalent, as this was first proven under heuristics in [EHL+18] then assuming only the generalised Riemann hypothesis in [Wes22b].

A pivotal result for the field is the introduction of the OneEnd problem as a hard problem equivalent to EndRing [PW24]. In the same work, the authors also demonstrate that OneEnd reduces to Isogeny. As both results are proven unconditionally, this provides the first rigorous reduction from EndRing to Isogeny that is free of any assumptions.

Together with [Wes22b], this article constitutes the core of the literature on rigorous reductions between hard problems. We summarise the global picture in Figure 3.1.

In the next section, we provide reductions without assuming the generalised Riemann hypothesis, resulting in unconditional equivalence between (almost) all the hard problems of isogeny-based cryptography. Let us give some insight into why **GRH** is necessary in Figure 3.1. These reductions require this assumption mainly to

(i) Compute a model for $B_{p,\infty}$,

(ii) Find smooth equivalent ideals,

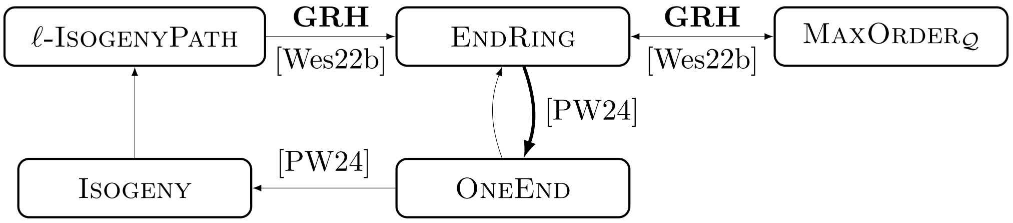


Figure 3.1: Former state of the art of (conditional) reductions between foundational problems of isogeny-based cryptography. All arrows are classical polynomial time reductions. Thin arrows have a $O(1)$ query-complexity, and the thick arrow has a polylog(p) query-complexity. Reductions with no reference are trivial, and all others are proved in the associated reference. The **GRH** label signifies that a reduction assumes the generalised Riemann hypothesis. The table comes from [HW25a].

(iii) Represent primes by quadratic forms .

Point (i) refers to finding a prime q_p such that the quaternion algebra $(\frac{-p,-q_p}{\mathbb{Q}})$ is isomorphic to $B_{p,\infty}$. By Proposition 1.3.39, this step is trivial when $p \not\equiv 1 \mod 8$ as q_p is simply equal to 1 or 2. Otherwise, **GRH** is necessary to guarantee that q_p is polynomial in $\log p$, so an exhaustive search algorithm finds q_p efficiently.

Without **GRH**, to the best of our knowledge, the best estimate on the size of q_p comes from [LO77]. It only ensures that q_p is polynomial in p. Thus, the exhaustive search is no longer feasible Unfortunately, there is currently no other way to find q_p. Note that without the knowledge of a small q_p, one does not have access to an elliptic curve defined over $\mathbb{F}_{p^2}$ such that its endomorphism ring is known. This is one of the main difficulties of working without **GRH**.

Item (ii) concerns the crucial `IdealToIsogeny` step in the reduction from ℓ-IsogenyPath to MaxOrder$_{\mathcal{Q}}$ [Wes22b, Theorem 7.4]. Before the `SIDH` attacks, one could only perform such translations when the ideal had a powersmooth norm [GPS17, Lemma 5]. Hence, computing an equivalent ideal with a suitable norm was crucial. In [KLPT14] a practical heuristic algorithm, now known as the `KLPT` algorithm, is presented to solve this problem. In [Wes22b], a more theoretical algorithm is proven to have polynomial time complexity under **GRH** only.

The recent unconditional `IdealToIsogeny` algorithm based on `CLAPOTI` [PR23], see Proposition 2.3.3, allows us to bypass this step by computing isogenies corresponding to any ideals, regardless of their norm. We recall that applying this procedure requires knowing the endomorphism ring of at least a curve defined over $\mathbb{F}_{p^2}$; thus, in some context, we fall back to the difficulty discussed in item (i).

Point item (iii) is a key step in the reduction from MaxOrder$_{\mathcal{Q}}$ to EndRing [Wes22b, Theorem 8.3]. It occurs when searching for a maximal order in the specific model of quaternion algebra $(\frac{-p,-q_p}{\mathbb{Q}})$. We shall not give details here since, by considering the more general MaxOrder problem, we completely address this difficulty.

In conclusion, by introducing a distinction between MaxOrder and MaxOrder$_{\mathcal{Q}}$ and using higher dimensional results, we solve most of the issues encountered. The main remaining obstacle is then to work without access to a special curve with known endomorphism ring. Nevertheless, there is still one black spot.

- How to reduce ℓ-IsogenyPath to another problem *unconditionally*?

In fact, searching for isogenies of degree a power of a prime, brings us back to similar difficulties as in item (ii) — finding ideals of a certain form. While the reduction in [Wes22b] from ℓ-IsogenyPath to MaxOrder$_{\mathcal{Q}}$ can be done with GRH — thanks to the rigorous `KLPT`-algorithm proven in [Wes22b] — there is, to the best of our knowledge, no unconditional way to do it.

There is at least one attempt that was made public. In [Mam24], the author tries to reduce ℓ-IsogenyPath to EndRing assuming only access to a factorisation oracle. However, according to the author himself, there are currently some issues in the proof. If these were fixed, this would imply unconditional quantum equivalence between ℓ-IsogenyPath and EndRing; and thus between all the problems presented in this chapter, thanks to the results of Section 3.2.

3.2 They are *unconditionally* equivalent

The content of this section comes from [HW25a]. Our main goal is to prove the unconditional reductions of Figure 3.2, which leads to Theorem 3.2.1 — the unconditional equivalence of all the hard problems presented in Section 3.1, except the ℓ-IsogenyPath problem.

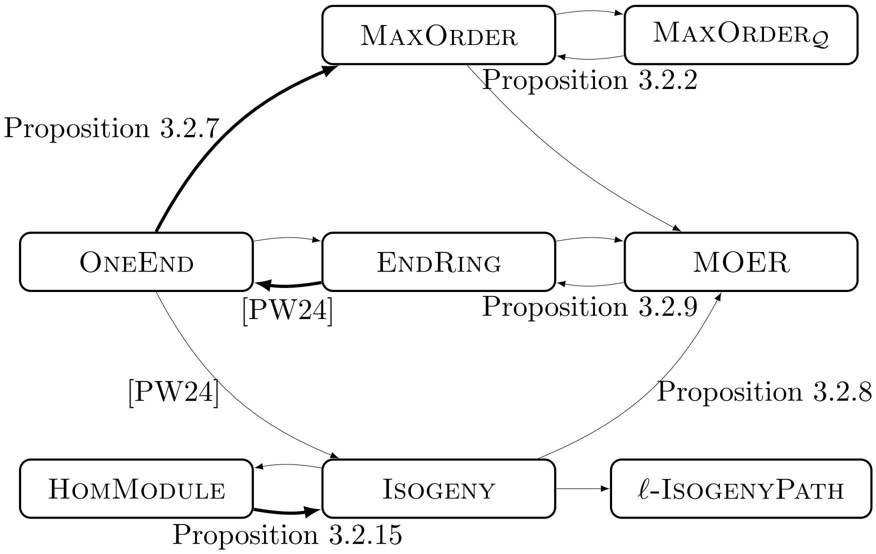


Figure 3.2: Summary of the relations between fundamental isogeny-based problems. All arrows are unconditional classical polynomial time reductions. Thin arrows have a $O(1)$ query-complexity, and thick arrows have a polylog(p) query-complexity. Reductions with no reference are trivial, and all others are proved in the associated reference. Reductions involving MaxOrder$_{\mathcal{Q}}$ require oracle access to $\mathcal{Q}$.

Theorem 3.2.1. *The problems* Isogeny*,* EndRing*,* OneEnd*,* MOER*,* MaxOrder*,* MaxOrder$_{\mathcal{Q}}$ *and* HomModule *are all equivalent under probabilistic polynomial time reductions. Reductions involving* MaxOrder$_{\mathcal{Q}}$ *require oracle access to* $\mathcal{Q}$.

Map of the proof of Theorem 3.2.1

The strategy consists in proving the computational reductions exhibited in Figure 3.2. Each arrow represents a computational reduction, and comes with a pointer to the proof.

Note that our unconditional reductions are substantially different from the existing conditional reductions. To eliminate any reliance on GRH, we avoid all arguments that rely on the "good" distribution of numbers represented by quadratic forms. This forbids us from using the powerful tools of KLPT-type algorithms [KLPT14]. In particular, we need to construct different "paths" in the network of reductions, and develop new types of arguments. We prove the reductions in the following order.

- The novel distinction between the two computational problems MaxOrder and $\text{MaxOrder}_{\mathcal{Q}}$ is discussed in Section 3.2. Their equivalence, proved in Proposition 3.2.2, hinges on a recent result [CKMZ22, Proposition 4.1] to compute isomorphisms between quaternion algebras, when some maximal orders are known in each.

- The reduction from OneEnd to MaxOrder is the object of Section 3.2. Navigating between a problem which deals with endomorphisms (like OneEnd) and another which deals with purely quaternionic data (like MaxOrder) typically requires to connect instances to some special elliptic curve E_0 for which both $\text{End}(E_0)$ and its embedding in the quaternions are already known. This curve E_0 provides an "endomorphism/quaternion" dictionary. Without GRH, there is no guarantee that a special curve E_0 can be found. We thus need to develop a new strategy. To reduce OneEnd (say on some input E) to MaxOrder, we solve MaxOrder on E and on a few "close neighbours" of E. Doing so, we construct a "local" correspondence between neighbours of E and quaternionic orders, and we prove that from enough such "local" information, we can reconstruct a full "endomorphism/quaternion" dictionary.

- The reduction from Isogeny to MOER is the object of Proposition 3.2.8. This reduction hinges on recent advances in isogeny-based cryptography facilitating the conversion of ideals in quaternionic orders into the corresponding isogenies [PR23].

- The reduction from MOER to EndRing is the object of Proposition 3.2.9. Of all the reductions, this one resembles the most closely an existing reduction: the reduction from MaxOrder to EndRing in [EHL$^+$18, Wes22b]. However, these former reductions required GRH to provably avoid hard factorisations. Instead, we show that no factorisation is needed if we are free to choose our own model $(\frac{a,b}{\mathbb{Q}})$ for the quaternion algebra. The parameters a and b are possibly hard to factor, but it does not matter: the result [CKMZ22, Proposition 4.1] allows one to convert the solution to more standard models without factoring.

- Finally, the HomModule problem is the object of Section 3.2, where we prove that it reduces to Isogeny. Given two curves E_1 and E_2, the strategy is the following. First, we exploit the reduction from EndRing to Isogeny proved in [PW24] to compute bases of $\text{End}(E_1)$ and $\text{End}(E_2)$. Then, we solve Isogeny again to find some $\varphi : E_1 \to E_2$. Through algebraic arguments, we prove that one can extract a basis of $\text{Hom}(E_1, E_2)$ from the data of $\text{End}(E_1)$, $\text{End}(E_2)$, and φ.

The Maximal Order problem

Let us discuss the MaxOrder problem, and a subtlety in its definition when one does not assume GRH. The classical definition makes the implicit assumption that a reference quaternion algebra $B_{p,\infty}$ is provided. However, there is no "canonical" model of $B_{p,\infty}$. When $p \equiv 3 \bmod 4$ (respectively $p \equiv 5 \bmod 8$), one can argue that the algebra $(\frac{-p,-1}{\mathbb{Q}})$ (respectively $(\frac{-p,-2}{\mathbb{Q}})$) is a natural model for $B_{p,\infty}$. However, when $p \equiv 1 \bmod 8$, there is no uniform value of q for which $B_{p,\infty} \simeq (\frac{-p,-q}{\mathbb{Q}})$. In order to fix an algebra for each p, previous works fix a procedure $\mathcal{Q}$ such that on input p, the output $\mathcal{Q}(p)$ is a prime satisfying $B_{p,\infty} \simeq (\frac{-p,-\mathcal{Q}(p)}{\mathbb{Q}})$. This $\mathcal{Q}(p)$ is typically set to be the smallest prime number with the requested property. While a convenient choice, it is somewhat arbitrary. Furthermore, without GRH, there is no guarantee for this value to be small, nor easy to find.

As this model $B_{p,\infty} = (\frac{-a,-b}{\mathbb{Q}})$ *might* be hard to compute (without GRH, when $p \equiv 1 \bmod 8$), the original definition of MaxOrder becomes ambiguous: are a and b provided, or are they to be computed? We settle for a definition of MaxOrder where a and b are left to be found. Let us show that the impact of this choice is minimal: it is equivalent to the variant MaxOrder$_{\mathcal{Q}}$ where the algebra is imposed to be of the classical form $(\frac{-p,-\mathcal{Q}(p)}{\mathbb{Q}})$, for any procedure $\mathcal{Q}$ which returns a suitable prime. The key is Proposition 1.3.38, which allows one to translate solutions accross different models of $B_{p,\infty}$, so the choice of a particular model does not matter.

Proposition 3.2.2 (MaxOrder$_{\mathcal{Q}}$ is equivalent to MaxOrder)**.** *Given oracle access to $\mathcal{Q}$, the two problems* MaxOrder *and* MaxOrder$_{\mathcal{Q}}$ *are equivalent under probabilistic polynomial time reductions. The reductions make a single query to each oracle.*

Proof. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve. We first prove that MaxOrder reduces to MaxOrder$_{\mathcal{Q}}$. Indeed, if $\mathcal{O} \subset (\frac{-p,-\mathcal{Q}(p)}{\mathbb{Q}})$ is a solution of MaxOrder$_{\mathcal{Q}}$, then $(p, \mathcal{Q}(p), \mathcal{O})$ is a solution of MaxOrder.

We now prove that MaxOrder$_{\mathcal{Q}}$ reduces to MaxOrder. Let $\mathcal{O} \subseteq (\frac{-a,-b}{\mathbb{Q}})$ be a solution of MaxOrder, and let $q = \mathcal{Q}(p)$. Let Λ_0 be the (non-maximal) order spanned by the canonical basis $(1, i, j, k)$ of $(\frac{-p,-q}{\mathbb{Q}})$. Thanks to the orthogonality of this basis and since the discriminant of any order $\mathcal{O} = (\alpha_1, \dots, \alpha_4)$ is given by $\mathrm{disc}(\mathcal{O}) = \sqrt{|\det((\langle \alpha_i, \alpha_j \rangle)_{i,j})|}$, the discriminant of Λ_0 is pq. The factorisation of $\mathrm{disc}(\Lambda_0)$ being known, one can construct a maximal order $\mathcal{O}_0 \supseteq \Lambda_0$ in polynomial time with [Voi13, Theorem 7.14].

From Proposition 1.3.38, one can compute an order $\mathcal{O}'$ in $(\frac{-p,-q}{\mathbb{Q}})$ isomorphic to $\mathcal{O}$. Then, $\mathcal{O}'$ is a solution of MaxOrder$_{\mathcal{Q}}$. □

Finding endomorphisms from quaternions

We now develop an unconditional reduction from the OneEnd problem to the MaxOrder problem. The main difficulty is that without GRH, there is no general way to compute a "special" elliptic curve E_0 for which both $\mathrm{End}(E_0)$ and its embedding in the quaternions are already known. Such a curve provides an "endomorphism/quaternion" dictionary, and previous literature on MaxOrder made critical use of that fact. Without such an E_0, we need to develop a completely different strategy. To reduce OneEnd (say on some input E) to MaxOrder, we solve the MaxOrder problem on E, giving an order $\mathcal{O}$, but also on a few of its "neighbours". We thereby construct a "local" correspondence: a canonical bijection between ℓ-isogenies from E and ideals of norm ℓ in the order $\mathcal{O}$. This is done in Algorithm 2. We then prove that this information can be converted into isomorphisms $\mathrm{End}(E[\ell]) \simeq \mathcal{O}/\ell\mathcal{O}$ which all descend from the same implicit isomorphism $\mathrm{End}(E) \simeq \mathcal{O}$, via Algorithm 3. Finally, from this local data, we reconstruct a full "endomorphism/quaternion" dictionary $\mathrm{End}(E) \simeq \mathcal{O}$ in Algorithm 4.

At several steps of this process, one might fail to construct the dictionary (for instance when the isomorphism $\mathrm{End}(E) \simeq \mathcal{O}$ is not unique). In such a scenario, a non-scalar endomorphism of E is revealed, and we have solved OneEnd anyway. If no such failure occurs, we successfully obtain a dictionary, which in turn reveals (all!) non-scalar endomorphisms of E.

Algorithm 2 Computing a bijection between ℓ-isogenies and ideals, given a MaxOrder oracle

Input: A supersingular elliptic curve $E/\mathbb{F}_{p^2}$, a prime $\ell \neq p$, a list $(G_i)_{i=0}^{\ell}$ of all subgroups of order ℓ in E, an algebra $B = (\frac{-a,-b}{\mathbb{Q}}) \simeq B_{p,\infty}$, a maximal order $\mathcal{O} \simeq \mathrm{End}(E)$ in B, and access to an oracle for MaxOrder.

Output: Either an endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$, or the list $(I_i)_{i=0}^{\ell}$ of left $\mathcal{O}$-ideals such that $\mathcal{O}_R(I_i) \simeq \mathrm{End}(E/G_j)$ if and only if $i = j$.

```
1: Compute $\varphi_i : E \to E/G_i$ using Vélu's formulas for $i = 0, \dots, \ell$
2: if $E/G_i \simeq E/G_j$ for some $i \neq j$ then
3:     $\gamma \leftarrow$ an isomorphism between $E/G_i$ and $E/G_j$
4:     return $\hat{\varphi}_j \circ \gamma \circ \varphi_i \in \mathrm{End}(E) \setminus \mathbb{Z}$
5: if $E/G_i \simeq (E/G_j)^{(p)}$ for some $i \neq j$ then
6:     $\gamma \leftarrow$ an isomorphism between $E/G_i$ and $(E/G_j)^{(p)}$
7:     $\phi_p \leftarrow$ the Frobenius isogeny $\phi_p : E/G_j \to (E/G_j)^{(p)}$
8:     return $\hat{\varphi}_j \circ \phi_p \circ \gamma \circ \varphi_i \in \mathrm{End}(E) \setminus \mathbb{Z}$
9: for $i = 0, \dots, \ell$ do
10:    $(B_i, \tilde{\mathcal{O}}_i) \leftarrow$ an algebra $B_i \simeq B_{p,\infty}$ and a maximal order $\tilde{\mathcal{O}}_i \subset B_i$ such that $\tilde{\mathcal{O}}_i \simeq$
       $\mathrm{End}(E/G_i)$.                    ▷ Using the oracle for MaxOrder on $E/G_i$
11:    $\mathcal{O}_i \leftarrow$ an order in $B$ isomorphic to $\mathrm{End}(E/G_i)$ ▷ Proposition 1.3.38 on $(B, \mathcal{O})$ and
       $(B_i, \tilde{\mathcal{O}}_i)$.
12:    $J_i \leftarrow I(\mathcal{O}, \mathcal{O}_i)$ the connecting ideal              ▷ [KV10, Algorithm 3.5]
13:    $\langle \alpha_1, \dots, \alpha_4 \rangle \leftarrow$ a Minkowski-reduced basis of $J_i$                     ▷ [NS09]
14:    if $\mathrm{Nrd}(\alpha_2) \leq \ell \, \mathrm{Nrd}(J_i)$ then
15:        $\alpha \leftarrow$ a non-scalar endomorphism of $E$ of degree at most $\ell^2$ ▷ For instance, by
       exhaustive enumeration of isogenies of degree at most $\ell^2$ from $E$.
16:        return $\alpha$
17:    $I_i \leftarrow J_i \overline{\alpha}_1 / \mathrm{Nrd}(J_i)$        ▷ The unique left $\mathcal{O}$-ideal of norm $\ell$ equivalent to $J_i$.
18: return $(B, \mathcal{O}, (I_i)_{i=0}^{\ell})$
```

Lemma 3.2.3. *Algorithm 2 is correct and runs in time polynomial in the length of the input, in ℓ, and in the length of the* MaxOrder *oracle outputs.*

Proof. The claim that the running time is polynomial follows from the references provided in the comments of Algorithm 2.

Let us prove that the algorithm is correct.

First, the endomorphism $\alpha = \hat{\varphi}_j \circ \gamma \circ \varphi_i$ returned at Line 4 is not scalar. Indeed, suppose by contradiction that $\alpha \in \mathbb{Z}$. It is of degree ℓ^2, so $\alpha = [\ell]$. We thus have $\hat{\varphi}_j \circ \varphi_j = [\ell] = \hat{\varphi}_j \circ \gamma \circ \varphi_i$, hence $\varphi_j = \gamma \circ \varphi_i$, hence $\ker(\varphi_j) = \ker(\varphi_i)$, hence $G_i = G_j$, contradicting that $i \neq j$.

Second, the endomorphism returned at Line 8 is not scalar either. Indeed, it has degree $\ell^2 p$, which is not a square, so it cannot be scalar.

Note that after Line 8, the rings $\mathrm{End}(E/G_i)$ are pairwise non-isomorphic. Indeed, by [Voi21, Lemma 42.4.1], if $\mathrm{End}(E/G_i) \simeq \mathrm{End}(E/G_j)$, then E/G_i is isomorphic to either E/G_j or to its Galois conjugate $(E/G_j)^{(p)}$, in which case the algorithm has terminated before Line 8. In particular, the codomains of all ℓ-isogenies from E have pairwise distinct endomorphism rings, hence left ideals of norm ℓ in $\mathcal{O}$ are uniquely identified by the isomorphism class of their right-order.

At each iteration of the for-loop, we consider two cases.

- If $\operatorname{Nrd}(\alpha_2) \leq \ell \operatorname{Nrd}(J_i)$, by definition of Minkowski bases, we have that $\operatorname{Nrd}(\alpha_1) \leq \ell \operatorname{Nrd}(J_i)$. As $J_i \bar{J}_i = \operatorname{Nrd}(J_i)\mathcal{O}$, the element $\alpha_1\bar{\alpha}_2/\operatorname{Nrd}(J_i)$ is in $\mathcal{O}$. Furtheremore, $\alpha_1\bar{\alpha}_2/\operatorname{Nrd}(J_i)$ is not a scalar, otherwise α_1 and α_2 would be linearly dependent. Therefore $\mathcal{O}$ (hence also $\operatorname{End}(E)$) contains a non-scalar element of norm
$$\operatorname{Nrd}\left(\frac{\alpha_1\bar{\alpha}_2}{\operatorname{Nrd}(J_i)}\right) = \frac{\operatorname{Nrd}(\alpha_1)\operatorname{Nrd}(\alpha_2)}{\operatorname{Nrd}(J_i)^2} \leq \ell^2.$$
Then, a non-trivial endomorphism of degree at most ℓ^2 can be found in time polynomial in $\log p$ and ℓ by exhaustive search.

- Otherwise, $\operatorname{Nrd}(\alpha_2) > \ell \operatorname{Nrd}(J_i)$. Let us prove that in that case, the ideal $I_i = J_i\bar{\alpha}_1/\operatorname{Nrd}(J_i)$ is the unique left $\mathcal{O}$-ideal of norm ℓ with $\mathcal{O}_R(I_i) \simeq \operatorname{End}(E/G_i)$. Recall that by the Deuring correspondence, the lattice $\operatorname{Hom}(E, E/G_i)$ (for the quadratic form deg) is isomorphic to J_i (with quadratic form $q_{J_i} : \alpha \mapsto \operatorname{Nrd}(\alpha)/\operatorname{Nrd}(J_i)$). Therefore, there exists an element $\beta \in J_i$ such that $q_{J_i}(\beta) = \ell$. Since $q_{J_i}(\beta) < q_{J_i}(\alpha_2)$, the element β must be a multiple of α_1: there exists $m \in \mathbb{Z}$ such that $\beta = m\alpha_1$. Since
$$\ell = q_{J_i}(\beta) = q_{J_i}(m\alpha_1) = m^2 q_{J_i}(\alpha_1)$$
and ℓ is prime, we must have that $m = 1$ and $q_{J_i}(\alpha_1) = \ell$. This implies that $\operatorname{Nrd}(I_i) = \ell$.

  The unicity of I_i follows from the previously established fact that left ideals of norm ℓ in $\mathcal{O}$ are uniquely identified by the isomorphism class of their right-order.

The unicity of each I_i proves that if Line 18 is reached, we indeed have $\mathcal{O}_R(I_i) \simeq \operatorname{End}(E/G_j)$ if and only if $i = j$. □

Lemma 3.2.4. *Let $\rho : M_2(\mathbb{F}_\ell) \to M_2(\mathbb{F}_\ell)$ be a ring automorphism. If $\ker(m) = \ker(\rho(m))$ for all $m \in M_2(\mathbb{F}_\ell)$, then ρ is the identity.*

Proof. All automorphisms of $M_2(\mathbb{F}_\ell)$ are inner, so there exists $p \in \mathrm{GL}_2(\mathbb{F}_\ell)$ such that $\rho(m) = p^{-1}mp$ for all $m \in M_2(\mathbb{F}_\ell)$.

First, suppose there exists a line $L \subset \mathbb{F}_\ell^2$ such that $p(L) \neq L$. Then, there exists $m \in M_2(\mathbb{F}_\ell)$ such that $m(L) = p(L)$ and $m(p(L)) = \{0\}$. We obtain
$$\rho(m)(L) = (p^{-1}mp)(L) = p^{-1}(m(p(L)) = p^{-1}(\{0\}) = \{0\}.$$
By construction, m cannot be invertible or the zero matrix; consequently, both $\ker(\rho(m))$ and $\ker(m)$ have dimension 1. Therefore $L = \ker(\rho(m)) = \ker(m) = p(L)$, a contradiction. We deduce that for any line $L \subset \mathbb{F}_\ell^2$, we have $p(L) = L$. Since p fixes all lines in $\mathbb{F}_\ell^2$, all vectors of $\mathbb{F}_\ell^2$ are eigenvectors of p, so p is a scalar matrix. In particular, $\rho(m) = p^{-1}mp = m$. □

Corollary 3.2.5. *Consider rings $R \simeq R' \simeq M_2(\mathbb{F}_\ell)$. Let $\iota_1, \iota_2 : R \to R'$ be two ring isomorphisms. If $\iota_1(I) = \iota_2(I)$ for all left-ideals I in R, then $\iota_1 = \iota_2$.*

Proof. Fix two isomorphisms $g : R' \to M_2(\mathbb{F}_\ell)$ and $f : M_2(\mathbb{F}_\ell) \to R$, and define
$$\rho_i = g \circ \iota_i \circ f : M_2(\mathbb{F}_\ell) \to M_2(\mathbb{F}_\ell).$$
Let $\rho = \rho_2^{-1} \circ \rho_1$. Let us prove that ρ satisfies the condition of Lemma 3.2.4. Let $m \in M_2(\mathbb{F}_\ell)$. Let $J = M_2(\mathbb{F}_\ell)m = \{\tilde{m} \in M_2(\mathbb{F}_\ell) \mid \ker(m) \subseteq \ker(\tilde{m})\}$ be the left-ideal generated by m. Then, its image $f(J)$ is a left-ideal in R, hence $\iota_1(f(J)) = \iota_2(f(J))$, and
$$\rho(J) = f^{-1} \circ \iota_2^{-1} \circ \iota_1 \circ f(J) = f^{-1} \circ \iota_2^{-1} \circ \iota_2 \circ f(J) = J.$$

In particular, $\rho(m) \in \rho(J) = J$, so $\ker(m) \subseteq \ker(\rho(m))$. Since ρ is an isomorphism, the matrices m and $\rho(m)$ have the same rank, hence $\ker(m) = \ker(\rho(m))$.

We can thus apply Lemma 3.2.4, and deduce that ρ is the identity. In particular, we obtain $\rho_1 = \rho_2$, therefore $\iota_1 = \iota_2$. □

Algorithm 3 Computing an isomorphism between quaternions and endomorphisms modulo ℓ, given a MaxOrder oracle

Input: A supersingular elliptic curve $E/\mathbb{F}_{p^2}$, a prime ℓ, an algebra $B \simeq B_{p,\infty}$, a maximal order $\mathcal{O} \simeq \mathrm{End}(E)$ in B, and access to an oracle for MaxOrder.

Output: Either an endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$, or an isomorphism $\lambda : \mathcal{O}/\ell\mathcal{O} \to \mathrm{End}(E[\ell])$.

```
$(G_i)_{i=0}^{\ell} \leftarrow$ a list of all subgroups of order $\ell$ of the elliptic curve $E$.
Using the oracle access, run Algorithm 2 on the list $(G_i)_{i=0}^{\ell}$ to obtain either
    • a non trivial endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$,
    • or a list $(I_i)_{i=0}^{\ell}$ such that $I_i$ is the unique left $\mathcal{O}$-ideal of norm $\ell$ with $\mathcal{O}_R(I_i) \simeq \mathrm{End}(E/G_i)$.
if $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$ was found then
    return $\alpha$
$g \leftarrow$ an isomorphism from $\mathrm{End}(E[\ell])$ to $M_2(\mathbb{F}_\ell)$.
$f \leftarrow$ an isomorphism $\mathcal{O}/\ell\mathcal{O}$ to $M_2(\mathbb{F}_\ell)$.
$J_i \leftarrow \{\alpha \in \mathrm{End}(E[\ell]) | G_i \subset \ker\alpha\}$, for $i \in \{0, \dots, \ell\}$.
$h \leftarrow$ an automorphism from $M_2(\mathbb{F}_\ell)$ to $M_2(\mathbb{F}_\ell)$ such that $h(f(\tilde{I}_i)) = g(J_i)$ where $\tilde{I}_i$ is the reduction of $I_i$ modulo $\ell$.
$\lambda \leftarrow g^{-1} \circ h \circ f : \mathcal{O}/\ell\mathcal{O} \to \mathrm{End}(E[\ell])$.
return $\lambda$.
```

Lemma 3.2.6. *Algorithm 3 is correct and runs in time polynomial in the length of the input, in ℓ, and in the length of the output of the oracle for* MaxOrder.

Proof. Let $\iota : \mathcal{O} \xrightarrow{\sim} \mathrm{End}(E)$ be an isomorphism. Let us prove that the isomorphism λ computed by Algorithm 3 is its reduction modulo ℓ, thereby also proving the correctness of this algorithm.

Since I_i is the unique left $\mathcal{O}$-ideal of norm ℓ with $\mathcal{O}_R(I_i) \simeq \mathrm{End}(E/G_i)$, we have that

$$\iota(I_i) = \{\alpha \in \mathrm{End}(E) | G_i \subseteq \ker(\alpha)\}.$$

Then, reduced modulo ℓ, the equality becomes

$$\iota_\ell(\tilde{I}_i) = J_i,$$

where $\tilde{I}_i$ is the reduction of I_i modulo ℓ. On the other hand, by construction, we have $\lambda(\tilde{I}_i) = J_i$. Thus, by Corollary 3.2.5, the isomorphism λ_ℓ is equal to the isomorphism λ.

Now, we demonstraste the complexity of the algorithm by giving the complexity of its different steps. Obtaining the list of subgroups of order ℓ of the elliptic curve E can be done by computing a basis of the ℓ-torsion of E. This takes a polynomial time in ℓ and in $\log p$.

One can define the isomorphisms g, f and h by mapping the basis of the domain to a basis of the codomain such that the map verifies the respective required properties. Since there are $O(\ell^4)$ ordered bases of $M_2(\mathbb{F}_\ell)$, these constructions can be carried out using an exhaustive search.

Finally, by Lemma 3.2.3, running Algorithm 2 takes a polynomial time in the length of the input, in ℓ and in the length of the output of the oracle for MaxOrder. All the previously discussed complexities are encompassed within this running time.

□

Algorithm 4 Computing a non-scalar endomorphism, given a MaxOrder oracle

Input: A supersingular elliptic curve $E/\mathbb{F}_{p^2}$ and an access to an oracle for MaxOrder.

Output: An endomorphism $\theta \in \mathrm{End}(E) \setminus \mathbb{Z}$.

```
1: $\mathcal{O} \leftarrow$ a maximal order in a quaternion algebra such that $\mathcal{O} \simeq \mathrm{End}(E)$    ▷ Using MaxOrder oracle
2: $(\beta_i)_{i=1}^4 \leftarrow$ a Minkowski-reduced basis of $\mathcal{O}$    ▷ [NS09]
3: $\alpha \leftarrow \beta_2$    ▷ $\alpha$ is a shortest non-scalar vector in $\mathcal{O}$
4: $\ell \leftarrow 1, N \leftarrow 1$
5: while $N < \mathrm{Nrd}(\alpha)$ do
6:     $\ell \leftarrow$ the next prime after $\ell$ which is coprime to $\mathrm{Nrd}(\alpha)$
7:     $N \leftarrow \ell N$
8:     $(P_\ell, Q_\ell) \leftarrow$ a basis of the $\ell$-torsion $E[\ell]$
9:     $\lambda_\ell \leftarrow$ the isomorphism $\mathcal{O}/\ell\mathcal{O} \simeq \mathrm{End}(E[\ell])$    ▷ Using Algorithm 3
10:    $(P'_\ell, Q'_\ell) \leftarrow (\lambda_\ell(P_\ell), \lambda_\ell(Q_\ell))$
11: $\theta \leftarrow$ IsogenyInterpolation$((P_\ell, Q_\ell)_\ell, (P'_\ell, Q'_\ell)_\ell)$    ▷ Proposition 2.1.1
```

Proposition 3.2.7 (OneEnd reduces to MaxOrder)**.** *Algorithm 4 is correct and runs in probabilistic polynomial time in the length of the instance and in the length of the oracle's output.*

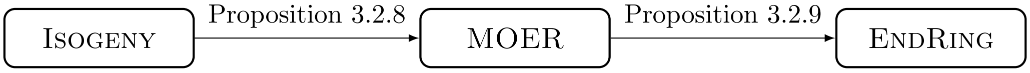
Proof. By the references cited in the comments, each step is at most polynomial in $\log p$, in the length of the MaxOrder oracle's output and in ℓ (whenever a prime ℓ is involved in the computation). Therefore, to prove the claimed complexity, it remains only to establish bounds on the number of iterations of the loop at line 5 and on the considered primes ℓ. By Proposition 1.3.33, we have that $\mathrm{Nrd}(\alpha) \leq 2p^{2/3}$. Therefore, by the prime number theorem, the while loop at line 5 has $O(\log p)$ iterations and the largest ℓ considered is $O(\log p)$, proving the claimed complexity.

The correctness of Algorithm 4 comes from the fact that we return the output of the `IsogenyInterpolation` algorithm called on input corresponding to the evaluation of $\iota(\alpha)$ on the N-torsion subgroup $E[N]$ with $N > \mathrm{Nrd}(\alpha)$, where ι is the Deuring isomorphism.

□

Reductions to the Endomorphism Ring problem

We now turn to proving that Isogeny and MOER reduce in polynomial time to EndRing, thereby completing the equivalence of all problems presented in Figure 3.2, except for HomModule, which we discuss in the next subsection. We shall proceed by proving the following sequence of reductions:



The main difficulty in reducing the Isogeny problem between two curves to the MaxOrder problem lies in translating a connecting ideal between two maximal orders, which are isomorphic to the endomorphism ring of the curves, into an isogeny between the curves. Before the break of SIDH [CD23, MMP$^+$23, Rob23a], this process first required finding a more suitable ideal, using algorithms similar to the one introduced in [KLPT14] (we refer to such algorithms as `KLPT`-like algorithms), which can be proven under GRH [Wes22b]. Then computing the corresponding isogeny using an elliptic curve with known endomorphism ring as a dictionary between endomorphisms and quaternions. Thanks to Proposition 2.3.3, it is now possible to directly compute the isogeny corresponding to the connecting ideal. However, one still needs to know an elliptic curve with an explicit basis of its endomorphism ring. This is why, instead of reducing Isogeny to MaxOrder, we reduce it to the MOER problem, ensuring access to such a curve.

Proposition 3.2.8 (Isogeny reduces to MOER)**.** *Given access to a* MOER *oracle, one can solve the* Isogeny *problem in time polynomial in the length of its input and in the length of the oracle's output.*

Proof. Let E_1 and E_2 be two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$. For $i \in \{1, 2\}$, a MOER oracle provides a maximal order $\mathcal{O}_i$ in a quaternion algebra $B_i \simeq B_{p,\infty}$ together with an isomorphism $\varepsilon_i : \mathcal{O}_i \xrightarrow{\simeq} \mathrm{End}(E_i)$. By proposition 1.3.38, one can compute in polynomial time an isomorphism $\varepsilon : B_2 \xrightarrow{\simeq} B_1$. Then $\mathcal{O}_2' := \varepsilon(\mathcal{O}_2)$ is a maximal order in B_1 isomorphic to $\mathrm{End}(E_2)$. Using [KV10, Algorithm 3.5], one can compute efficiently the connecting ideal $I = I(\mathcal{O}_1, \mathcal{O}_2')$. Finally, by Proposition 2.3.3, one can compute the isogeny $\varphi_I : E_1 \to E_2$ in polynomial time. □

We reduce MOER to EndRing by adapting the strategy of [EHL$^+$18, Algorithm 6]. The freedom to choose a model for $B_{p,\infty}$ in the definition of MOER allows us to eliminate all heuristics in the proof of [EHL$^+$18, Algorithm 6]. We recall that, using Proposition 1.3.38, one can always translate a MOER solution into any target quaternion algebra where a maximal order is already known.

Proposition 3.2.9 (MOER reduces to EndRing)**.** *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$ together with a basis of its endomorphism ring* $\mathrm{End}(E)$*, one can solve the* MOER *instance corresponding to the curve E in time polynomial in* $\log p$ *and in the length of the elements in the provided basis of* $\mathrm{End}(E)$*.*

Proof. Let $(\gamma_i)_{i=0}^4$ be a basis of the endomorphism ring of a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$. By [EHL$^+$18, Lemma 4 and Lemma 5], one can compute, in time polynomial in $\log p$ and in $\log \max_{i=1}^4(\deg(\gamma_i))$, a rational invertible linear transformation F sending $(\gamma)_{i=1}^4$ to some orthogonal basis $(1, \alpha, \beta, \alpha\beta)$. In particular, $(1, \alpha, \beta, \alpha\beta)$ is a basis of the endomorphism algebra $\mathrm{End}(E) \otimes \mathbb{Q}$ such that $\alpha^2 < 0$, $\beta^2 < 0$ and $\alpha\beta = -\beta\alpha$. Hence, it is isomorphic to the quaternion algebra $B = (\frac{\alpha^2, \beta^2}{\mathbb{Q}})$, with basis $(1, i, j, ij)$ such that $i^2 = \alpha^2, j^2 = \beta^2$ and $ij = -ji$. Let $\varepsilon : \mathrm{End}(E) \otimes \mathbb{Q} \xrightarrow{\sim} B$ be the explicit isomorphism sending $(1, \alpha, \beta, \alpha\beta)$ to $(1, i, j, ij)$. By applying F^{-1} to $(1, i, j, ij)$ we get a maximal order $\mathcal{O} = \varepsilon((\gamma_i)_{i=1}^4)$ in B isomorphic to $\mathrm{End}(E)$. Finally, since B is isomorphic to $\mathrm{End}(E) \otimes \mathbb{Q}$, which is itself isomorphic to $B_{p,\infty}$, the solution we found satisfies all the conditions of the MOER problem. □

The Homomorphism Module problem

The HomModule problem has not been formally studied in the previous literature, yet it naturally appears in isogeny-based cryptography. For instance, the homomorphism module between the commitment curve and the challenge curve in SQISign [DKL+20] is the space of all possible responses during an identification. While it is clear that Isogeny reduces to HomModule, we prove in this subsection that both are actually equivalent.

The relation between HomModule and Isogeny is reminiscent of the relations between EndRing and OneEnd. The latter equivalence has been proved in [PW24]. In this same paper, the authors also proved that OneEnd reduces to Isogeny, both of these reductions being unconditional. Therefore, there is a probabilistic polynomial time algorithm solving EndRing given an Isogeny oracle. In order to take advantage of this fact to solve HomModule we prove that knowing an isogeny between two elliptic curves and their respective endomorphism rings, one can compute efficiently a basis of the homomorphism module between the said curves, Proposition 3.2.13 and Proposition 3.2.14. This leads to the main result of the section, Proposition 3.2.15, proving that HomModule reduces to Isogeny.

Lemma 3.2.10. *Let $\varphi_i : E \to E_i$, for $i \in \{1, 2\}$, be separable isogenies such that* $\ker \varphi_1 \cap \ker \varphi_2 = 0$*. Then,*

$$\mathrm{Hom}(E_1, E)\varphi_1 + \mathrm{Hom}(E_2, E)\varphi_2 = \mathrm{End}(E).$$

Proof. Let $\varphi_3 : E \to E_3$ be a separable isogeny with $\ker \varphi_3 = \ker \varphi_1 + \ker \varphi_2$. Since $\ker \varphi_1 \cap \ker \varphi_2 = 0$, we have $|\ker \varphi_3| = |\ker \varphi_1||\ker \varphi_2|$. Each

$$I_i = \mathrm{Hom}(E_i, E)\varphi_i = \{\alpha \in \mathrm{End}(E) \mid \ker \varphi_i \subseteq \ker \alpha\}$$

is a left $\mathrm{End}(E)$-ideal of reduced norm $\mathrm{Nrd}(I_i) = |\ker \varphi_i|$. We have

$$I_1 \cap I_2 = \{\alpha \in \mathrm{End}(E) \mid (\ker \varphi_1 + \ker \varphi_2) \subseteq \ker \alpha\} = I_3.$$

We have,

$$\begin{aligned} |I_1/(I_1 \cap I_2)|^{1/2} = \frac{\mathrm{Nrd}(I_3)}{\mathrm{Nrd}(I_1)} &= \frac{|\ker \varphi_3|}{|\ker \varphi_1|} = \frac{|\ker \varphi_1||\ker \varphi_2|}{|\ker \varphi_1|} \\ &= \mathrm{Nrd}(I_2) = |\mathrm{End}(E)/I_2|^{1/2}. \end{aligned}$$

By the *second isomorphism theorem,*

$$I_1/(I_1 \cap I_2) \cong (I_1 + I_2)/I_2 \subseteq \mathrm{End}(E)/I_2,$$

and since the leftmost and rightmost quotients have the same cardinality, we deduce $(I_1 + I_2)/I_2 = \mathrm{End}(E)/I_2$, hence $I_1 + I_2 = \mathrm{End}(E)$. □

Lemma 3.2.11. *Let $\varphi_i : E \to E_i$, for $i \in \{1, 2\}$, be separable isogenies such that* $\ker \varphi_1 \cap \ker \varphi_2 = 0$*. Then, for any elliptic curve E',*

$$\mathrm{Hom}(E_1, E')\varphi_1 + \mathrm{Hom}(E_2, E')\varphi_2 = \mathrm{Hom}(E, E').$$

Proof. Clearly $\mathrm{Hom}(E_1, E')\varphi_1 + \mathrm{Hom}(E_2, E')\varphi_2 \subseteq \mathrm{Hom}(E, E')$, so let us prove the second inclusion. By Lemma 3.2.10,

$$\begin{aligned} \mathrm{Hom}(E, E') &= \mathrm{Hom}(E, E')\,\mathrm{End}(E) \\ &= \mathrm{Hom}(E, E')(\mathrm{Hom}(E_1, E)\varphi_1 + \mathrm{Hom}(E_2, E)\varphi_2) \\ &\subseteq \mathrm{Hom}(E_1, E')\varphi_1 + \mathrm{Hom}(E_2, E')\varphi_2, \end{aligned}$$

which proves the result. □

Proposition 3.2.12. *Let $\varphi : E \to E'$ be a separable isogeny. Then,*

$$\mathrm{span}_{\mathbb{Z}}(\mathrm{End}(E')\varphi\,\mathrm{End}(E)) = m\,\mathrm{Hom}(E,E'),$$

where $m \in \mathbb{Z}$ is the largest integer dividing φ.

Proof. Clearly $\mathrm{span}_{\mathbb{Z}}(\mathrm{End}(E')\varphi\,\mathrm{End}(E)) \subseteq m\,\mathrm{Hom}(E,E')$, so let us prove the other inclusion. Write $\varphi = m\psi$ with $\ker\psi$ cyclic. Let $n = \deg(\psi)$. The kernel $\ker\psi \cong \mathbb{Z}/n\mathbb{Z}$ is a cyclic subgroup of $E[n] \cong (\mathbb{Z}/n\mathbb{Z})^2$. The action of $\mathrm{End}(E)/n\,\mathrm{End}(E)$ on $E[n]$ is isomorphic to the action of $M_2(\mathbb{Z}/n\mathbb{Z})$ on $(\mathbb{Z}/n\mathbb{Z})^2$, so there exists an endomorphism $\alpha \in \mathrm{End}(E)$ (of degree coprime with n) such that $\ker(\psi) \cap \alpha^{-1}(\ker\psi) = 0$. In other words, $\ker(\psi) \cap \ker(\psi\alpha) = 0$. Applying Lemma 3.2.11, we deduce

$$\mathrm{End}(E')\psi + \mathrm{End}(E')\psi\alpha = \mathrm{Hom}(E,E').$$

We deduce

$$m\,\mathrm{Hom}(E,E') = \mathrm{End}(E')\varphi + \mathrm{End}(E')\varphi\alpha \subseteq \mathrm{span}_{\mathbb{Z}}(\mathrm{End}(E')\varphi\,\mathrm{End}(E)),$$

which proves the proposition. □

Recall that any isogeny can be factored as $\phi\varphi$ where φ is separable, and ϕ is purely inseparable (ϕ might be an isomorphism). Then, the following proposition generalized Proposition 3.2.12 to arbitrary isogenies.

Proposition 3.2.13. *Let $\varphi : E \to E''$ be a separable isogeny, and $\phi : E'' \to E'$ a purely inseparable isogeny. Then,*

$$L = \mathrm{span}_{\mathbb{Z}}(\mathrm{End}(E')\phi\varphi\,\mathrm{End}(E)) = m\phi\,\mathrm{Hom}(E,E''),$$

where $m \in \mathbb{Z}$ is the largest integer dividing φ.

Proof. Since $\phi : E'' \to E'$ is purely inseparable, we have $\mathrm{End}(E')\phi = \phi\,\mathrm{End}(E'')$. The result then immediately follows from Proposition 3.2.12. □

Proposition 3.2.14. *Let E, E' and E'' be supersingular elliptic curves, and $\phi : E'' \to E'$ a purely inseparable isogeny. Given a basis of $\phi\,\mathrm{Hom}(E,E'')$, one can compute a basis of $\mathrm{Hom}(E,E')$ in polynomial time.*

Proof. Let $(b_i)_{i=1}^4$ be the provided basis of the lattice $L = \phi\,\mathrm{Hom}(E,E'')$. Let $p^n = \deg(\phi)$. If $n = 2m$ is even, then $\phi = p^m\alpha$ where $\alpha : E'' \to E'$ is an isomorphism. Then $(b_i/p^m)_{i=1}^4$ is a basis of $\alpha\,\mathrm{Hom}(E,E'') = \mathrm{Hom}(E,E')$. If $n = 2m+1$ is odd, one can similarly divide by p^m, and without loss of generality we now consider the case where ϕ is the p-Frobenius. Consider the quadratic form

$$q : L \longrightarrow \mathbb{Z} : \varphi \longmapsto \deg(\varphi)/p.$$

We have

$$p\,\mathrm{Hom}(E,E') = L \cap (p\,\mathrm{Hom}(E,E')) = \{\varphi \in L \mid q(\varphi) \equiv 0 \bmod p\}.$$

The equation $q(\varphi) \equiv 0 \bmod p$ defines an $\mathbb{F}_p$-linear subspace of L/pL which can be computed as the kernel of the Gram matrix over $\mathbb{F}_p$. □

Proposition 3.2.15 (HomModule reduces to Isogeny)**.** *Algorithm 5 is correct and runs in time polynomial in $\log p$ and in the length of the oracle outputs.*

Algorithm 5 Reducing HomModule to Isogeny

Input: Two supersingular elliptic curves E_1 and E_2 and an access to an oracle of Isogeny.

Output: Four isogenies $\varphi_i : E_1 \to E_2$, $i \in \{1, ..., 4\}$ generating $\mathrm{Hom}(E_1, E_2)$ as a $\mathbb{Z}$-module.

```
1: $(\alpha_1, \alpha_2, \alpha_3, \alpha_4) \leftarrow$ a basis of $\mathrm{End}(E_1)$        ▷ [PW24, Theorem 8.6]
2: $(\beta_1, \beta_2, \beta_3, \beta_4) \leftarrow$ a basis of $\mathrm{End}(E_2)$         ▷ [PW24, Theorem 8.6]
3: Compute an isogeny $\varphi : E_1 \to E_2$                                ▷ Using the Isogeny oracle
4: $v \leftarrow v_p(\deg \varphi)$                                           ▷ $p$-adic valuation of $\deg \varphi$
5: $S \leftarrow \{\beta_j \circ \varphi \circ \alpha_i\} \subset \mathrm{Hom}(E_1, E_2)$
6: $(\gamma_1, \gamma_2, \gamma_3, \gamma_4) \leftarrow$ a basis of the lattice generated by $S$   ▷ [BP89]
7: $m \leftarrow \left(16 \det(\langle \gamma_i, \gamma_j \rangle)/p^{4v+2}\right)^{1/8}$
8: $B_0 \leftarrow (\gamma_1/m, \gamma_2/m, \gamma_3/m, \gamma_4/m)$              ▷ Theorem 2.2.1
9: Extract a basis of $\mathrm{Hom}(E, E')$ from $B$                          ▷ Proposition 3.2.14
10: return $B$
```

Proof. The running time of each step is ensured by the corresponding reference. In particular, each is polynomial in $\log p$ and in the length of the oracle's outputs.

Let us prove the correctness of the algorithm by treating the cases where the isogeny φ, returned by the Isogeny oracle at Step 3, is inseparable and separable independently.

We now assume that $\varphi = \phi \circ \psi$ where $\psi : E_1 \to E'$ is a separable isogeny and $\phi : E' \to E_2$ is a purely inseparable isogeny of degree $p^v \geq 1$, where $v := v_p(\deg \varphi)$ (when $v = 0$, the purely inseparable part ϕ is an isomorphism). Let m be the largest integer that divides ψ. Then, by Proposition 3.2.13, the set S generates the lattice $m\phi \,\mathrm{Hom}(E_1, E')$. For a basis $(\gamma_1, \dots, \gamma_4)$ of the lattice generated by S, we have

$$\begin{aligned} \det(\langle \gamma_i, \gamma_j \rangle) &= \mathrm{Vol}(m\phi \,\mathrm{Hom}(E_1, E'))^2 \\ &= (m^4 \deg(\phi)^2 \mathrm{Vol}(\mathrm{Hom}(E_1, E')))^2 \\ &= m^8 p^{4v+2}/16, \end{aligned}$$

thus the computation at Step 7 gives the correct m. In particular, the basis $B = (\gamma_1/m, \dots, \gamma_4/m)$ generates $\phi \,\mathrm{Hom}(E_1, E')$. From it, one can compute a basis of $\mathrm{Hom}(E_1, E_2)$ using Proposition 3.2.14. □

3.3 Compact problems

In our hard problem definitions, we ask for solutions that provide an efficient representation for each output isogeny. This does not ensure that the length of the solutions is polynomial in $\log p$. In this section, we prove that these problems are equivalent to "compact" problems.

Definition 3.3.1 (Compact problem)**.** *A problem P is said to be* ***compact*** *if the size of its solutions is polynomial in the size of its input.*

Specifically, for isogeny-based cryptography, a problem is compact if its solutions have length polynomial in $\log p$. The choice of this terminology comes from the definition of *compact representations of endomorphisms* in [EHL+18], where it is used to characterise endomorphisms with a representation of polynomial size. The variants of hard problems we define aim for some kind of "minimal" solutions:

- For problems like EndRing, where the solution is a basis of a $\mathbb{Z}$-module of rank 4, we ask for Minkowski-reduced bases. In this case, by [vdW56], we know that the elements of the basis achieve the four successive minima, which reflects a natural minimality condition.

- For problems like Isogeny, asking for a single isogeny, we ask for an isogeny with the smallest possible degree.

Clearly, the reduction from a hard problem to such a variant is straightforward. We shall prove the other direction, and therefore the equivalence, for most of the hard problems. These equivalences might seem counterintuitive. Indeed, this implies, for instance, that having access to an Isogeny oracle that returns only isogenies of large degree is enough to find the smallest isogeny between two elliptic curves.

The EndRing family is compact

Let us first consider the EndRing, MaxOrder, MOER and HomModule problems. For each of them, we define a "reduced" variant asking for a Minkowski-reduced basis. Since isogenies are assumed to be efficiently represented, this implies that the reduced variants of EndRing and HomModule are compact problems. Nevertheless, for the reduced variants of MaxOrder and MOER to be compact, one also needs to ask for a model of the quaternion algebra and for bases of the maximal orders of length polylog(p). After defining formally each reduced variant, we prove that they are compact and reduce to the original problem.

We begin with the simplest cases: EndRing and HomModule.

Definition 3.3.2 (reduced-HomModule). *Given E and E', two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$, find four isogenies generating* $\mathrm{Hom}(E, E')$ *as a $\mathbb{Z}$-module such that they form a Minkowski-reduced basis.*

Proposition 3.3.3. *The* reduced-HomModule *problem is compact.*

Proof. Let $(\varphi_i)_{i=1}^4$ be a Minkowski-reduced basis of $\mathrm{Hom}(E, E')$ for E and E' two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$. By definition, the efficient representation of each isogeny φ_i has size polynomial in $\log(\deg(\varphi_i))$ and $\log p$. Since the elements of the Minkowski-reduced basis achieve the successive minima, we have that the degree of the isogeny φ_i is $O(p)$; see Proposition 1.3.33. Thus the representation of each isogeny has length polynomial in $\log p$. □

Proposition 3.3.4. *The* reduced-HomModule *problem reduces to the* HomModule *problem in probabilistic polynomial time in the length of the instance.*

Proof. Let E, E' be two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$ and $(\varphi_i)_{i=1}^4$ be a basis of $\mathrm{Hom}(E, E')$. Let us compute a reduced-HomModule solution from this basis.

By Proposition 1.3.30, the homomorphism module $\mathrm{Hom}(E, E')$ is a lattice of rank 4. Note that since, for any $i \in [\![1, 4]\!]$, the isogeny φ_i is efficiently represented, one can perform all the necessary operations to compute a Minkowski-reduced basis of the lattice in time polynomial in the length of the input basis. Indeed, by Proposition 1.3.31, one can evaluate the bilinear map in polynomial time, and thus compute Gram matrices in polynomial time. In addition, by the `IsogenyDivision` algorithm, Theorem 2.2.1, for any $(q_i)_{i=1}^4 \in \mathbb{Q}^4$, one can compute in polynomial time an efficient representation of the isogeny $\sum_{i=1}^4 q_i\varphi_i$, if well-defined, of length polynomial in $\log p$ and in the length of its degree. Thus, using [NS09], one can compute a Minkowski-reduced basis of $\mathrm{Hom}(E, E')$, where each isogeny is efficiently represented, in time polynomial in the length of the input basis. □

The situation for the EndRing problem is completely analogous.

Problem 3.3.5 (reduced-EndRing). *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find four endomorphisms generating* $\operatorname{End}(E)$ *as a* $\mathbb{Z}$*-module such that they form a Minkowski-reduced basis.*

Proposition 3.3.6. *The* reduced-EndRing *problem is compact and reduces to* EndRing *in probabilistic polynomial time in the length of the instance.*

Proof. The proofs of Proposition 3.3.3 and Proposition 3.3.4 also apply here with $E = E'$. □

It has been proven that one can construct a Minkowski-reduced basis of length polylog(p) for any maximal order in $B_{p,\infty}$, [EHL$^+$18, Theorem 2]. This proof relies on the existence of the special maximal order $\mathcal{O}_0$ in $B_{p,\infty}$ as defined in Proposition 1.3.39. Without **GRH**, when the characteristic of the field is congruent to 1 mod 8, it is no longer possible to work directly with $B_{p,\infty}$ and the special order $\mathcal{O}_0$. Hence, we have to proceed differently to obtain an unconditional reduction.

Problem 3.3.7 (reduced-MaxOrder). *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find a quaternion algebra $B = (\frac{-a,-b}{\mathbb{Q}}) \simeq B_{p,\infty}$ and four quaternions generating a maximal order in B isomorphic to* $\operatorname{End}(E)$ *such that the four quaternions form a Minkowski-reduced basis. In addition, the total output must be of length polynomial in* $\log p$.

The compactness of reduced-MaxOrder is directly given by its definition. We now prove its reduction to the MaxOrder problem.

Proposition 3.3.8. *The* reduced-MaxOrder *problem reduces to the* MaxOrder *problem in probabilistic polynomial time in the length of the instance.*

Proof. Let us compute a solution to the reduced-MaxOrder problem from a given MaxOrder solution. This input solution consists of a basis $(\theta_i)_{i=1}^4$ which generates a maximal order $\mathcal{O}$ in a quaternion algebra $C = (\frac{-c,-d}{\mathbb{Q}}) \simeq B_{p,\infty}$.

First, we compute a Minkowski-reduced basis $(1, \alpha_0, \beta_0, \gamma_0)$ from the basis $(\theta_i)_{i=1}^4$, this takes polynomial time by [NS09]. There is no guarantee that the length of α_0, β_0 and γ_0 expressed in terms of the canonical basis of C, as well as the length of the integers c and d, are bounded by a polynomial in $\log p$. In order to provide a solution of length polynomial in $\log p$, we shall compute a more suitable quaternion algebra representation. The fact that, by Proposition 1.3.33, the reduced norm of α_0, β_0 and γ_0 are polynomial in p — in fact they are even $O(p)$ — is crucial.

Let us define the quaternions

$$\alpha := \alpha_0 - \frac{1}{2}\operatorname{Trd}(\alpha_0),$$
$$\beta := \beta_0 - \frac{1}{2}\operatorname{Trd}(\beta_0) - \frac{\operatorname{Trd}(\alpha\beta_0)}{2\alpha^2}\alpha.$$

According to [Sil09, proof of Theorem 9.3], we have $-a := \alpha^2 < 0$, $-b := \beta^2 < 0$, and $\alpha\beta = -\beta\alpha$. Moreover, $a = \operatorname{Nrd}(\alpha)$ and $b = \operatorname{Nrd}(\beta)$, as their reduced traces are null. This choice for α and β actually follows the first three steps of the Gram-Schmidt orthogonalization process. Then, the quaternion algebra $B := (\frac{-a,-b}{\mathbb{Q}})$ is isomorphic to $\mathrm{End}(E)^0$ and is given by $\mathbb{Q} + \alpha\mathbb{Q} + \beta\mathbb{Q} + \alpha\beta\mathbb{Q}$.

We now show that the rational numbers c and d have been replaced by rationals with representations of length polylog(p). A short computation yields

$$\mathrm{Nrd}(\alpha) = \mathrm{Nrd}(\alpha_0) + \frac{1}{2}\mathrm{Trd}(\alpha_0)^2,$$
$$\mathrm{Nrd}(\beta) = \mathrm{Nrd}(\beta_0) + \frac{1}{2}\mathrm{Trd}(\beta_0)^2 + \frac{3}{4\,\mathrm{Nrd}(\alpha)}\mathrm{Trd}(\alpha\beta_0)^2.$$

We recall that for any quaternion γ, we have $\mathrm{Trd}(\gamma)^2 < 4\,\mathrm{Nrd}(\gamma)$. Thus $\mathrm{Nrd}(\alpha) < 3\,\mathrm{Nrd}(\alpha_0)$. Then, we can write the rational a and b as

$$a = \frac{2\,\mathrm{Nrd}(\alpha_0) + \mathrm{Trd}(\alpha_0)^2}{2},$$
$$b = \frac{2a(2\,\mathrm{Nrd}(\beta_0) + \mathrm{Trd}(\beta_0)^2) + 3\,\mathrm{Trd}(\alpha\beta_0)^2}{4a}.$$

By Proposition 1.3.33, we have that $\mathrm{Nrd}(\alpha_0), \mathrm{Trd}(\alpha_0)^2, \mathrm{Nrd}(\beta_0), \mathrm{Trd}(\beta_0)^2$ and a are $O(p)$. In addition,

$$\mathrm{Trd}(\alpha\beta_0)^2 < 4\,\mathrm{Nrd}(\alpha\beta_0) = 4\,\mathrm{Nrd}(\alpha)\,\mathrm{Nrd}(\beta_0) = O(p^2).$$

Hence, there exist representations of a and b of length $O(\log(p))$.

It only remains to show that the basis $(1, \alpha_0, \beta_0, \gamma_0)$ of the maximal order $\mathcal{O}$ has polynomial length when expressed in the canonical basis $(1, \alpha, \beta, \alpha\beta)$ of B. By construction, the quaternions $1, \alpha_0, \beta_0$ can be written as

$$1 = (1, 0, 0, 0),$$
$$\alpha_0 = (\frac{1}{2}\mathrm{Trd}(\alpha_0), 1, 0, 0),$$
$$\beta_0 = (\frac{1}{2}\mathrm{Trd}(\beta_0), \frac{\mathrm{Trd}(\alpha\beta_0)}{2\alpha^2}, 1, 0).$$

With the same arguments as above, every coefficient is given by a representation of length polynomial in $\log(p)$.

We now turn to expressing γ_0 in this basis. Applying the last step of Gram-Schmidt orthogonalization process, we define

$$\gamma := \gamma_0 - \frac{1}{2}\mathrm{Trd}(\gamma_0) - \frac{\mathrm{Trd}(\alpha\gamma_0)}{2\alpha^2}\alpha - \frac{\mathrm{Trd}(\beta\gamma_0)}{2\beta^2}\beta.$$

Note that $\mathrm{Nrd}(\gamma)$ is polynomial in p. Furthermore, as shown in the proof of [EHL$^+$18, Lemma 5], γ is a multiple of $\alpha\beta$. Hence, there is some rational number $\frac{r}{s}$ such that

$$\gamma_0 := \frac{1}{2}\mathrm{Trd}(\gamma_0) + \frac{\mathrm{Trd}(\alpha\gamma_0)}{2\alpha^2}\alpha + \frac{\mathrm{Trd}(\beta\gamma_0)}{2\beta^2}\beta + \frac{r}{s}\alpha\beta.$$

By the same arguments as above, every coefficient involved in the writing of γ_0 in terms of $(1, \alpha, \beta, \alpha\beta)$, except r and s, have a length polynomial in $\log p$. In fact, this is also the case for r and s, thanks to the following simply observation. By definition of r and s,

$$\frac{r^2}{s^2} = \frac{\mathrm{Nrd}(\alpha)\,\mathrm{Nrd}(\beta)}{\mathrm{Nrd}(\gamma)},$$

then the rational $\sqrt{\frac{\mathrm{Nrd}(\alpha)\,\mathrm{Nrd}(\beta)}{\mathrm{Nrd}(\gamma)}}$ is equal to $\frac{r}{s}$ such that its numerator and denominator have length polynomial in $\log p$. □

The last reduced problem we study here is the variant of MOER. The proof of its equivalence with the standard MOER problem is directly given by the above propositions.

Definition 3.3.9 (REDUCED-MOER)**.** *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find four endomorphisms $(\alpha_i)_{i=1}^4$ generating* End(E) *as a $\mathbb{Z}$-module, a quaternion algebra $B = \frac{(-a,-b)}{\mathbb{Q}} \simeq B_{p,\infty}$, and four quaternions $(\beta_i)_{i=1}^4$ in B such that*

$$\mathrm{End}(E) \times \mathbb{Q} \longrightarrow B : \alpha_i \longmapsto \beta_i$$

is an isomorphism. The bases generated by $(\alpha_i)_{i=1}^4$ and $(\beta_i)_{i=1}^4$ must be Minkowski-reduced. In addition, the total length of the output must be of length polynomial in $\log p$*.*

Once again, the compactness of the REDUCED-MOER is included in its definition.

Proposition 3.3.10. *The* REDUCED-MOER *problem reduces to the* MOER *problem in probabilistic polynomial time in the length of the instance.*

Proof. Given the isomorphism

$$\mathrm{End}(E) \longrightarrow \mathcal{O} : \theta_i \longmapsto \gamma_i, \forall i \in [\![1,4]\!],$$

where $\mathcal{O}$ is a maximal order in a quaternion algebra $C = (\frac{-c,-d}{\mathbb{Q}})$, let us compute a solution of the REDUCED-MOER problem for the elliptic curve $E/\mathbb{F}_{p^2}$.

We begin with the same first step as in the proof of Problem 3.3.7 and Proposition 3.3.6: computing a Minkowski-reduced basis. In this case, we perform the reduction on $(\theta_i)_{i=1}^4$ and $(\gamma_i)_{i=1}^4$ simultaneously, in order to obtain a new isomorphism

$$\mathrm{End}(E) \longrightarrow \mathcal{O} : \alpha_i \longmapsto \beta_i, \forall i \in [\![1,4]\!],$$

where the bases $(\alpha_i)_{i=1}^4$ and $(\beta_i)_{i=1}^4$ are now Minkowski-reduced.

Moreover, as for REDUCED-ENDRING, the basis $(\alpha_i)_{i=1}^4$ has length polynomial in $\log p$. Then, we apply the same method as in Problem 3.3.7 to compute two rational numbers a and b, of length polynomial in $\log p$, together with four quaternions generating the quaternion algebra $(\frac{-a,-b}{\mathbb{Q}})$. Again as in the proof of Problem 3.3.7, the Minkowski-reduced basis of the maximal order $\mathcal{O}$ can be expressed in length polynomial in $\log p$ in terms of the canonical basis of $(\frac{-a,-b}{\mathbb{Q}})$. Thus, we have a compact solution for the given MOER instance. □

The ISOGENY family is compact

Let us turn now to the equivalence between the ISOGENY problem (resp. the ONEEND problem) and variant problems which are compact. These variants ask for an efficient representation of polynomial length of a smallest isogeny (resp. endomorphism) in terms of degree. This result can be seen as a direct application of Theorem 3.2.1.

Definition 3.3.11 (MINIMAL-ISOGENY)**.** *Given two supersingular elliptic curves E and E' defined over $\mathbb{F}_{p^2}$, find an isogeny $\varphi : E \to E'$ with the smallest possible degree.*

Proposition 3.3.12. *The* MINIMAL-ISOGENY *problem is compact.*

Proof. By Proposition 1.3.33, the smallest isogeny between two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$ has degree $O(p)$. Thus its efficient representations have length polynomial in $\log p$. □

Proposition 3.3.13. *The* MINIMAL-ISOGENY *problem reduces to the* ISOGENY *problem in probabilistic polynomial time in the length of the instance.*

Proof. On the one hand, by equivalence between the different hard problems, Theorem 3.2.1, and by reduction from REDUCED-HOMMODULE to HOMMODULE, Proposition 3.3.4, we have that the REDUCED-HOMMODULE problem reduces to the ISOGENY problem.

On the other hand, the first element of a Minkowski-reduced basis, corresponding to a REDUCED-HOMMODULE solution, is a suitable solution to the MINIMAL-ISOGENY problem. Indeed, it is an isogeny with the smallest possible degree and it has an efficient representation of length polynomial in $\log p$. This comes from the definition of the REDUCED-HOMMODULE problem and from the fact that a Minkowski-reduced basis achieves all the successive minima, see [vdW56].

Hence, we have the following sequence of reductions

$$\text{MINIMAL-ISOGENY} \rightarrow \text{REDUCED-HOMMODULE} \rightarrow \text{HOMMODULE} \rightarrow \text{ISOGENY}.$$

Thus MINIMAL-ISOGENY reduces to ISOGENY as claimed. □

Definition 3.3.14 (MINIMAL-ONEEND)**.** *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$, find an endomorphism of E with the smallest possible degree.*

Proposition 3.3.15. *The* MINIMAL-ONEEND *problem is compact and reduces to the* ONEEND *problem in probabilistic polynomial time in the length of the instance.*

Proof. The proof of compactness is analogous to Proposition 3.3.12. Then the reduction is the same as for Proposition 3.3.13 except that it relies on REDUCED-ENDRING and Proposition 3.3.6, instead of REDUCED-HOMMODULE and Proposition 3.3.4. □

Remark 3.3.16. *Given the current state of the art, proving a polynomial time reduction from the problem of finding the smallest path between two vertices in the ℓ-isogeny graph to the ℓ-*ISOGENYPATH *problem seems unreachable — even under* ***GRH*** *or plausible heuristic assumptions.*

A natural approach is to compute a connecting ideal between the two input elliptic curves such that its reduced norm is the smallest possible power of ℓ. Since the MOER *problem reduces to the ℓ-*ISOGENYPATH *problem, it is feasible to compute suitable maximal orders and derive isogenies from connecting ideals.*

As the diameter of the ℓ-isogeny graph is $O(\log_\ell(p))$, the degree of the smallest chain of ℓ-isogenies between two curves is $O(p)$. However, there is currently no algorithm achieving this bound. For instance, the heuristic ***KLPT*** *algorithm [KLPT14, Section 4.5] produces ideals of reduced norm ℓ^e with $e \sim \frac{7}{2}(\log_\ell p)$. While its variant proven under* ***GRH*** *guarantees only that the reduced norm is polynomial in p, [Wes22b, Theorem 6.3].*

This minimal degree isogeny between two elliptic curves can vary significantly. For instance, neighbors in the 2-isogeny graph are, by definition, connected by a small isogeny of degree 2. In contrast, two random curves may only have isogenies of degree exponential in $\log p$ between them. The same occurs for the ONEEND problem, where the minimal non-trivial endomorphisms of two different curves might have degrees that differ widely.

A less restrictive way to control the degree of ISOGENY solutions, compared to the MINIMAL-ISOGENY problem, has been formalised in [PW24]. We extend it to ONEEND. Given a function $\lambda : \mathbb{Z}_{>0} \rightarrow \mathbb{Z}_{>0}$, we define ISOGENY_λ (resp. ONEEND_λ) by limiting the set of solutions to isogenies (resp. endomorphisms) whose degree length is bounded by λ evaluated at the length of the input. For example, an isogeny $\varphi : E \rightarrow E'$ is a solution to the ISOGENY_λ on input (E, E') such that

$$\log(\deg(\varphi)) \leq \lambda(\log p).$$

In particular, the ISOGENY$_\lambda$ and ONEEND$_\lambda$ problems are compact when the function λ is a polynomial.

When the function λ grows sufficiently fast, the ISOGENY$_\lambda$ (resp. ONEEND$_\lambda$) problem becomes equivalent to the standard ISOGENY (resp. ONEEND) problem. More precisely, the function λ must satisfy the condition

$$\lambda(\log(p)) < \log(N(p)),$$

where $N(p)$ is an upper bound on the degree of the smallest isogenies between any two elliptic curves (resp. smallest endomorphisms of any curve) defined over $\mathbb{F}_{p^2}$. The current best bounds are $O(\sqrt{p})$ for isogenies and $O(p^{2/3})$ for endomorphisms, see Proposition 1.3.33.

Proposition 3.3.17. *Let $\lambda : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ be a function such that $\lambda(\log x) \geq \log(\frac{2\sqrt{2x}}{\pi})$. Then the* ISOGENY$_\lambda$ *problem reduces to the* ISOGENY *problem in probabilistic polynomial time in the length of the instance.*

Proof. By Proposition 1.3.33, the smallest isogenies between two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$ have degree at most $\frac{2\sqrt{2p}}{\pi}$. Hence, an isogeny φ solving the MINIMAL-ISOGENY problem verifies that

$$\lambda(\log p) > \log(\frac{2\sqrt{2p}}{\pi}) \geq \log(\deg \varphi).$$

Thus, within this setting, the ISOGENY$_\lambda$ problem reduces to the MINIMAL-ISOGENY problem. We conclude that ISOGENY$_\lambda$ reduces to ISOGENY, using Proposition 3.3.13. □

Proposition 3.3.18. *Let $\lambda : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ be a function such that $\lambda(\log x) > \log(2x^{2/3})$. Then the* ONEEND$_\lambda$ *problem reduces to the* ONEEND *problem in probabilistic polynomial time in the length of the instance.*

Proof. The proof is completely analogous to the proof of Proposition 3.3.17. The only differences are the bound on the smallest endomorphisms, Proposition 1.3.33, and the use of a reduction from MINIMAL-ONEEND to ONEEND, Proposition 3.3.15 (instead of MINIMAL-ISOGENY and Proposition 3.3.13). □

Remark 3.3.19. *As discussed in Remark 3.3.16, there is a heuristic algorithm to find chains of ℓ-isogenies of degree $O(p^{7/2})$, and an algorithm proven under **GRH** to find chains of degree polynomial in p. Hence, it should be possible to reduce ℓ-ISOGENYPATH$_\lambda$ to ℓ-ISOGENYPATH,* heuristically *when*

$$\lambda(\log p) > \frac{7}{2}\log(p)$$

and under **GRH** *when*

$$\lambda(\log p) > c\log(p),$$

for some constant c, which might be difficult to determine.

3.4 Worst-case to average-case reductions

The content of this section has been extracted from [HW25b, Section 7]. Our goal is to prove Theorem 3.4.2 which provides worst-case to average-case reductions between all the problems presented in Section 3.1. In this section, we have presented a set of hard problems, on which the security of isogeny-based cryptography relies, before proving their unconditional equivalence in Section 3.2. We recall that the current best algorithms take exponential time to solve any of them, see Proposition 3.1.3. Nevertheless, this does not guarantee that schemes relying on those problems are secure. The fact that any isogeny-based problem can be solved in exponential time, via a reduction to EndRing for instance, does not imply that all the problem instances are hard. In reality, some instances of EndRing are well-known to be easy, for example those given by the elliptic curves output by Proposition 1.3.40. This raises an important question:

- What if the instances encountered in cryptographic schemes fall into the set of easy problems?

Therefore, to ensure the security of isogeny-based cryptography, one needs to study the "real" cases of these hard problems — the ones found when running protocols. The security of cryptographic schemes typically relies on the hardness of random instances. These random instances follow mainly two specific distributions:

- Most of them follows the *stationary distribution*, which naturally appears from random walks in isogeny-graphs; see Section 1.5. The uniform distribution over supersingular elliptic curves might also appeared as a natural choice, both distributions are actually statistically indistinguishable.

- For schemes based on an EGA framework, the random instances are intended to follow a uniform distribution over the considered orbit. For REGA frameworks, the situation is more complex since the distribution depends on the choice of generating set.

A **worst-case to average-case reduction** is a reduction from a given problem, without any restriction on its instances — referred to as the **worst-case** of the problem — to another problem whose instances follow the above distributions — this is the **average-case**. The remainder of this section explores worst-case to average-case reductions for the first category of problems; the oriented case is left as an open question.

Let us formally define the average case for the hard problems we consider.

Definition 3.4.1. *Let P be a problem from the list ℓ-*IsogenyPath*,* Isogeny*,* EndRing*,* OneEnd*,* MOER*,* MaxOrder*,* $\text{MaxOrder}_{\mathbb{Q}}$ *and* HomModule*. The input of the problem P consists of one or two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$. We define the* average-case *for P as the case where the input curves are drawn from the stationary distribution on SS_p.*

As we have seen in Section 1.5, the stationary distribution is the one emerging from random walks in isogeny graphs. It appears to be the distribution of elliptic curves in a wide variety of schemes as early as Charles–Goren–Lauter hash function [CLG09], and up to the latest advances on the SQIsign digital signature scheme [DLRW24, BDD$^+$24]. Actually, in most of protocols, the usual method to generate a random supersingular elliptic curve over a finite field is to perform, from a known elliptic curve, a random walk in an isogeny-graph.

The main result of this subsection is Theorem 3.4.2, which states that for any pair of foundational problems (P, Q) introduced above, there is a reduction from P in the worst-case to Q in the average case. Thus, the instances encountered in isogeny-based schemes are guaranteed to be hard if there exists at least one hard instance for one of these problems. For some of these problems, self-reductions from worst-case to average-case are folklore. Our contribution is summarised in Figure 3.3 where we provide the most complete picture to date for understanding worst-case to average-case reductions with a minimal number of assumptions. It yields the global theorem Theorem 3.4.2.

Theorem 3.4.2. *For any pair of problems (P, Q) chosen from the problems* ISOGENY, ENDRING*,* ONEEND*,* MOER*,* MAXORDER*,* MAXORDER$_\mathcal{Q}$*,* HOMMODULE*, and* ℓ-ISOGENYPATH *there exists a probablistic polynomial time worst-case to average-case reduction from P to Q. All reductions hold unconditionally, with the two following exceptions which require the generalized Riemann hypothesis:*

- *if* $P = \ell$-ISOGENYPATH*, or*
- *if* $Q \in \{$MAXORDER, MAXORDER$_\mathcal{Q}\}$ *and* $p \equiv 1 \bmod 8$.

Reductions involving MAXORDER$_\mathcal{Q}$ *require oracle access to* $\mathcal{Q}$*.*

Map of the proof of Theorem 3.4.2

The strategy consists in proving that the worst-case ONEEND problem reduces to the average case of any other problem in the list. The precise network of reductions is summarized in Figure 3.3, page 79. We conclude from the fact, established in Theorem 3.2.1, that the worst case of any problem in the list reduces to a worst-case ONEEND instance (except for $P = \ell$-ISOGENYPATH, which relies on the conditional reduction of [Wes22b]).

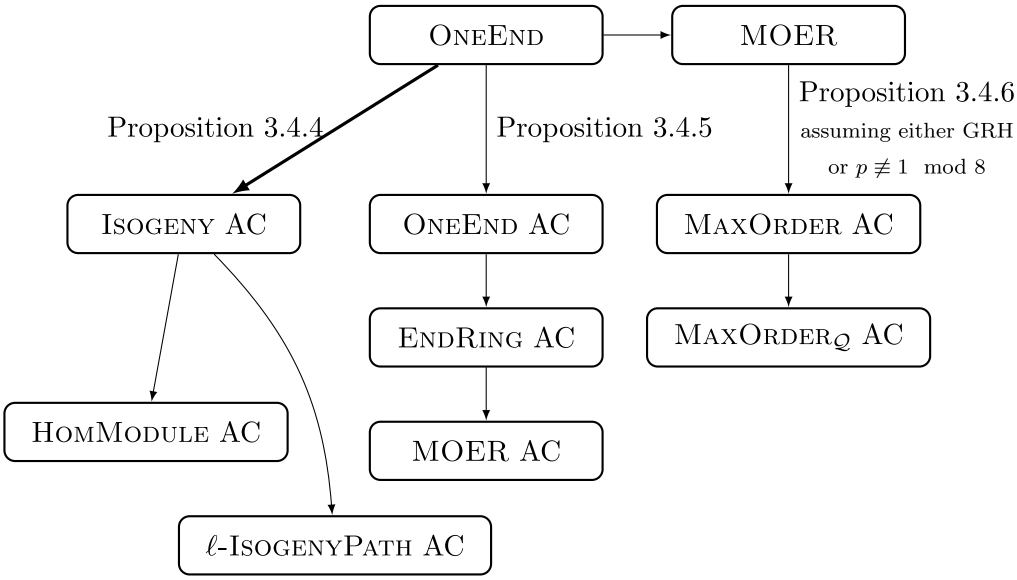


Figure 3.3: Summary of worst-case to average-case reductions. Each arrow represents a probabilistic polynomial time reduction. Let "AC" label means "average-case". All reductions are one-to-one except for the thick arrow, which requires on average fewer than 3 oracle calls. Each arrow is proved in the associated reference. Arrows without reference are trivial reductions.

Discussion of Theorem 3.4.2 and comparison with previous work.

Some of the worst-case to average-case reductions between the problems of interest are considered standard knowledge. It is well known, for instance, that random walks in ℓ-isogeny graphs can be used to re-randomize an instance of the ℓ-ISOGENYPATH, leading naturally to a self-reduction. This straightforward approach extends to other cases, but is not sufficient to obtain the full network of reductions proved in Theorem 3.4.2. For instance, the reduction from the worst-case ONEEND problem to the average case ISOGENY, HomModule or ℓ-IsogenyPath problems is obtained by modifying a worst-case reduction proposed in [PW24]. The reduction from the worst-case OneEnd problem to the average case MaxOrder or MaxOrder$_Q$ problems relies on recent advances facilitating the conversion between ideals and isogenies and the division of isogenies.

The straightforward reductions.

In Figure 3.3, every non-labeled arrow denotes a "trivial" reduction. To be more precise, these reductions simply forward an instance for a given problem to an instance for a (at least as hard) variant of this problem. In particular, the input of the average-case problems involved in those reductions always follows the same distribution. For example, the input distribution of the average-case Isogeny problem is a pair of two elliptic curves following the stationary distribution which is also the input distribution of the average-case HomModule. In addition, the solution we get for the harder problem directly includes a solution to the weaker one. For instance, solving the MOER problem also yields a solution to the corresponding EndRing instance. Therefore, all these reductions are trivial, and we only need to prove the reductions from worst-case OneEnd to the average-case problems to complete the figure. We shall address each proof of these non-trivial reductions in a dedicated subsection.

The proof of the main theorem.

We first prove the proof of Theorem 3.4.2, before demonstrating the remaining reductions.

Proof of Theorem 3.4.2. Let P and Q be two problems chosen from the problems ℓ-IsogenyPath, Isogeny, EndRing, OneEnd, MOER, MaxOrder, MaxOrder$_Q$, and HomModule. By Theorem 3.2.1, if P is not ℓ-IsogenyPath, we have a probabilistic polynomial time reduction from P in the worst-case to OneEnd in the worst-case. Otherwise, assuming the generalised Riemann hypothesis, there is a probabilistic polynomial time reduction from ℓ-IsogenyPath in the worst-case to OneEnd in the worst-case by [Wes22b] and [PW24]. Then using the results summarized in Figure 3.3, there is a probabilistic polynomial time reduction from OneEnd in the worst-case to Q in the average-case. □

OneEnd reduces to average-case Isogeny

As there exist solutions of Isogeny of arbitrarily large degree, we ease the analysis of the reduction by considering the Isogeny$_\lambda$ problem — introduced in [PW24] and discussed in Section 3.3 — where the degree of solutions is bounded.

In [PW24], the authors have proven that one can solve the OneEnd problem in expected polynomial time in $\log(p)$ and $\lambda(\log p)$ by calling on average at most 3 times an Isogeny$_\lambda$ oracle. We now adapt this reduction [PW24, Algorithm 6] to ensure that it produces a *semi average-case* Isogeny instance, in the sense that *at least one* of the elliptic curves involved follows a distribution indistinguishable from the stationary distribution. We then prove that the *semi* average-case Isogeny$_\lambda$ reduces to the average-case Isogeny$_\lambda$, and deduce the claimed expected polynomial time reduction from OneEnd to Isogeny$_\lambda$.

Since we are mainly tweaking the parameters of [PW24, Algorithm 6] to provide this worst-case to average-case reduction, we properly state it as Algorithm 6 to make the proof easier to follow.

Algorithm 6 Solving OneEnd given an Isogeny$_\lambda$ oracle [PW24, Algorithm 6]

```
    Input: A supersingular elliptic curve $E/\mathbb{F}_{p^2}$, a parameter $\varepsilon > 0$, an oracle
$\mathcal{O}_{\text{Isogeny}}$ solving the Isogeny$_\lambda$ problem.
    Output: An endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$.

$S \leftarrow$ an arbitrary nonzero point in $E[2]$
$n \leftarrow \lceil 2\log_3(p) - 4\log_3(\varepsilon) \rceil$
while True do
    $\varphi \leftarrow$ a non-backtracking random walk $\varphi : E \to E'$ of length $n$ in the 3-isogeny graph
    $\nu \leftarrow$ the isogeny $\nu : E' \to E''$ of kernel $\langle \varphi(S) \rangle$
    $\psi \leftarrow \mathcal{O}_{\text{Isogeny}}(E'', E)$, an isogeny $\psi : E'' \to E$
    $\alpha \leftarrow (\psi \circ \nu \circ \varphi)/2^e \in \mathrm{End}(E)$ for the largest possible $e$
    if $2 | \deg(\alpha)$ then return $\alpha$
```

Proposition 3.4.3. *Let $c_1, c_2 > 0$, and consider the following variant of [PW24, Algorithm 6] where*

- *the parameter ε is smaller than $1/p$,*
- *the length of the non-backtracking random walks in the 3-isogeny graph is n, where n satisfies $n \geq c_1 \log(p) - c_2 \log(\varepsilon)$.*

There exist absolute computable constants c_1 and c_2 such that this algorithm computes an endomorphism in expected polynomial time in $\log p$, $\lambda(\log p)$ and n with at most 3 calls to an Isogeny$_\lambda$ *oracle. In addition, these calls are done on* semi *average-case instances.*

Proof. The proof of correctness and the complexity analysis of [PW24, Algorithm 6] still apply to this variant, as we can set c_1 and c_2 such that n is larger than $\lceil 2\log_3(p) - 4\log_3(\varepsilon) \rceil$, as needed. In particular, for $p > 6$, the algorithm computes an endomorphism in expected polynomial time in $\log p$, $\lambda(\log p)$ and n with at most 3 calls to an Isogeny$_\lambda$ oracle.

We now prove the statement about the distribution of the instances given to the oracle. Let E be the OneEnd instance (a supersingular elliptic curve over $\mathbb{F}_{p^2}$). Let us denote by $\mathcal{O}_{\text{Isogeny}}$ the Isogeny$_\lambda$ oracle we have access to. Following [PW24, Algorithm 6], a point $S \in E[2]$ is fixed, and each call to the oracle $\mathcal{O}_{\text{Isogeny}}$ is done on an instance (E'', E) where E'' is the codomain of the composition of isogenies $\nu \circ \varphi$ where $\varphi : E \to E'$ is a non-backtracking random walk in the 3-isogeny graph of length n and $\nu : E' \to E''$ is an isogeny of kernel $\langle \varphi(S) \rangle$. As ν and φ have coprime degree, the distribution of E'' is the same as the codomain of $\varphi' \circ \nu'$ where $\nu' : E \to E/\langle S \rangle$, and $\varphi' : E/\langle S \rangle \to E''$ is a non-backtracking random walk in the 3-isogeny graph of length n. By [BCC$^+$23, Theorem 11], we can set c_1 and c_2 to ensure that for any $n \geq c_1 \log(p) - c_2 \log(\varepsilon)$, the distribution of E'' is at statistical distance at most ε from the stationary distribution. In particular, each call to the oracle $\mathcal{O}_{\text{Isogeny}}$ is done on *semi* average-case instances.

□

Proposition 3.4.4 (OneEnd reduces to average-case Isogeny)**.** *Solving an instance of the worst-case* OneEnd *problem can be reduced in expected polynomial time in $\log p$ and $\lambda(\log p)$ to solving 3 average instances of the* Isogeny$_\lambda$ *problem.*

Proof. First, we show that a *semi* average instance of Isogeny$_{\lambda_c}$ reduces to an average instance of Isogeny$_\lambda$, where $\lambda_c(n) := \lambda(n) + 2cn + 1$, with c the O-constant of Corollary 1.5.6. Note that c depends only on p and on some positive parameter ε, which we fix to be $\varepsilon := 1/p$. Let E_0 and E_1 be two supersingular elliptic curves defined over $\mathbb{F}_{p^2}$ such that E_1 is sampled from the stationary distribution. By Corollary 1.5.6, one can compute a random walk $\eta : E_0 \to E_2$ in the 2-isogeny graph of length $\lceil \tau(p, \varepsilon) \rceil$ such that the distribution followed by E_2 is at total variation distance at most ε to the stationary distribution. Then, as $\varepsilon = 1/p$, the two distributions are computationally indistiguishable.

Moreover, since $\tau(p, \varepsilon) = O(\log(p) - \log(\varepsilon))$ and $\varepsilon = 1/p$, we have that $\lceil \tau(p, \varepsilon) \rceil \leq 2c \log(p)+1$. Then, the random $2^{\lceil \tau(p,\varepsilon) \rceil}$-walk isogeny η verifies that $\log(\deg(\eta)) \leq 2c \log p + 1$.

A call to an Isogeny$_\lambda$ oracle on the pair (E_2, E_1), which is indistinguishable from an average-case instance, returns an isogeny $\varphi : E_2 \to E_1$ such that $\log \deg(\varphi) \leq \lambda(\log p)$. Then, the isogeny $\psi := \varphi \circ \eta : E_0 \to E_1$ verifies that $\log(\deg \psi) \leq 2c \log(p) + \lambda(\log p) + 1 = \lambda_c(\log(p))$. Thus the isogeny ψ is a solution to the initial Isogeny$_{\lambda_c}$ problem corresponding to (E_0, E_1). This concludes the proof that a *semi* average instance of Isogeny$_{\lambda_c}$ reduces to an average instance of Isogeny$_\lambda$.

We can now conclude the proof: by Proposition 3.4.3 and because $\lambda_c(\log p)$ is polynomial in $\lambda(\log p)$ and $\log p$, the OneEnd worst-case problem reduces in expected polynomial time in $\log(p)$ and $\lambda(\log(p))$ to 3 *semi* average instances of Isogeny$_{\lambda_c}$, which itself reduces to 3 average instances of Isogeny$_\lambda$ in polynomial time as proven above. □

OneEnd reduces to average-case OneEnd

The reduction presented in this subsection is analogous to the most folkoric methods for self-reducing the Isogeny problem from the worst-case to the average-case, leveraging the rapid mixing properties of isogeny graphs.

Proposition 3.4.5 (OneEnd reduces to average-case OneEnd). *Solving an instance of the worst-case* OneEnd *problem can be reduced to solving an average-case instance of the* OneEnd *problem in time polynomial in* $\log p$ *and in the length of the averace-case solution.*

Proof. Let E be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. Let $\eta : E \to E'$ be a random-walk in the 2-isogeny graph of length $n = \lceil \tau(p, 1/p) \rceil$, so that the distribution followed by E' is asymptotically indistinguishable from the stationary distribution by Corollary 1.5.6. Then, from a solution $\theta : E' \to E'$ to the average-case OneEnd instance corresponding to the curve E', one obtains a non trivial endomorphism $\hat{\eta} \circ \theta \circ \eta : E \to E$ which is a solution to the worst-case instance of OneEnd given by E. Indeed, this endomorphism is non trivial; otherwise there exists $n \in \mathbb{Z}$ such that $\hat{\eta} \circ \theta \circ \eta = n$ so $[\deg \eta] \circ \theta = n$, thus θ is a scalar endomorphism which is a contradiction. □

MOER reduces to average-case MaxOrder

The main challenge in proving unconditional reductions to the MaxOrder problem lies in leveraging the information obtained from the quaternion world to aid in isogeny computation, without having an access to a dictionnary between endomorphisms and quaternions. Indeed, we recall that when $p \equiv 1 \mod 8$, there is currently no known polynomial time algorithm free from GRH that can compute a supersingular elliptic curve defined over $\bar{\mathbb{F}}_p$ together with an embedding of its endomorphism ring into some quaternion algebra isomorphic to $B_{p,\infty}$. In Section 3.2, we address this difficulty by "locally" computing this embedding for sufficiently many primes, allowing us to apply the recent `IsogenyInterpolation` algorithm. Unfortunately, this method requires solving the MaxOrder problem for elliptic curves which are close to each other in the same isogeny graph. Thus, it cannot be turned into a reduction to the average-case MaxOrder problem. For this reason, the reduction presented below requires the construction of a curve E_0 for which a solution of MOER is known. This requires either $p \not\equiv 1 \mod 8$, or to assume GRH.

Proposition 3.4.6 (MOER reduces to average-case MaxOrder)**.** *An instance of the worst-case* MOER *can be reduced to an average instance of the* MaxOrder *problem in polynomial time in the length of the input. If* $p \equiv 1 \mod 8$*, this result assumes GRH.*

Proof. By Proposition 1.3.40, one can compute in polynomial-time a curve E_0 together with a quaternionic order $\mathcal{O}_0$ and an isomorphism $\varepsilon_0 : \mathcal{O}_0 \xrightarrow{\sim} \mathrm{End}(E_0)$ (i.e., a solution to MOER). This result assumes GRH in the case where $p \equiv 1 \mod 8$. We denote by B the quaternion algebra containing $\mathcal{O}_0$.

Let E be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. Let us solve the MOER problem for the elliptic curve E calling once a MaxOrder oracle on an average elliptic curve E'.

Let $N = \prod_{i=1}^{n} \ell_i$, where ℓ_i is the i-th smallest prime number, Let $\eta : E \to E'$ be a random N-walk. By Corollary 1.5.6, by choosing n such that $\log(N) \geq \tau(p, 1/p)$, one can ensure that E' follows a distribution statistically indistinguishable from the stationary distribution. In particular, as $\tau(p, 1/p) = O(\log p)$, by the prime number theorem, it is sufficient to consider primes up to some $\ell_n = O(\log(p))$. Thus, the computation of η takes a time polynomial in $\log p$.

Let us now solve the worst-case MOER instance corresponding to E from a solution $\mathcal{O}'$ to the average instance given by E'. Thanks to Proposition 1.3.38, one can assume that $\mathcal{O}'$ is a maximal order in the quaternion algebra B.

First, we compute a connecting ideal I between $\mathcal{O}'$ and $\mathcal{O}_0$, [KV10, Algorithm 3.5], and the corresponding isogeny using Proposition 2.3.3. By running [DLRW24, Algorithm 8], where the final division is done using Proposition 2.2.1, one obtains an isomorphism between $\mathrm{End}(E')$ and a maximal order $\mathcal{O}'$ in polynomial time. Then by using [Wes22b, Lemma 7.1] on the isogeny $\hat{\eta}$, one can compute, in polynomial time in $\log p$, the corresponding left $\mathcal{O}'$ ideal $I_{\hat{\eta}}$ such that $\mathcal{O}_R(I) = \mathcal{O}$. Hence, thanks again to [DLRW24, Algorithm 8], we obtain an explicit isomorphism between $\mathrm{End}(E)$ and a maximal order in B. □

Chapter 4

Oriented isogeny-based cryptography

In Section 4.2, we address the difficulty of solving the EndRing *problem given one non-trivial endomorphism. This is done by solving the* Primitivisation *problem in polynomial time and by providing a rigorous complexity analysis of the* $\mathfrak{O}$-Vectorisation *problem. Both results are part of the publication:*

[HW25a] Arthur Herlédan Le Merdy and Benjamin Wesolowski. The supersingular endomorphism ring problem given one endomorphism. IACR Communications in Cryptology*, 2(1), 2025.*

In Section 4.3, we present the ***PEGASIS*** *algorithm to compute group actions, which achieves practical running times. We mainly focus on the step of solving the* ***CLAPOTI*** *equation. This work has been published in:*

[DEF$^+$25a] Pierrick Dartois, Jonathan Komada Eriksen, Tako Boris Fouotsa, Arthur Herlédan Le Merdy, Riccardo Invernizzi, Damien Robert, Ryan Rueger, Frederik Vercauteren, and Benjamin Wesolowski. PEGASIS: Practical effective class group action using 4-dimensional isogenies. Cryptology ePrint Archive, Paper 2025/401, 2025. Accepted in the conference Advances in Cryptology – CRYPTO 2025 of the IACR.

The security of a group-action-based scheme relies on the difficulty of inverting group actions. This problem is called the Vectorisation problem, as formalised in Section 1.2. The best quantum algorithms to solve Vectorisation problems for generic groups (i.e. using only group operations and group actions) are based on the subexponential quantum algorithm due to Kuperberg [Kup05]. Nevertheless, once the group action is instantiated in a concrete setup, additional operations may be available reducing or breaking the security of the schemes. This is the same situation as for group-based cryptography where some choices of groups lead to classical polynomial attacks — even if Shoup's theorem guarantees classical exponential security for generic groups [Sho97].

Orientations appear to be a suitable method to systematically construct (restricted) effective group actions from supersingular elliptic curves. For a given order $\mathfrak{O}$, we call the problem of inverting the class group action on $\mathfrak{O}$-oriented elliptic curves the $\mathfrak{O}$-Vectorisation problem. While elliptic curves (with some exceptions) are considered to be good generic group candidates, it is not clear that oriented elliptic curves offer an interesting framework for cryptography based on group actions. The rich structure of oriented elliptic curves might imply some specific flaws which would make them insecure frameworks to perform cryptographic group actions. For instance, we have seen in Chapter 3 that knowing the endomorphism rings of elliptic curves provides a trapdoor to solve the Isogeny problem between these curves; is it also the case for $\mathfrak{O}$-Vectorisation? In Section 4.1, we recall the positive answer provided in the literature by summarising the state-of-the-art relations between **oriented** problems — problems involving orientations — and **foundational** problems — problems introduced in Chapter 3.

A natural question is then to study the EndRing problem in the context of oriented elliptic curves. Indeed, if orientations provide enough additional information to solve

EndRing efficiently, then $\mathfrak{O}$-Vectorisation is an easy problem. If that were the case, $\mathfrak{O}$-oriented elliptic curves would not be suitable for cryptographic applications. In Section 4.2, we answer by addressing a more general question

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

Under the generalised Riemann hypothesis, we prove Theorem 4.2.1 and Theorem 4.2.2 which state that solving EndRing, given this additional information, can be done in classical exponential time and quantum subexponential time. To the best of our knowledge, the asymptotical complexities we provide are as good as those of the best heuristic algorithms in the literature. A crucial step in our work is the reduction of this problem to the $\mathfrak{O}$-Vectorisation problem. This corresponds exactly to the Primitivisation problem — introduced in [ACL$^+$23] as a plausibly hard problem for quantum computers. We show that it can actually be solved in quantum polynomial time. We conclude this section by demonstrating in Section 4.2 how `IsogenyDivision` algorithm developed in Section 2.2 allows us to navigate efficiently in the volcano of oriented isogenies. In previous literature, it was only possible to take a limited number of steps in this graph because of the degrading quality of the representation of oriented isogenies. We then provide a direct application of this result by proving a polynomial time reduction from $(\mathbb{Z}+c\mathfrak{O})$-EndRing to $\mathfrak{O}$-EndRing, for c a smooth integer; prior to this work, c needed to be powersmooth [Wes22a, Theorem 5].

A natural next question concerns the practicality of cryptographic constructions based on oriented elliptic curves. As discussed in Section 1.4, the group action framework provided by `CSIDH` is promising for cryptographic applications. Nevertheless, it suffers from the limitation of being only a *restricted* effective group action (REGA). While this suffices for basic cryptographic protocols, such as key exchange, more advanced schemes require a non-restricted effective group action (EGA) to be properly deployed. To address this limitation, a solution based on precomputation was proposed in [BKV19], enabling the construction of an EGA from the REGA framework of `CSIDH`. The approach involves computing the structure of the ideal class group — this step has subexponential complexity in the size of the discriminant [HM89] — in order to efficiently find small representatives of the given ideals. Nevertheless, this method is difficult to scale due to the precomputation step. Subsequent papers [DFK$^+$23, CLP24, ABE$^+$24] have improved the efficiency of this precomputation through various optimisations and by carefully choosing a suitable order for the orientation. The highest level of security achieved by a concrete instantiation of such an EGA is equivalent to `CSIDH`-1536, as demonstrated by a proof-of-concept implementation of the `PEARL-SCALLOP` scheme. Unfortunately, this remains highly impractical — approximately 12 minutes per group action.

In Section 4.3, we introduce the original EGA framework called `PEGASIS`, based on `CLAPOTI`, addressing this issue. This is currently the construction that achieves the highest level of security in the literature while remaining practical.

4.1 Oriented problems

We begin by introducing the specific problems that oriented schemes, such as `CSIDH`, rely on, and their relation with isogeny-based cryptography in general. The results presented here mainly come from [Wes22a] — in which most of the relations have been proven — and from [EL24], which improves some of them. Most of the time, the orientations considered in the problems are primitive; this ensures that they induce group action suitable for cryptographic application by Proposition 1.4.4.

The Vectorisation family

Let us start by formally stating Vectorisation and Parallelisation for $\mathfrak{O}$-oriented problems. The latter translates directly into the oriented context as the $\mathfrak{O}$-Parallelisation problem.

Problem 4.1.1 ($\mathfrak{O}$-Parallelisation)**.** *Given $(E,\iota) \in \mathrm{SS}_p(\mathfrak{O})$, and its images $\mathfrak{a} \star (E,\iota)$ and $\mathfrak{b} \star (E,\iota)$ by two $\mathfrak{O}$-ideals $\mathfrak{a}$ and $\mathfrak{b}$, find the elliptic curve $\mathfrak{ab} \star (E,\iota)$.*

The former comes in two flavours in the literature. The exact analogue of Vectorisation is known as the Effective $\mathfrak{O}$-Vectorisation problem.

Problem 4.1.2 (Effective $\mathfrak{O}$-Vectorisation)**.** *Given (E,ι), $(E',\iota') \in \mathrm{SS}_p(\mathfrak{O})$ two primitively $\mathfrak{O}$-oriented elliptic curves, find an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $\mathfrak{a} \star (E,\iota) = (E',\iota')$.*

The $\mathfrak{O}$-Vectorisation problem, in turn, designates a variant where the orientation induced by the action is not required to be the orientation of the target curve.

Problem 4.1.3 ($\mathfrak{O}$-Vectorisation)**.** *Given (E,ι), $(E',\iota') \in \mathrm{SS}_p(\mathfrak{O})$ two primitively $\mathfrak{O}$-oriented elliptic curves, find an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $E^{\mathfrak{a}} \simeq E'$.*

Both versions are actually quantumly equivalent. While the reduction from $\mathfrak{O}$-Vectorisation to Effective $\mathfrak{O}$-Vectorisation is trivial, the reduction in the other direction relies on reductions involving the $\mathfrak{O}$-EndRing problem, an oriented variant of EndRing. We discuss them in the next subsection.

Notice that, in the literature, Effective $\mathfrak{O}$-Vectorisation is defined to take as input a third oriented curve $(F,\jmath)$ and requires an *efficient* representation of the action $\varphi_{\mathfrak{a}} : (F,\jmath) \to \mathfrak{a} \star (F,\jmath)$. This efficiency condition ensures that solutions can be used for cryptographic applications, for instance, to break a Diffie-Hellman key exchange. Nevertheless, thanks to the `CLAPOTI` algorithm, we are now able to remove this requirement without any trade-off. For instance, the reduction from breaking a Diffie-Hellman key exchange to solving the Effective $\mathfrak{O}$-Vectorisation problem still holds.

Proposition 4.1.4 ($\mathfrak{O}$-Parallelisation to Effective $\mathfrak{O}$-Vectorisation)**.** *The* $\mathfrak{O}$-Parallelisation *problem reduces to the* Effective $\mathfrak{O}$-Vectorisation *problem in probabilistic polynomial time in the length of the input.*

Proof. Let $(E,\iota), \mathfrak{a} \star (E,\iota), \mathfrak{b} \star (E,\iota)$ be three $\mathfrak{O}$-oriented elliptic curves. By solving the Effective $\mathfrak{O}$-Vectorisation problem for the curves (E,ι) and $\mathfrak{a} \star (E,\iota)$, one obtains an ideal $\mathfrak{c}$ equivalent to $\mathfrak{a}$. Then, by Corollary 2.3.2, one can compute the oriented elliptic curve $\mathfrak{c} \star (\mathfrak{b} \star (E,\iota)) = \mathfrak{ab} \star (E,\iota)$ in time polynomial in the length of the input. □

We introduce a final Vectorisation-like problem. In this version, we assume the knowledge of only one of the two orientations. This is presumably the hardest one.

Problem 4.1.5 ($\mathfrak{O}$-Uber)**.** *Given $(E,\iota) \in \mathrm{SS}_p(\mathfrak{O})$ and a primitively $\mathfrak{O}$-orientable elliptic curve E', find an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $E^{\mathfrak{a}} \simeq E'$.*

This problem comes from [DDF$^+$21], where the authors reduce the security of `SIDH`, `OSIDH` [CK20], `CSIDH`, `Séta` [DDF$^+$21] to this problem. Hence, it provides a very general framework to study the security of oriented and unoriented isogeny-based cryptography. Note that the first two schemes are broken, see Section 2.1 for `SIDH` and [DD22] for `OSIDH`. This problem was also introduced with an effective variant. Once again, thanks to `CLAPOTI`, this is no longer necessary.

The oriented EndRing family

We now introduce several variants of the EndRing problem involving orientations. These problems are useful to understand better the relations between oriented problems but also the connection with the foundational problems. As the latter have been shown to be all equivalent to each other in Chapter 3, it is sufficient to study the links with EndRing to obtain the global picture.

First, we consider the case where a primitive orientation on the elliptic curve is given. Since additional information is provided, this problem is clearly easier than EndRing.

Problem 4.1.6 ($\mathfrak{O}$-EndRing). *Given $(E, \iota) \in \mathrm{SS}_p(\mathfrak{O})$, find four endomorphisms generating $\mathrm{End}(E)$ as a $\mathbb{Z}$-module.*

Nevertheless, this problem offers an interesting point of view on the security of oriented isogeny-based cryptography as it is equivalent to $\mathfrak{O}$-Vectorisation, with some subtleties. The reduction to $\mathfrak{O}$-Vectorisation relies on the construction of a special $\mathfrak{O}$-oriented curve, i.e. an $\mathfrak{O}$-oriented elliptic curve with known endomorphism ring. By Proposition 1.3.40, we know that computing a special curve requires assuming **GRH** when $p \equiv 1 \mod 8$. Here, there is the additional requirement of finding a special curve that is primitively $\mathfrak{O}$-oriented. This is possible under **GRH**, when the factorisation of the discriminant order is known.

Lemma 4.1.7 ((**GRH**) [Wes22a][Lemma 4]). *Given a quadratic order $\mathfrak{O}$ together with the factorisation of its discriminant, one can find a primitively $\mathfrak{O}$-oriented elliptic curve together with a solution to its* MOER *problem in probabilistic polynomial time in* $\log(p)$ *and* $\log(|\operatorname{disc}(\mathfrak{O})|)$.

Once such a special primitively $\mathfrak{O}$-oriented curve (E_0, ι_0) has been computed, one can solve the $\mathfrak{O}$-Vectorisation problem between (E_0, ι_0) and any primitively $\mathfrak{O}$-orientable E to obtain a connecting ideal between the two curves maximal orders. In other words, we know a solution to the MOER instance of E_0 and a connecting ideal between the maximal orders of E and E_0. Since this is the same situation as in the end of the proof of Proposition 3.4.6, applying the same process, we solve the MOER instance given by E in polynomial time. This leads to the following proposition.[1]

Proposition 4.1.8 (**GRH**, Proposition 7 in [Wes22a]). *Given the factorisation of* $\operatorname{disc}(\mathfrak{O})$*, the $\mathfrak{O}$-EndRing problem reduces to $\mathfrak{O}$-Vectorisation*

The other direction is proved through the sequence of reductions

$$\mathfrak{O}\text{-Vectorisation} \longrightarrow \text{Effective } \mathfrak{O}\text{-Vectorisation} \longrightarrow \text{EndRing}(\mathfrak{O}).$$

Thanks to the recent result [EL24, Corollary 5], there is now an unconditional reduction from Effective $\mathfrak{O}$-Vectorisation to EndRing($\mathfrak{O}$). It is worth noticing that in this proof, the `CLAPOTI` algorithm is, once again, central to avoid heuristics and **GRH**. The previous result, [Wes22a, Theorem 2] was proven under **GRH** and assuming the factorisation of the order discriminant is known. Moreover, the complexity also depended on the size of the 2-torsion of the class group.

We shall see in the next section that, up to some integer factorisation, the $\mathfrak{O}$-EndRing problem is actually equivalent to computing the endomorphism ring of a curve given a

[1] The proof of [Wes22a, Proposition 7] actually concludes by returning a solution to MaxOrder (which is equivalent to EndRing).

non-trivial endomorphism. We call this problem, for an endomorphism $\alpha \in \mathrm{End}(E)$, the α-EndRing problem. However, as the α-EndRing problem does not directly extend to other curves than E, α being defined with respect to E, we state its definition using orientations.

Problem 4.1.9 (α-EndRing). *Given a supersingular elliptic curve E defined over $\mathbb{F}_{p^2}$ and a (not necessarily primitive) orientation $\iota : \mathbb{Z}[\alpha] \hookrightarrow \mathrm{End}(E)$, find four endomorphisms generating $\mathrm{End}(E)$ as a $\mathbb{Z}$-module.*

With this formulation, α is thus an element of a quadratic order $\mathcal{O}$ such that $\mathcal{O} \otimes_{\mathbb{Z}} \mathbb{Q} \simeq \mathrm{End}^0(E)$. This allows us to consider this problem for families of curves, while being equivalent to knowing one non-trivial endomorphism of the curve.

One direction of this equivalence is trivial. If an orientation is known, then the image of the order generator, which is a non-trivial endomorphism, is also known. For the other direction, given a non-trivial endomorphism α, one can find a suitable quadratic integer ω such that $\mathbb{Z}[\omega] \hookrightarrow \mathrm{End}(E), \omega \mapsto \alpha$ properly defines an embedding, by setting ω to be a root of the characteristic polynomial of α; as made explicit in Definition 1.3.10. Since the coefficients of this polynomial are the trace and the degree of the endomorphism α, which are computable in polynomial time (see Remark 1.3.32), finding ω is easy.

There is a trivial reduction from $\mathfrak{O}$-EndRing to α-EndRing. The other direction corresponds to solving the Primitivisation problem which was introduced in [ACL+23] as a presumably hard problem, even for quantum computers.

Problem 4.1.10 (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 $\mathfrak{O}$.*

We prove in Section 4.2 that this problem can actually be solved in classical probabilistic polynomial time plus the factorisation of an integer.

Remark 4.1.11. *We emphasise that α-EndRing, which asks for the endomorphism ring given one non-scalar endomorphism, and OneEnd, which asks for a single non-scalar endomorphism, are fundamentally different problems that should not be confused with each other.*

On the one hand, the OneEnd problem is equivalent to EndRing. However, this equivalence requires computing a polynomial number of endomorphisms. On the other hand, the α-EndRing problem is simply the EndRing problem with additional non-trivial information.

Hence, while α-EndRing reduces to OneEnd, the other direction does not hold. In fact, successfully reducing an OneEnd instance to an α-EndRing instance implies that one has managed to find a non-trivial endomorphism α. Thus, this reduction is morally equivalent to solving the OneEnd problem itself.

We also introduce a variant of EndRing for primitively $\mathfrak{O}$-orientable elliptic curves where the orientation is not provided.

Problem 4.1.12 ($\mathrm{EndRing}|_{\mathfrak{O}}$). *Given an $\mathfrak{O}$-orientable curve E, compute a basis of $\mathrm{End}(E)$.*

We have the following trivial sequence of reductions

$$\mathfrak{O}\text{-EndRing} \longrightarrow \mathrm{EndRing}|_{\mathfrak{O}} \longrightarrow \mathrm{EndRing}. \tag{4.1}$$

Nevertheless, the other way around is not so direct as it involves computing $\mathfrak{O}$-orientations. We call this problem of computing an $\mathfrak{O}$-orientation, the $\mathfrak{O}$-Orienteering problem.

Problem 4.1.13 ($\mathfrak{O}$-Orienteering). *Given a primitively $\mathfrak{O}$-orientable elliptic curve E defined over $\mathbb{F}_p^2$, compute an $\mathfrak{O}$-orientation over E.*

This problem has been studied in several papers, for instance [ACD$^+$24], where it appears to be a hard problem. At the current state of the art, even given the endomorphism ring of an $\mathfrak{O}$-orientable elliptic curve, finding an $\mathfrak{O}$-orientation in polynomial time is only feasible for orders $\mathfrak{O}$ with discriminant up to $O(p^{4/3})$. This bound comes from [EL24], where the authors improved the previous bounds given in [Wes22a, ACD$^+$24]. The $\mathfrak{O}$-Orienteering problem distinctly splits oriented problems into two distinct sets, as illustrated by the unidirectional reductions in Figure 4.1.

To conclude the discussion regarding the sequence (4.1), we observe that one can obtain a polynomial reduction from EndRing to $\mathfrak{O}$-EndRing whenever the two following conditions are verified:

- the number of primitively $\mathfrak{O}$-orientable elliptic curves over the total number of supersingular elliptic curves is $1/\operatorname{polylog}(p)$,
- the $\mathfrak{O}$-Orienteering problem can be solved in polynomial time and returns "`False`" when the curve is not orientable in polynomial time.

The idea is to perform $\operatorname{polylog}(p)$ random walks and solve the $\mathfrak{O}$-Orienteering problem for each of the landing curves, expecting to obtain at least one $\mathfrak{O}$-oriented elliptic curve. Then, one can solve the $\mathfrak{O}$-EndRing problem for this curve and transfer the knowledge of the endomorphism ring to the original curve.

The issue with this reduction is that having both conditions at the same time seems difficult; they are, in a way, mutually exclusive. On the one hand, the greater the discriminant of the order, the greater the number of curves orientable by this order. On the other hand, the difficulty to compute an orientation grows with the size of the discriminant.

This is why, in Figure 4.1, we simply indicate the trivial reduction from EndRing$|_{\mathfrak{O}}$ to $\mathfrak{O}$-EndRing when every supersingular elliptic curve defined over $\mathbb{F}_{p^2}$ is $\mathfrak{O}$-orientable.

Example 4.1.14. *Let us detail the particular case of elliptic curves oriented by the Frobenius, such as `CSIDH`. In this situation, the curve is (non necessarily primitively) orientable if and only if it is defined over $\mathbb{F}_p$, which is trivial to verify. Then, one can easily check if it is primitively oriented just by verifying if the curve is on the floor or on the surface; see [CD20, Remark 1]. Furthermore, the orientation is immediately given by the Frobenius. Nevertheless, among the $O(p)$ supersingular elliptic curves defined over $\mathbb{F}_{p^2}$ only $O(\sqrt{p})$ are defined over $\mathbb{F}_p$. Hence, reducing* EndRing *to* $\mathbb{Z}[\pi]$-EndRing *using random walks seems highly infeasible in polynomial time.*

We now introduce the last variant of EndRing which asks not only for a basis of the endomorphism ring but also for a primitive $\mathfrak{O}$-orientation. This problem, which is the hardest version of EndRing presented here, is equivalent to the $\mathfrak{O}$-Uber problem.

Problem 4.1.15 ($\mathfrak{O}$-EndRing*). *Given a primitively $\mathfrak{O}$-orientable curve E, compute a solution $\varepsilon : \mathcal{O} \to \operatorname{End}(E), (\theta)_{i=1}^4 \mapsto (\alpha)_{i=1}^4$ for the* MOER *problem of E, and an embedding $\jmath : \mathfrak{O} \hookrightarrow B_{p,\infty}$ such that $\varepsilon \circ \jmath$ is an $\mathfrak{O}$-orientation.*

Currently, this equivalence is only proven under **GRH** and given the factorisation of $\operatorname{disc}(\mathfrak{O})$.

These assumptions are necessary for the reduction from $\mathfrak{O}$-EndRing* to $\mathfrak{O}$-Uber, [Wes22a, Proposition 8]. This direction is actually similar to reducing $\mathfrak{O}$-EndRing to $\mathfrak{O}$-Vectorisation and, thus, also relies on the construction of a $\mathfrak{O}$-oriented special curve.

The other direction previously required the same assumptions, [Wes22a, Corollary 4], in addition to a runtime polynomial in the 2-torsion of the class group. Since it is analogous to reducing Effective $\mathfrak{O}$-Vectorisation to $\mathfrak{O}$-EndRing, one can remove these assumptions thanks to [EL24].

Proposition 4.1.16 ($\mathfrak{O}$-Uber to $\mathfrak{O}$-EndRing*)**.** *The $\mathfrak{O}$-Uber problem reduces to the $\mathfrak{O}$-EndRing* problem in probabilistic polynomial time in the length of the instance.*

Proof. Once the $\mathfrak{O}$-EndRing* problem is solved for the input $\mathfrak{O}$-orientable curve, we are exactly in the same situation as in the proof of [EL24, Corollary 5]. □

It is also worth noting that, while the reduction from EndRing$|_{\mathfrak{O}}$ to $\mathfrak{O}$-EndRing* is trivial, the other direction is proven only up to discriminants of order $O(p^{4/3})$, thanks again to the upper bound provided in [EL24].

Summary of relations

Let us state the global network of relations in Figure 4.1 The remainder of this chapter will focus on the bottom problems, the top part will not be discussed further.

4.2 The EndRing problem given one endomorphism

The material presented in this section comes from [HW25a]. The objective is to provide rigorous proofs for Theorem 4.2.1 and Theorem 4.2.2, which state the classical and quantum complexity of the α-EndRing problem.

Theorem 4.2.1 (**GRH**)**.** *There is a classical algorithm solving the α-EndRing problem in expected time $l^{O(1)}|\operatorname{disc}(\mathbb{Z}[\alpha])|^{1/4}$, where l is the length of the input.*

Theorem 4.2.2 (**GRH**)**.** *There is a quantum algorithm solving the α-EndRing problem in expected time $l^{O(1)}L_{|\operatorname{disc}(\mathbb{Z}[\alpha])|}[1/2]$, where l is the length of the input.*

We construct the corresponding algorithms by following a sequence of reductions implied by the discussion in Section 4.1:

$$\alpha\text{-EndRing} \xrightarrow{\text{Primitivisation}} \mathfrak{O}\text{-EndRing} \xrightarrow{\qquad} \mathfrak{O}\text{-Vectorisation}$$

We begin by computing a primitive orientation from the oriented elliptic curve given as input. This is done in the subsequent subsection by solving the Primitivisation problem in polynomial time plus the cost of factoring the discriminant of the order. We also present a direct application of this result: a polynomial algorithm to compute actions of smooth ideals. Then, in the next subsection, we construct rigorous classical and quantum algorithms to solve the $\mathfrak{O}$-Vectorisation problem. Combining both results with [Wes22a, Theorem 3], we obtain Theorem 4.2.1 and Theorem 4.2.2.

Primitivisation

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

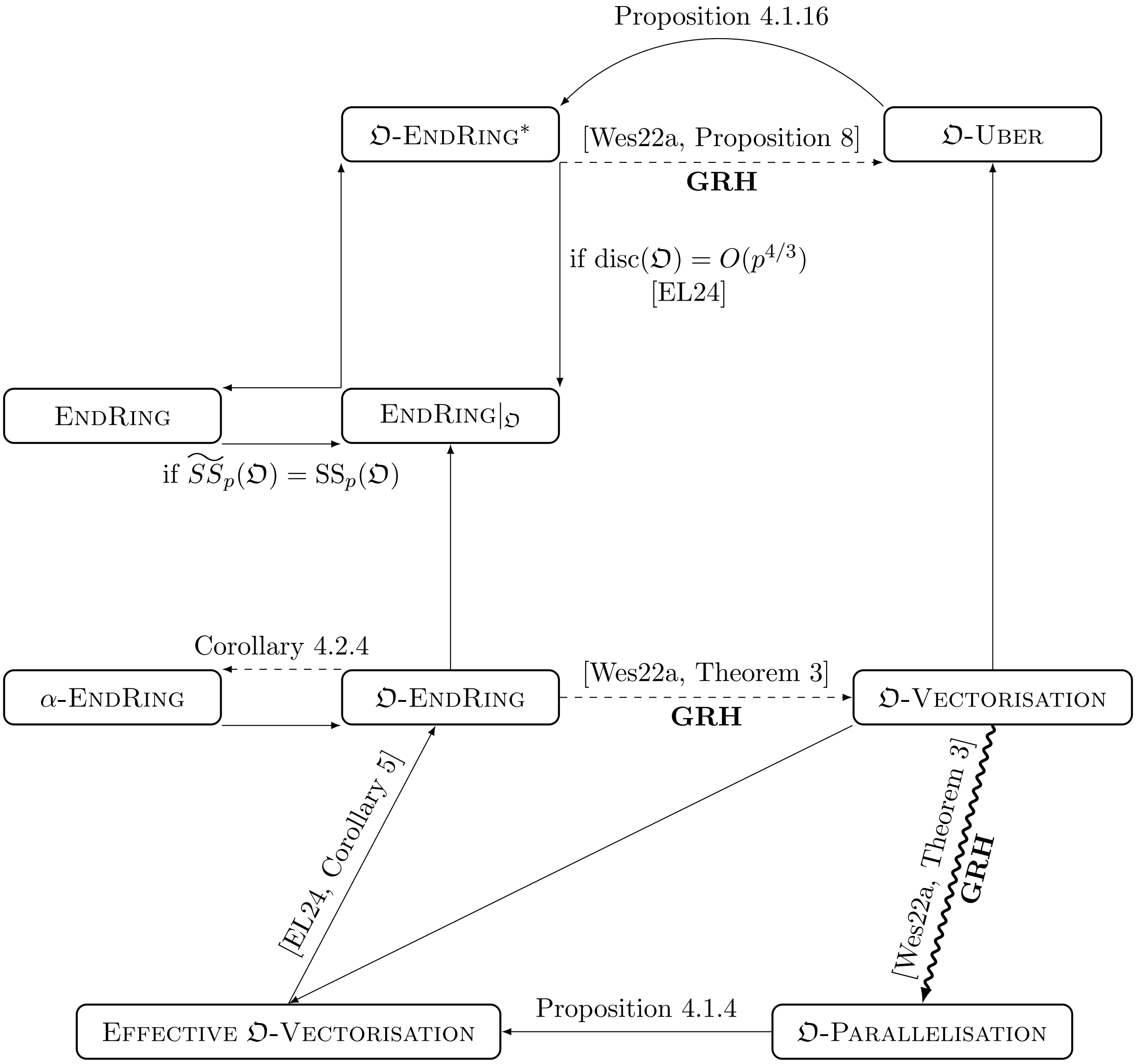


Figure 4.1: Summary of relations between oriented and fundamental problems. Each arrow is a probabilistic polynomial time reduction. The straight arrows are classical reductions and the snakes ones are quantum reductions. Thin arrows have a $O(1)$ query-complexity, thick ones have a polylog(p) query-complexity. Dashed arrows assume, depending on the context, the factorisation of disc$(\mathbb{Z}[\alpha])$ or disc$(\mathfrak{O})$. The arrow labeled **GRH** are reduction proven under the generalised Riemann hypothesis. Each arrow is proved in the associated reference. Arrows without reference are trivial reductions. This figure is based on [Wes22a, Figure 1].

containing $\mathbb{Z}[\pi]$. Therefore computing the endomorphism ring of an ordinary curve really is a case of the Primitivisation problem.

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

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

First, Theorem 4.2.3 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 4.2.4 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 4.2.3 (Primitivisation). *There exists an algorithm that takes as input:*

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

and returns a primitive orientation $\iota : \mathfrak{O} \hookrightarrow \mathrm{End}(E)$ such that $\mathbb{Z}[\theta] \subseteq \mathfrak{O}$. In particular, the order $\mathfrak{O}$ is generated by an element ω such that

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

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

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

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

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

Let $\mathfrak{O} \subseteq O_K$ be the largest order such that ι extends to an embedding $\mathfrak{O} \hookrightarrow \mathrm{End}(E)$. That embedding is the primitivisation of ι, so the algorithm aims at determining $\mathfrak{O}$. The inclusions $\mathbb{Z}[\alpha] \subseteq \mathfrak{O} \subseteq O_K$ suggest that $\mathfrak{O}$ can be determined by starting from $\mathbb{Z}[\alpha]$, and testing if the orientation can be extended locally at each prime factor of the conductor, as in the computation of the endomorphism ring of ordinary elliptic curves (see [Rob22b]). This is described in Algorithm 7.

Algorithm 7 PRIMITIVISATION

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

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

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

Let us prove that Algorithm 7 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 7 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 7.

Since $f_\alpha \leq 4N(\alpha) = 4N$, there are $O(\log(N))$ divisions using Algorithm 1. By Theorem 2.2.7, both checking if $\theta'/p_i \in \mathrm{End}(E)$ and getting a representation of the new endomorphism can be done in $O(\max(D, M^2)B^8 \log^2(N') \log(B))$ operations over $\mathbb{F}_{p^2}$ plus the cost of the computation of the bases $E[\ell^e]$ for each prime power divisor ℓ^e of N' plus the cost of the computation of $O(\log N')$ evaluations of θ' over these bases plus the cost of decomposing $N' - N$ as a sum of four squares.

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

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

□

Corollary 4.2.4 demonstrates that Primitivisation can be solved in polynomial time when the input is efficiently represented by applying Theorem 4.2.3.

Corollary 4.2.4. *There exists an algorithm that takes as input:*

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

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

Proof. As for Theorem 2.2.1, 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. □

A direct application: Computing the action of smooth ideals The class group action over $SS_{\mathfrak{O}}(p)$ is a key notion of the orientations' theory and is a central ingredient of the algorithms presented in Section 4.2 to solve the oriented problems we are interested in, such as the $\mathfrak{O}$-Vectorisation problem.

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

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

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

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

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

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

Corollary 4.2.4, primitivising this orientation can be done efficiently given the factorisation of the conductor of the order, i.e. the factorisation of $N(\mathfrak{a})$ here. This provides a new polynomial algorithm to compute action of smooth ideals. To compute the action of any ideal, we shall use the polynomial time `CLAPOTI` algorithm, Proposition 2.3.1.

Corollary 4.2.5. *Let (E, ι) be an $\mathfrak{O}$-oriented supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. Let $\mathfrak{a}$ be an invertible $\mathfrak{O}$-ideal of B-smooth norm. If (E, ι) is efficiently represented, then one can compute an efficient representation of $[\mathfrak{a}] \star (E, \iota)$ in time polynomial in B, $\log p$, $\log \mathrm{N}(\mathfrak{a})$ and in the length of the representation of ι.*

Proof. The prime factorisation $\ell_1^{e_1} \dots \ell_m^{e_m}$ of the norm of $\mathfrak{a}$ can be computed in time polynomial in B and in $\log \mathrm{N}(\mathfrak{a})$. From this factorisation, one deduces the decomposition of $\mathfrak{a}$ as a product of e_1 prime ideals of norm ℓ_1 with e_2 prime ideals of norm ℓ_2 and so on. Then, we can compute in time polynomial in B, $\log p$ and in $\log \mathrm{N}(\mathfrak{a})$ the isogeny $\varphi_{\mathfrak{a}}$, up to post-composition with isomorphisms, as a chain of isogenies with prime degree. Finally, one just has to primitivise the orientation $\varphi_{\mathfrak{a}} \circ \iota(\omega) \circ \hat{\varphi}_{\mathfrak{a}}$ using Corollary 4.2.4 to get an efficient representation of $\mathfrak{a} \star (E, \iota)$ in polynomial time. □

Remark 4.2.6. *We recall that in this section, as in Chapter 3, our objective is to develop* rigorous *algorithms to study the foundations of isogeny-based cryptography, not to provide practical algorithms. More precisely, our aim here is to present an algorithm solving* Primitivisation *and to show how it can be used to compute smooth ideal actions in polynomial time without relying on any heuristics. Notice that this application remains interesting, even after the introduction of `CLAPOTI`, since it exploits the smoothness of ideal norms.*

On the other hand, higher dimensional isogenies can also be used to practically compute group actions, at the cost of relying on heuristics. This direction is developed in Section 4.3.

Remark 4.2.7. *The first version of [HW25a], which appeared prior to [PR23], presented an algorithm for computing the action of any ideal with a rigorously proven complexity under **GRH**. This complexity matched the previous state of the art while removing all heuristic assumption that were previously required. The approach involved computing a smooth representative of the input ideal using algorithms with a subexponential complexity proven under **GRH**, such as [CJS14, Algorithm 1], followed by the application of Corollary 4.2.5.*

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

In this subsection, we analyse, under **GRH** only, the complexity of classical and quantum algorithms solving the $\mathfrak{O}$-Vectorisation problem. These algorithms are as good as the best algorithms in previous literature, which were based on heuristics. These reductions are key ingredients for solving the α-EndRing problem.

Classical algorithm Prior to this work, the best known complexity for a classical algorithm solving the $\mathfrak{O}$-Vectorisation problem was $l^{O(1)}|\operatorname{disc}(\mathfrak{O})|^{1/4}$, with l the length of the input; e.g. using a meet-in-the-middle approach as in [DG16]. Such complexity analyses were based on heuristics. Indeed, one needs to compute multiple actions of ideals to solve $\mathfrak{O}$-Vectorisation and without using higher dimensional isogenies to compute efficiently smooth ideal actions, one could only handle powersmooth ideals. Thus, one had to assume some heuristics about the distribution of powersmooth ideals. Thanks to Corollary 4.2.5, 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. We recall that $\mathfrak{O}$-Vectorisation and Effective $\mathfrak{O}$-Vectorisation are in fact both equivalent to $\mathfrak{O}$-EndRing, see Figure 4.1.

Algorithm 8 almost solves Effective $\mathfrak{O}$-Vectorisation — 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 8 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').

```
1:  $x \leftarrow \lceil \log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})| \rceil$.
2:  $\Sigma_x \leftarrow \{[\mathfrak{p}] \in \operatorname{Cl}(\mathfrak{O})$, such that $\gcd(f_{\mathfrak{O}}, \mathfrak{p}) = 1$ and $N(\mathfrak{p}) \leq x$ prime$\}$.
3:  $S_x \leftarrow \Sigma_x \cup \{[\mathfrak{p}]^{-1}$ for $[\mathfrak{p}] \in \Sigma_x\}$.
4:  $T[\texttt{enc}((E, \iota))] \leftarrow (1)$.
5:  while $\#T < \sqrt{\# \operatorname{Cl}(\mathfrak{O})}$ do
6:      $y \leftarrow \operatorname{Unif}\{y \in \mathbb{N}^{\# S_x}$ such that $||y||_1 = \lceil \log |\operatorname{disc} \mathfrak{O}| \rceil\}$.
7:      $\mathfrak{a} \leftarrow S_x^y$.
8:      if $T[\texttt{enc}(\mathfrak{a} \star (E, \iota))]$ is empty then
9:          $T[\texttt{enc}(\mathfrak{a} \star (E, \iota))] \leftarrow \mathfrak{a}$.
10: $\mathfrak{a} \leftarrow (1)$.
11: while $T[\texttt{enc}(\mathfrak{a} \star (E', \iota'))]$ is empty do
12:     $y \leftarrow \operatorname{Unif}\{y \in \mathbb{N}^{\# S_x}$ such that $||y||_1 = \lceil \log |\operatorname{disc} \mathfrak{O}| \rceil\}$.
13:     $\mathfrak{a} \leftarrow S_x^y$.
14: return $\bar{\mathfrak{a}} T[\texttt{enc}(\mathfrak{a} \star (E', \iota'))]$.
```

Lemma 4.2.8 (GRH). *Algorithm 8 runs in expected time* $l^{O_\varepsilon(1)} |\operatorname{disc}(\mathfrak{O})|^{1/4}$ *where* l *is the length of the input, and is correct.*

Proof. First of all, notice that using a dictionary structure for the table T, one can add and search for elements in time $O(\log \#T)$. From [DDF+21, Section 5.3], we have the estimate $\# \operatorname{Cl}(\mathfrak{O}) = O(\log(|\operatorname{disc}(\mathfrak{O})|)\sqrt{|\operatorname{disc}(\mathfrak{O})|})$. Then insertions and searches in the table T can be done in $O(\log |\operatorname{disc}(\mathfrak{O})|)$. Moreover, we use the `enc` function, see Section 1.4, to have a unique encoding of oriented elliptic curves. Finally, notice that, for $\operatorname{disc}(\mathfrak{O})$ large enough, we have that $\lceil \log |\operatorname{disc}(\mathfrak{O})| \rceil \geq C \frac{\log \# \operatorname{Cl}(\mathfrak{O})}{\log \log |\operatorname{disc}(\mathfrak{O})|}$, where C is the constant from Proposition 1.5.2. Thus Proposition 1.5.2 applies properly to the random walks given by the vectors y such that $||y||_1 = \lceil \log |\operatorname{disc}(\mathfrak{O})| \rceil$ at Step 6 and 12.

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

[5-9] It is expected that this first while loop will end after $O(\sqrt{\# \operatorname{Cl}(\mathfrak{O})})$ iterations. Indeed, by applying Proposition 1.5.2 to subset of vertices $H := \operatorname{Cl}(\mathfrak{O}) - T$, we get that the probability to get a new element for the table T is greater than $\frac{1}{2} - \frac{1}{\sqrt{\# \operatorname{Cl}(\mathfrak{O})}}$.

In particular, for $\operatorname{disc}(\mathfrak{O}) \geq 36$, one can expect to add a new element to the table T after at most 3 draws of random smooth ideals.

By Corollary 4.2.5, computing an efficient representation of $\mathfrak{a} \star (E, \iota)$ is done in polynomial time in l_1, in $\log p$ and in $\log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|$, where l_1 is the length of the representation of ι.

[11-14] This while loop is also expected to end after $O(\sqrt{\# \operatorname{Cl}((\mathfrak{O})})$ iterations, since, thanks again to Proposition 1.5.2, each iteration has a probability of success greater than $\frac{1}{2\sqrt{\# \operatorname{Cl}(\mathfrak{O})}}$. This time, we apply the proposition regarding the landing subset T.

Moreover, as in the first loop, using Corollary 4.2.5, one can compute the action of $\mathfrak{a}$ in time polynomial in l_2, $\log p$ and $\log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O})|$, where l_2 is the length of the representation of ι'.

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

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

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

Hence,

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

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

Remark 4.2.9. *Algorithm 8 needs space exponential in the length of the input. A space-efficient algorithm is conceivable using a Pollard-ρ approach, as it is used to find isogenies between ordinary elliptic curves in [BS12]. A rigorous analysis of such algorithms typically requires access to a random oracle, and we do not pursue this direction here.*

Algorithm 8 is a central subprocedure in our classical resolution of $\mathfrak{O}$-Vectorisation and α-EndRing, as well as Effective $\mathfrak{O}$-Vectorisation. These applications of Algorithm 8 require to move from one orbit to the other using the $\mathfrak{O}$-twists.

Theorem 4.2.10 (GRH, Effective $\mathfrak{O}$-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 a $\lceil \log^{2+\varepsilon} |\operatorname{disc}(\mathfrak{O}))| \rceil$-smooth ideal $\mathfrak{a}$ such that $\mathfrak{a} \star (E, \iota) = (E', \iota')$, in expected time $l^{O_\varepsilon(1)} |\operatorname{disc}(\mathfrak{O})|^{1/4}$ where l is the length of the input.*

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

By [ACL+24, Theorem 4.4], if p is ramified in K, then the action of $\operatorname{Cl}(\mathfrak{O})$ has only one orbit. Thus by running Algorithm 8 with the inputs $(E, \iota), (E', \iota')$ and ε, we get an $\mathfrak{O}$-ideal $\mathfrak{a}$ such that $\mathfrak{a} \star (E, \iota) \simeq (E', \iota')$.

Otherwise, if p is inert in K, again by [ACL+24, Theorem 4.4], the action of $\operatorname{Cl}(\mathfrak{O})$ has two orbits. We then run two instances of Algorithm 8 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. By

Proposition 1.4.5, we know that (E, ι) and its $\mathfrak{O}$-twist $(E, \bar{\iota})$ are not in the same orbit 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'). Since $\mathfrak{a}$ has been returned by Algorithm 8 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. 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`. □

Theorem 4.2.11 (**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 $E^{\mathfrak{a}} \simeq E'$ 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 1.4.4. Let O be the orbit of (E', ι'). From the same proposition, we know that (E, ι) or its $\mathfrak{O}$-twist $(E, \bar{\iota})$ is in O. Thus, by running two instances of Algorithm 8 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 4.2.8. □

From the previous results, we can now prove Theorem 4.2.1.

Proof of Theorem 4.2.1. Let E be a primitively $\mathfrak{O}$-orientable elliptic curve defined over $\mathbb{F}_{p^2}$ and $\iota : \mathbb{Z}[\alpha] \hookrightarrow \operatorname{End}(E)$ be an orientation of E such that $\mathbb{Z}[\alpha] \subseteq \mathfrak{O}$. Let us prove that the computation of the endomorphism ring $\operatorname{End}(E)$ 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 the length of $\operatorname{disc}(\mathbb{Z}[\alpha])$, see for instance [Pom87]. Then, by Corollary 4.2.4, 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 $\operatorname{End}(E)$ 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 4.1.8. Finally, by Theorem 4.2.11 and since $|\operatorname{disc}(\mathbb{Z}[\alpha])|$ is greater than $|\operatorname{disc}(\mathfrak{O})|$, the $\mathfrak{O}$-VECTORISATION problem can be solved in $l^{O(1)} |\operatorname{disc}(\mathbb{Z}[\alpha])|^{1/4}$. □

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.

In [Wes22a, Proposition 4], this method was used to obtain a heuristic algorithm with the same complexity as Lemma 4.2.14. As for the classical algorithm, the heuristic comes from the previous necessity of computing powersmooth representative of an ideal before acting by it.

Problem 4.2.12 (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 $s \in A$ verifying $f_1(x) = f_0(s + x)$ for any $x \in A$, find the element s called the shift hidden by f_0 and f_1.*

In this work, 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 Kuperberg's quantum algorithm to solve it in a subexponential number of queries of the black-box functions f_0 and f_1.

Theorem 4.2.13 (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 9 which solves $\mathfrak{O}$-Vectorisation assuming that the two input curves are in the same orbit. This algorithm is analogous to [CJS14, Algorithm 3].

Algorithm 9 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 $\mathfrak{a} \star (E_0, \iota_0) = (E_1, \iota_1)$

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 $j \in [\![1, \ldots, k]\!]$.
3: Solve the Hidden Shift problem instance given with the black-box functions, for
   $j \in \{0, 1\}$,

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

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

Lemma 4.2.14 (GRH). *The Algorithm 9 is correct and runs in $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$ expected time where l is the length of the input.*

Proof. Let us prove the complexity of Algorithm 9:

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

[3] By Kuperberg's algorithm, Theorem 4.2.13, one can solve the instance of the `Hidden Shift` problem with time and query complexity $L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$. All the queries are done on the function f_0 and f_1 which can be evaluated in time polynomial in the length of the input using `CLAPOTI`, see Corollary 2.3.2. Thus this step in done in $l^{O(1)} L_{|\operatorname{disc}(\mathfrak{O})|}[1/2]$, where l is the length of the input.

[4] To compute the reduced representative of the ideal class $[\mathfrak{b}_1^{s_1} \ldots \mathfrak{b}_k^{s_k}]$, 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 $[\mathfrak{b}_1^{s_1} \ldots \mathfrak{b}_k^{s_k}]$ from the reduced representatives of $[\mathfrak{b}_1^{s_1}]$,...,$[\mathfrak{b}_k^{s_k}]$ in time polynomial in $\log|\operatorname{disc}\mathfrak{O}|$. Hence, using the standard result $\# \mathrm{Cl}(\mathfrak{O}) = O(\log(|\operatorname{disc}(\mathfrak{O}))\sqrt{\operatorname{disc}(\mathfrak{O})})$, this whole step is done in time polynomial in $\log|\operatorname{disc}(\mathfrak{O})|$.

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

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

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

□

Theorem 4.2.15 (**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 $E_0^{\mathfrak{a}} \simeq E_1$.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 9. The first one with the inputs (E_0, ι_0) and (E_1, ι_1) and the second one with the inputs $(E_0, \bar{\iota_0})$ and (E_1, ι_1). When the action has 2 orbits (see Proposition 1.4.4), one of these instances corresponds to a scenario where there is no solution to the Hidden Shift problem. It can be assumed that running Algorithm 9 on this incorrect instance will either yield an erroneous output or fail to terminate. However, since we can check the validity of a solution in polynomial time using with `CLAPOTI`, this does not pose any issues. Consequently, the complexity in Theorem 4.2.15 directly follows from Lemma 4.2.14. □

This leads us to the following proof of Theorem 4.2.2.

Proof of Theorem 4.2.2. By Corollary 4.2.4 and because the factorisation of $\operatorname{disc}(\mathbb{Z}[\alpha])$ can be computed in quantum polynomial time, the α-EndRing problem reduces to $\mathfrak{O}$-EndRing in time polynomial in the length of the instance. Notice that the discriminant of the order returned by this primitivisation step can only decrease in absolute value. Then, by Proposition 4.1.8, α-EndRing reduces to $\mathfrak{O}$-Vectorisation in probabilistic time polynomial in the length of the input. Hence, by Theorem 4.2.15, α-EndRing can be solved in expected time $l^{O(1)} L_{|\operatorname{disc}\mathbb{Z}[\alpha]|}[1/2]$. □

Ascending the volcano

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

Adding K-orientations to an ℓ-isogeny graph of supersingular elliptic curves gives a structure of volcano to each of its connected components, analogous to the structure of isogeny graphs of ordinary elliptic curves. We introduce formally this notion before showing how the `IsogenyDivision` algorithm, presented in Section 2.2, can be used to navigate efficiently in this volcano. Then we provide an application of this result which improves [Wes22a, Theorem 5] by proving that $(\mathbb{Z}+c\mathfrak{O})$-EndRing reduces to $\mathfrak{O}$-EndRing in polynomial time in the largest prime factor of c (instead of it largest *prime power* factor).

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

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

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

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

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

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

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

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

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

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

Example 4.2.16. *In Section 1.4, we have defined the elliptic curves of the* CSIDH *framework — primitively oriented by $\mathbb{Z}[\sqrt{-p}]$ — to be on the floor and those from the* CSURF *framework — primitively oriented by $\mathbb{Z}[(\sqrt{-p}+1)/2]$ — to be on the surface. This correspond to the $\mathbb{Q}(\sqrt{-p})$-oriented 2-isogeny graph.*

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

Lemma 4.2.17 (Walking to the crater)**.** *Let $(E,\iota)\in SS_{\mathfrak{O}}(p)$ be an $\mathfrak{O}$-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.*

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

Let $\varphi_1:(E,\iota)\to(E_1,\iota_1)$ be the unique ascending isogeny of degree ℓ from (E,ι). We denote by $\mathfrak{O}_1$ the order such that (E_1,ι_1) is $\mathfrak{O}_1$-primitively oriented and $\mathfrak{O}$ is a suborder of $\mathfrak{O}_1$. Let ω_1 be a generator of $\mathfrak{O}_1$. We assume, without loss of generality, that $\mathfrak{O}$ is given by a generator ω of the form $\omega=\ell\omega_1$. Then as shown in [Wes22a, Lemma 11], $\ker\varphi=\ker(\iota(\omega))\cap E[\ell]$. As $\iota(\omega)$ is efficiently represented, $\ker\varphi$ can be computed in time polynomial in $\ell,\log p$ and l_0, where l_0 is the length of the representation of ι. One just has to compute a basis of $E[\ell]$ and to take a generator of the cyclic subgroup of $E[\ell]$ that vanishes under $\iota(\omega)$. It provides a representation of φ given by its kernel generated by a point living in an extension of degree $O(\ell^2)$. Thus, it is possible to compute the elliptic curve $E_1 = E/\ker\varphi$ and its orientation ι_1 induced by φ_1 in time polynomial in $\ell, \log p$ and in l_0.

On the one hand, to recover E_1, we use Vélu's formula. 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 2.2.1, 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 4.2.18. *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.

The first step if to compute the unique isogeny $\varphi : E \to E'$ of degree c such that $\varphi_*(\iota)$ is a primitive $\mathfrak{O}$-orientation. We start by computing 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 4.2.17, 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 4.2.17, every φ_i is given by e_i successive kernels of ℓ_i-isogenies living in extension of degree $O(P^+(c)^2)$. Then, one can compute an efficient representation of the primitively $\mathfrak{O}$-oriented elliptic curve $(E', \varphi_*(\iota))$ in time polynomial in $P^+(c)$ and in the length of the input by `IsogenyDivision` Theorem 2.2.7.

The $\mathfrak{O}$-EndRing oracle provides a basis for $\mathrm{End}(E')$ given the oriented curve $(E', \varphi_*(\iota))$. By Proposition 3.2.9, we can compute a solution to the MOER instance given by E' in polynomial time. Then to recover the endomorphism ring of E in polynomial time in the length of the input and in $P^+(c)$, we follow the same steps as at the end of the proof of Proposition 3.4.6 . In brief, we compute the ideal corresponding to φ. This ideal connects the maximal orders isomorphic to $\mathrm{End}(E)$ and $\mathrm{End}(E')$. Thus, we can compute a basis of $\mathrm{End}(E)$ using [DLRW24, Algorithm 8] and the `IsogenyDivision` algorithm. □

Corollary 4.2.19. *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 4.2.18 together with the complexity result on $\mathfrak{O}$-EndRing given by Theorem 4.2.1. □

Remark 4.2.20. *In [HW25a], Theorem 4.2.18 and Corollary 4.2.19 are actually proven under* ***GRH****. Here, thanks to the original unconditional reductions between hard problems, we provide these results without any assumption.*

4.3 A practical Effective Group Action from orientations

In this section, we present `PEGASIS` from [DEF+25a] — a practical framework for computing *effective group actions* which are suitable for post-quantum cryptography. We introduce material taken from the article with a point of view centred on the resolution of the `CLAPOTI` equation. By suitable for post-quantum cryptography, we mean that the group action is presumed difficult to invert even for a quantum computer. Then, it is possible to do post-quantum cryptography based on this group action following Section 1.2.

As Table 4.1 suggests, `PEGASIS` is a promising practical EGA based on oriented elliptic curves, achieving the highest level of security of the literature. Its proof-of-concept implementation in Sage outperforms the previously introduced REGAs implemented in C++, even at their lowest level of security. Moreover, it is roughly as fast as the Rust version of `KLaPoTi`; this comparison is more meaningful as they are both EGAs. Therefore, we expect `PEGASIS` to outperform `KLaPoTi`, once implemented at a low level in Rust or C++. However, it is important to emphasise that the goal of the paper [PPS24], which introduces `KLaPoTi`, is not to provide an optimised implementation of group actions, but rather to establish a polynomial-time asymptotically scalable algorithm.

As with all isogeny-based group actions, `PEGASIS` instantiates the action of an ideal class group on a set of elliptic curves. The core idea here is to optimise the setup in which to solve `CLAPOTI` Equation (2.5). In particular, this involves considering the action of an ideal class group over a set of oriented supersingular elliptic curves. On the one hand, supersingularity offers greater efficiency than ordinarity, due in part to improved control over the accessible torsion; see Section 1.4 for more details. On the other hand, the necessity to compute orientations and to carry them through successive actions can be very costly. Hopefully, by adopting a `CSIDH`-like setup, it is possible to exploit the advantages of supersingularity while avoiding the drawbacks of orientations. By doing so, with additional optimisations, `PEGASIS` can run by performing only computations over $\mathbb{F}_p$ without pushing forward any orientation.

In the rest of this section, we shall provide further details on the method developed to solve Equation (2.5) and how it is used to compute the group actions. Naturally, making `PEGASIS` practical extends beyond these steps. Nevertheless, we have chosen to focus the discussion on the aspects of this collective work in which we were most involved. Moreover, we directly present this algorithm with its specification to a `CSIDH`-like framework, the one used to obtain Table 4.1, rather than providing the most general construction. It is worth noting that most of the algorithmic steps presented do not depend on these specific choices of parameters. However, since it is not necessary here, we do not address the computation of orientations, which might be central to other setups.

Paper	Language	Time (s)				
		Approx. prime size (bits)				
		500	1000	1500	2000	4000
SCALLOP [DFK$^+$23]	`C++`	35	750			
SCALLOP-HD [CLP24]	Sage	88	1140			
PEARL-SCALLOP [ABE$^+$24]	`C++`	30	58	710		
`KLaPoTi` [PPS24]	Sage	207				
	Rust	1.95				
`PEGASIS`	Sage	1.53	4.21	10.5	21.3	121

Table 4.1: Comparison of `PEGASIS` and other effective group actions in the literature. Time different implementations take to evaluate one group action at different (rounded) prime sizes, measured in wall-clock seconds on the following hardware setups: Intel i5-6440HQ processor clocked at 3.5 GHz for `SCALLOP`, Intel Alder Lake CPU core clocked at 2.1 GHz for `SCALLOP-HD`, Intel i5-1038NG7 processor clocked at 2.0 GHz for `PEARL-SCALLOP`, and Intel Core i5-1235U at 4.0 GHz for `KLaPoTi` and `PEGASIS`. The table comes from [DEF$^+$25a].

`CSURF` in higher dimension

As previously suggested, for the sake of effectiveness, we consider a `CSIDH`-like setup. More precisely, our framework is similar to the one used in `CSURF` — being on the surface of the oriented volcano allows us to do all computations over $\mathbb{F}_p$, see [DEF$^+$25a] for more details on this aspect.

Let us formally state the `PEGASIS` instantiation, specialised to a `CSURF`-setup with parameter p. As always, p denotes the characteristic of the field over which the elliptic curves are defined. It also determines the group acting in `PEGASIS`. Hence, it is a crucial parameter for the cryptanalysis of any scheme based on this group action.

`PEGASIS`' setting. For a prime p congruent to 7 modulo 8 such that $p = c2^f - 1$ with c small:

- **The group** is the class group $\mathrm{Cl}(\mathfrak{O})$, where $\mathfrak{O}$ is the order $\mathbb{Z}[\frac{1+\sqrt{-p}}{2}]$ in the imaginary quadratic field $K := \mathbb{Q}(\sqrt{-p})$,
- **The set** is the set of primitively $\mathfrak{O}$-oriented elliptic curves defined over $\mathbb{F}_p$ up to K-isomorphism, denoted $\mathrm{SS}_{\mathbb{F}_p}(\mathfrak{O})$,
- **The action** is $\star : \mathrm{Cl}(\mathfrak{O}) \times \mathrm{SS}_{\mathbb{F}_p}(\mathfrak{O}) \to \mathrm{SS}_{\mathbb{F}_p}(\mathfrak{O})$ defined as in Section 1.4,
- **The public element** is an elliptic curve E_0 in $\mathrm{SS}_{\mathbb{F}_p}(\mathfrak{O})$, which is $\mathfrak{O}$-oriented by the mapping $\sqrt{-p} \mapsto \pi$,

For practicality, we provide a definition of a tuple which provides enough data to characterise an elliptic curve in a `PEGASIS` setting.

Definition 4.3.1. *A triplet (E, p, f) is called a **`PEGASIS`-triplet** if*

- *p is a prime congruent to 7 modulo 8,*

- *E is a supersingular elliptic curve defined over $\mathbb{F}_p$ such that* $\mathrm{End}_{\mathbb{F}_p}(E) = \mathbb{Z}[(\sqrt{-p}+1)/2]$,
- *f is a positive integer such that 2^f is the largest power of two dividing* $p+1$.

As seen in Section 1.4, any oriented elliptic curve obtained by successive actions on E_0 is oriented by the mapping $\sqrt{-p} \mapsto \pi$. Hence, all elliptic curves considered in this framework are oriented in the same way. Therefore, from now on, the orientations on the curves are implicit. Moreover, by abuse of notation, we write ι for the orientation mapping $\sqrt{-p} \mapsto \pi$, regardless of the curve considered.

Note that p does not have the same form as the primes used in CSURF. For instance in CSURF-512, the integer $p+1$ is a product of small primes lower than 389. The reason is that the accessible torsion we need to perform the group action computations is not the same. In a CSIDH-like scheme, $p+1$ needs to be smooth to guarantee the existence of many points of smooth order defined over $\mathbb{F}_p$. Here, $p+1$ needs to be divisible by a large power of 2 to have efficient higher dimensional machinery, as we shall see in this section. (We recall that $p+1$ is the number of $\mathbb{F}_p$-rational points of supersingular elliptic curves.)

In order for $(\mathrm{Cl}(\mathfrak{O}), \mathrm{SS}_{\mathbb{F}_p}(\mathfrak{O}), \star)$ to be an *effective* group action, it needs to admit specific polynomial time algorithms. Most of them concern only computations in the ideal class group; *group membership testing*, *equality testing*, *sampling*, *operating*, *inverting.* We are not suggesting any new algorithms for those procedures as they are already well enough studied to be efficiently used here. Then, there are the *representing*, *set membership testing* and *acting* algorithms. The first two can be chosen to be similar to CSURF, see Section 1.4, and the last one is the main contribution of PEGASIS.

The core of the PEGASIS algorithm is about efficiently computing the action of an arbitrary ideal class $[\mathfrak{a}]$ on an elliptic curve E in $\mathrm{SS}_{\mathbb{F}_p}(\mathfrak{O})$ by applying the CLAPOTI algorithm with practical optimisation. In particular, unlike in CSURF, the computation of group actions is not restricted to a small set of group elements anymore. In the remainder of this section, we focus on sharing the main ideas developed to solve the CLAPOTI equation efficiently. Since the proposed method requires slightly modifying the equation, we also need to detail how to exploit the obtained solution, as the CLAPOTI algorithm no longer applies directly.

Solving the CLAPOTI Equation

In order to solve Equation (2.5), instead of searching directly for ideals equivalent to $\mathfrak{a}$ verifying the norm condition, we fix two ideals $\mathfrak{b}, \mathfrak{c}$ in $[\mathfrak{a}]$, with coprime norms, and search for integers u, v and e such that

$$u\,\mathrm{N}(\mathfrak{b}) + v\,\mathrm{N}(\mathfrak{c}) = 2^e, \tag{4.2}$$

where $u, v > 0$ and $e < f - 2$.[2]

Notice that we fix the integer N from Equation (2.5) to be a power of two. This is motivated by the fact that the integer N determines the degree of the higher dimensional isogeny which embeds the action of $[\mathfrak{a}]$. To compute this isogeny, we decompose it into chains of lower degree isogenies and use generalised Vélu's formulas at each step, thus having an isogeny of degree a power 2 means that the degree of each isogeny in the chain is optimal.

Let us justify the conditions on u, v and e. First, there is simply no known method to exploit solutions of Equation (4.2) when $u, v \leq 0$, hence u, v are required to be positive.

[2] In [PR23], the authors already suggest this approach, while keeping the accessible torsion to be a generic integer N.

Then the integer e must be lower than $f-2$ to guarantee that 2^{e+2} divides $p+1$, so the subgroup of 2^{e+2}-torsion is accessible. We are asking for accessible 2^{e+2}-torsion, even if the kernel of the higher dimension isogeny is defined over the 2^e-torsion because additional torsion is necessary to properly use the theta model. We shall not detail this aspect, see [DEF$^+$25a] for more information.

There are two immediate concerns with the new equation:

(i) When does it admit solutions?

(ii) How to compute an ideal action from a given solution?

Remark 4.3.2. *Without loss of generality, we assume that the ideals $\mathfrak{a}$, $\mathfrak{b}$ and $\mathfrak{c}$ are* ***primitive****. An ideal $\mathfrak{a}$ is not primitive if there exists an ideal $\mathfrak{a}'$ and an integer m such that $m\mathfrak{a}' = \mathfrak{a}$. In particular $\mathfrak{a}'$ is equivalent to $\mathfrak{a}$ and has a smaller norm.*

Let us now answer points (i) and (ii).

(i) Ensuring existence of solutions

Let $\mathfrak{b}$ and $\mathfrak{c}$ be two fixed $\mathfrak{O}$-ideals equivalent to $\mathfrak{a}$ with coprime norms.

Tautologically, to ensure that there exist positive integers u and v verifying Equation (4.2), it is sufficient to prove that 2^e is greater than the largest positive integer not representable as $u\,\mathrm{N}(\mathfrak{a}) + v\,\mathrm{N}(\mathfrak{b})$, with $u, v > 0$. The problem of determining this largest integer is called the diophantine Frobenius problem. While the general problem is still an active field of research, our case is luckily solved, see [RA05] for instance. This largest integer is

$$(\mathrm{N}(\mathfrak{b}) - 1)(\mathrm{N}(\mathfrak{c}) - 1) - 1.$$

In particular, the smaller $\mathrm{N}(\mathfrak{b})\,\mathrm{N}(\mathfrak{c})$, the smaller e while guarantying the existence of solutions. On the one hand, this implies that testing ideals of ascending norm ensures that the exponent e is minimised. On the other hand, this has a negative impact on our framework. Indeed, as we can mainly hope for ideals of norm around $\sqrt{p}$, see Proposition 1.4.9, the value 2^e needs to be roughly greater than p to ensure existence of solutions. However, since 2^{e+2} divides $p+1$, we have that 2^e needs to be smaller than p. Thus, there are two necessary conditions which are difficult to satisfy at the same time. Being able to reduce the norm of $\mathfrak{b}$ and $\mathfrak{c}$ should improve the probability of existence of solutions.

In addition, we shall see that computing an action may be difficult depending on the form of u and v. Then, to find solutions of the most suitable form, one might want to obtain a set of solutions instead of a single one. To this end, it is also interesting to reduce as much as possible the norm of the ideals to not only guarantee existence of solutions but also to maximise their number.

To reduce ideal norms, each ideal is split into a smooth part and its remainder. By writing $\mathfrak{b}$ as $\mathfrak{b}_e\mathfrak{b}_k$ and $\mathfrak{c}$ as $\mathfrak{c}_e\mathfrak{c}_k$, where $\mathfrak{b}_e$ and $\mathfrak{c}_e$ have a smooth norm, we now have access to a new equation

$$u\,\mathrm{N}(\mathfrak{b}_k) + v\,\mathrm{N}(\mathfrak{c}_k) = 2^e, \tag{4.3}$$

which should have more solutions as the norms are smaller. Yet, it is not obvious how to recover the action of the ideal from Equation (4.3). Especially since this equation now differs from the `CLAPOTI` equation presented in Section 2.3 in two ways: there are additional factors u, v and the considered ideals are no longer equivalent to $\mathfrak{a}$. The fact that $\mathfrak{b}_k$ and $\mathfrak{c}_k$ are equivalent to $\mathfrak{a}$ up to a *smooth* factor is crucial to still compute the action. As we shall see, the idea of factoring out the smooth part of the norms can also be applied to the integers u and v to improve their chance to be of suitable forms.

Let us define formally this decomposition of ideals.

Definition 4.3.3. *Let $\mathfrak{B}$ be a set of primes. Let $\mathfrak{a}$ be an ideal.*

*We say that $\mathfrak{a}$ is **$\mathfrak{B}$-smooth** if its norm is only divisible by primes in $\mathfrak{B}$. Moreover, for two ideals $\mathfrak{a}_e$ and $\mathfrak{a}_k$ such that*

$$\mathfrak{a} = \mathfrak{a}_e \mathfrak{a}_k,$$

*where $\mathfrak{a}_e$ is $\mathfrak{B}$-smooth and $\mathrm{N}(\mathfrak{a}_k)$ is not divisible by any prime in $\mathfrak{B}$. We say that the ideal $\mathfrak{a}_e$ is the **$\mathfrak{B}$-smooth part** of $\mathfrak{a}$ and $\mathfrak{a}_k$ its remainder.*

Remark 4.3.4. *In the remainder of this section, the sets $\mathfrak{B}$ contain only primes splitting in $\mathbb{Q}(\sqrt{-p})$. This is the case in practice as we choose $\mathfrak{B}$ to be composed only of small primes. Then, they are not inert since they divide the norm of a primitive ideal and they are not ramified since they do not divide $-p$ which is the discriminant of $\mathbb{Q}(\sqrt{-p})$.*

(ii) Computing the action

Let us address how to handle solutions of Equation (4.3).

We assume that $\mathfrak{b} = \mathfrak{b}_k\mathfrak{b}_e$ and $\mathfrak{c} = \mathfrak{c}_k\mathfrak{c}_e$ are two ideals equivalent to $\mathfrak{a}$, with coprime norms, such that $\mathfrak{b}_e$ (resp. $\mathfrak{c}_e$) is the $\mathfrak{B}$-smooth part of $\mathfrak{b}$ (resp. $\mathfrak{c}$) — this set $\mathfrak{B}$ should be chosen to ensure optimal effectiveness. In addition, there exist $u, v > 0$ and $e < f - 2$ such that Equation (4.3) is verified.

The difficulty in adapting CLAPOTI depends on the form of the given solution (u, v). We consider four different cases, u and v can be:

1. **$\mathfrak{B}$-perfect**: perfect squares up to $\mathfrak{B}$-smooth factors,
2. **$\mathfrak{B}$-good**: sum of two squares up to $\mathfrak{B}$-smooth factors,
3. **$\mathfrak{B}$-semigood**: almost sum of two squares up to $\mathfrak{B}$-smooth factors,
4. **$\mathfrak{B}$-arbitrary**: no special structure after factoring out $\mathfrak{B}$-smooth parts.

Before formally defining and studying each case, let us state some facts. This list is ordered from the most restrictive kind of solutions to the most general ones, thus we have the following sequence of implications

$$\mathfrak{B}\text{-perfect} \implies \mathfrak{B}\text{-good} \implies \mathfrak{B}\text{-semigood} \implies \mathfrak{B}\text{-arbitrary}.$$

However, the computational complexity increases as it gains in generality. Note that for the sake of simplicity, only the situations where the integers u and v have the same form are considered in this work. Even though having integers u and v of two different forms is not expected to pose a problem.

We now study each form, one after the other, following the same steps. First, we define it formally. Then we prove that it is possible to compute the action from such solutions and we provide a complexity analysis. Finally, we discuss the probability to encounter those solutions and the possible advantages or drawbacks.

Since the presumably most costly steps are the computations of higher dimensional isogenies, we shall express the total complexities by listing these higher dimensional computations while ensuring that the other steps remain polynomial. Hence, one can gain insight into the complexity difference between two situations by examining the number of higher dimensional isogenies involved, as well as their dimension. However, note that the remaining polynomial part is not negligible and should also be further investigated.

Before discussing the simplest case, let us prove a central proposition to recover the action of $\mathfrak{b}_k$ and $\mathfrak{c}_k$ from the accessible actions of $\mathfrak{b}_e$ and $\mathfrak{c}_e$; accessible in the sense that they are computable efficiently, for instance using Corollary 4.2.5. To this end, we first prove a lemma to identify an isogeny from some coprimality and divisibility properties.

Lemma 4.3.5. *Let $\psi : E_1 \to E_2$, $\varphi : E_2 \to E_3$ be two separable isogenies between elliptic curves and n be an integer. If $[n]$ divides the composition $\varphi \circ \psi$, $\deg \varphi = n^2$ and $(\deg \psi, n) = 1$ then $\varphi = [n] \circ \lambda$ for some isomorphism $\lambda : E_2 \to E_3$.*

Proof. By definition, if $[n]$ divides $\varphi \circ \psi$, then there exists an isogeny $\phi : E_1 \to E_3$ such that

$$[n] \circ \phi = \varphi \circ \psi.$$

By duality, we have

$$\hat{\phi} \circ [n] = \hat{\psi} \circ \hat{\varphi}.$$

This implies that the group of n-torsion $E_3[n]$ is a subgroup of the kernel of $\hat{\psi} \circ \hat{\varphi}$. More precisely, by coprimality between n and $\deg \psi$, we have the inclusion

$$E_3[n] \subseteq \ker \hat{\varphi}.$$

Then, as φ is separable and $\deg(\varphi) = n^2$, the n-torsion subgroup and the kernel of $\hat{\varphi}$ have the same cardinality. Thus, $E_3[n]$ is equal to $\ker \hat{\varphi}$. Hence, by Proposition 1.3.12, there exists an isomorphism $\lambda : E_2 \to E_3$ such that $\hat{\varphi} = \hat{\lambda} \circ [n]$ and $\varphi = [n] \circ \lambda$.

□

Proposition 4.3.6. *Let $\mathfrak{B}$ be a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$ and (E, p, f) be a* `PEGASIS`*-triplet. Let $\mathfrak{b} = \mathfrak{b}_e\mathfrak{b}_k$ and $\mathfrak{c} = \mathfrak{c}_e\mathfrak{c}_k$ be two equivalent $\mathbb{Z}[(\sqrt{-p}+1)/2]$-ideals of coprime norms where $\mathfrak{b}_e$ (resp. $\mathfrak{c}_e$) are the $\mathfrak{B}$-smooth factors of $\mathfrak{b}$ (resp. $\mathfrak{c}$). For any generator α of the principal ideal $\mathfrak{b}\bar{\mathfrak{c}}$, there exists an isomorphism λ such that*

$$\varphi_{\mathfrak{c}_e} \circ \iota(\alpha) \circ \hat{\varphi}_{\mathfrak{b}_e} = [\mathrm{N}(\mathfrak{c}_e\mathfrak{b}_e)] \circ \lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ \varphi_{\mathfrak{b}_k}.$$

Proof. We consider the chain of isogenies

$$E \xrightarrow{\varphi_{E,\mathfrak{b}_e}} E^{\mathfrak{b}_e} \xrightarrow{\varphi_{E^{\mathfrak{b}_e},\mathfrak{b}_k}} E_1 \xrightarrow{\varphi_{E_1,\bar{\mathfrak{c}}_k}} E_2 \xrightarrow{\varphi_{E_2,\bar{\mathfrak{c}}_e}} E_3.$$

By Proposition 1.4.7, their composition has kernel $E[\mathfrak{b}\bar{\mathfrak{c}}]$. As the kernel of the endomorphism $\iota(\alpha)$ is also $E[\mathfrak{b}\bar{\mathfrak{c}}]$, by Proposition 1.3.12, there exists an isomorphism $\eta : E_3 \to E$ such that

$$\eta \circ \varphi_{E_2,\bar{\mathfrak{c}}_e} \circ \varphi_{E_1,\bar{\mathfrak{c}}_k} \circ \varphi_{E^{\mathfrak{b}_e},\mathfrak{b}_k} \circ \varphi_{E,\mathfrak{b}_e} = \iota(\alpha). \tag{4.4}$$

The endomorphism $[\mathrm{N}(\mathfrak{c}_e)]$ divides the isogeny $\varphi_{E,\mathfrak{c}_e} \circ \iota(\alpha)$, as it divides its dual. Indeed, again by equality of kernels, there exists an isomorphism μ such that

$$\widehat{\iota(\alpha)} = \mu \circ \varphi_{E^{\mathfrak{b}_e},\bar{\mathfrak{b}}\mathfrak{c}_k} \circ \varphi_{E,\mathfrak{c}_e}.$$

Then, we have the following explicit division by $[\mathrm{N}(\mathfrak{c}_e)]$

$$\begin{aligned}\widehat{\iota(\alpha)} \circ \hat{\varphi}_{E,\mathfrak{c}_e} &= \mu \circ \varphi_{E^{\mathfrak{c}_e},\mathfrak{c}_k\bar{\mathfrak{b}}} \circ \varphi_{E,\mathfrak{c}_e} \circ \hat{\varphi}_{E,\mathfrak{c}_e} \\ &= [\mathrm{N}(\mathfrak{c}_e)] \circ \mu \circ \varphi_{E^{\mathfrak{c}_e},\mathfrak{c}_k\bar{\mathfrak{b}}}.\end{aligned}$$

Then, Equation (4.4) implies that $[\mathrm{N}(\mathfrak{c}_e)]$ divides the composition of the isogenies

$$\varphi_{E,\mathfrak{c}_e} \circ \eta \circ \varphi_{E_2,\bar{\mathfrak{c}}_e} \circ \varphi_{E_1,\bar{\mathfrak{c}}_k} \circ \varphi_{E^{\mathfrak{b}_e},\mathfrak{b}_k} \circ \varphi_{E,\mathfrak{b}_e}.$$

Applying Lemma 4.3.5 to $\varphi = \varphi_{E,\mathfrak{c}_e} \circ \eta \circ \varphi_{E_2,\bar{\mathfrak{c}}_e}$, $\psi = \varphi_{E_1,\bar{\mathfrak{c}}_k} \circ \varphi_{E^{\mathfrak{b}_e},\mathfrak{b}_k} \circ \varphi_{E,\mathfrak{b}_e}$ and $n = \mathrm{N}(\mathfrak{c}_e)$, we deduce the existence of an isomorphism $\lambda : E_2 \to E^{\mathfrak{c}_E}$ such that

$$[\mathrm{N}(\mathfrak{c}_e)] \circ \lambda = \varphi_{E,\mathfrak{c}_e} \circ \eta \circ \varphi_{E_2,\bar{\mathfrak{c}}_e}.$$

Hence, we have

$$\varphi_{E,\mathfrak{c}_e} \circ \iota(\alpha) = [\mathrm{N}(\mathfrak{c}_e)] \circ \lambda \circ \varphi_{E_1,\bar{\mathfrak{c}}_k} \circ \varphi_{E^{\mathfrak{b}_e},\mathfrak{b}_k} \circ \varphi_{E,\mathfrak{b}_e}.$$

Composing on the right by the dual isogeny of $\varphi_{E,\mathfrak{b}_e}$, we obtain the claimed result. □

The $\mathfrak{B}$-perfect case. Let us start with the easiest situation, when (u, v) is $\mathfrak{B}$-perfect. This is the only time when the higher dimension computation only involves PPAV of dimension 2. Unfortunately, this is also the least probable situation.

Definition 4.3.7 ($\mathfrak{B}$-perfect)**.** *Let $\mathfrak{B}$ be a set of primes. We say that an integer $u \in \mathbb{Z}_{>0}$ is $\mathfrak{B}$-**perfect** if u can be written as $g_u x_u^2$ with $g_u, x_u \in \mathbb{Z}_{>0}$ such that g_u is a product of primes in $\mathfrak{B}$. A pair (u, v) is $\mathfrak{B}$-perfect if u and v are both $\mathfrak{B}$-perfect.*

Proposition 4.3.8. *Let $\mathfrak{B}$ be a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$. Let (E, p, f) be a PEGASIS triplet and $\mathfrak{a}$ be a $\mathbb{Z}[(\sqrt{-p}+1)/2]$-ideal. Given*

- *the elliptic curve $E/\mathbb{F}_p$,*
- *two ideals $\mathfrak{b} = \mathfrak{b}_e\mathfrak{b}_k$ and $\mathfrak{c} = \mathfrak{c}_e\mathfrak{c}_k$ of coprime norms equivalent to $\mathfrak{a}$ such that $\mathfrak{b}_e$ and $\mathfrak{c}_e$ are respectively the $\mathfrak{B}$-smooth parts of $\mathfrak{b}$ and $\mathfrak{c}$,*
- *an integer $e < f - 2$,*
- *a coprime $\mathfrak{B}$-perfect pair $(u, v) = (g_u x_u^2, g_v x_v^2)$ such that*

$$g_u x_u^2 \,\mathrm{N}(\mathfrak{b}_k) + g_v x_v^2 \,\mathrm{N}(\mathfrak{c}_k) = 2^e,$$

one can compute $j(E^{\mathfrak{a}})$ in time polynomial in $\max(\mathfrak{B})$ *and in the length of the input plus the cost of computing a 2^e-isogeny in dimension 2.*

Proof. This situation is almost the same as in CLAPOTI, except that the considered ideals are only equivalent to $\mathfrak{a}$ up to multiplication by ideals of smooth norm. There are also the smooth factors g_u (resp. g_v) of u (resp. v) to handle. Nevertheless, these changes significantly affect the Kani diagram used to compute the action of the ideal $\mathfrak{a}$. Let us define the isogenies involved in this new diagram.

We consider a generator α of the principal ideal $\mathfrak{b}\bar{\mathfrak{c}}$. Let λ be the isomorphism given by Proposition 4.3.6 such that the following chain of isogenies is equal to $\iota(\alpha)$

$$E \xrightarrow{\varphi_{\mathfrak{b}_e}} E^{\mathfrak{b}_e} \xrightarrow{\varphi_{\mathfrak{b}_k}} E_1 \xrightarrow{\lambda\circ\varphi_{\bar{\mathfrak{c}}_k}} E^{\mathfrak{c}_E} \xrightarrow{\hat{\varphi}_{\mathfrak{c}_e}} E.$$

We compute efficient representations of the isogenies $\varphi_{\mathfrak{b}_e}$ and $\varphi_{\mathfrak{c}_e}$ in time polynomial in $\log \mathrm{N}(\mathfrak{b}_e)$,$\log \mathrm{N}(\mathfrak{c}_e)$,$\log(p)$ and $\max(\mathfrak{B})$, using Corollary 4.2.5.

In addition, we compute two efficiently represented isogenies

$$\varphi_u : E^{\mathfrak{b}_e} \to E_u \text{ of degree } g_u$$

and

$$\varphi_v : E^{\mathfrak{c}_e} \to E_v \text{ of degree } g_v.$$

This can be done in time polynomial in $\log(u), \log(v), \log(p)$ and $\max(\mathfrak{B})$, by Corollary 1.3.20, as g_u and g_v are $\mathfrak{B}$-smooth.

We now consider the $(g_u x_u^2 \operatorname{N}(\mathfrak{b}_k), g_v x_v^2 \operatorname{N}(\mathfrak{c}_k))$-isogeny diamond

$$\begin{tikzcd}
 & E \arrow[d, "\varphi_{\mathfrak{b}_e}"] & & \\
E_u \arrow[r, "{[x_u]\circ\hat{\varphi}_u}"] \arrow[dd] & E^{\mathfrak{b}_e} \arrow[r, "\varphi_{\mathfrak{b}_k}"] & E_1 \arrow[d, "\lambda\circ\varphi_{\bar{\mathfrak{c}}_k}"] & \\
 & & E^{\mathfrak{c}_e} \arrow[r, "\hat{\varphi}_{\mathfrak{c}_e}"] \arrow[d, "{[x_v]\circ\varphi_v}"] & E \\
X \arrow[rr] & & E_v & ,
\end{tikzcd}$$

where the actions of $\mathfrak{b}_e$ and $\mathfrak{c}_e$ are also indicated to clarify the general structure. Since the bottom-left part of the Kani diagram is not useful for the rest of the proof, it is left blank. By Proposition 1.6.10, there exist isogenies and a PPAV completing those diagrams in a commutative way.

By Lemma 1.6.9, there is a 2^e-isogeny $F : E_u \times E_v \to E_1 \times X$ such that

$$\ker F = \{([x_u^2 g_u \operatorname{N}(\mathfrak{b}_k)](P), [x_u x_v]\varphi_v \circ \lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ \varphi_{\mathfrak{b}_k} \circ \hat{\varphi}_u(P) \text{ such that } P \in E_u[2^e]\}.$$

By Proposition 4.3.6, this kernel can be written as

$$\ker F = \{([x_u^2 g_u \operatorname{N}(\mathfrak{b}_k)](P), [x_u x_v \operatorname{N}(\mathfrak{c}_e\mathfrak{b}_e)^{-1}]\varphi_v \circ \varphi_{\mathfrak{c}_e} \circ \iota(\alpha) \circ \hat{\varphi}_{\mathfrak{b}_e} \circ \hat{\varphi}_u(P) \text{ such that } P \in E_u[2^e]\}.$$

One can define this kernel directly from points of E, since, by coprimality between $g_u \operatorname{N}(\mathfrak{b}_e)$ and 2^e, we have

$$\varphi_u \circ \varphi_{\mathfrak{b}_e}(E[2^e]) = E_u[2^e].$$

Then, with additional simplifications, the kernel of F is given as

$$\ker F = \{([x_u \operatorname{N}(\mathfrak{b}_k)]\varphi_u \circ \varphi_{\mathfrak{b}_e}(P), [x_v \operatorname{N}(\mathfrak{c}_e)^{-1}]\varphi_v \circ \varphi_{\mathfrak{c}_e} \circ \iota(\alpha)(P) \text{ such that } P \in E[2^e]\}.$$

Since (E, p, f) is a `PEGASIS` triplet, the 2^e-torsion, with $e < f$, is defined over $\mathbb{F}_{p^2}$. Hence, computing this kernel only requires to evaluate chains of efficiently represented isogenies on points defined over $\mathbb{F}_{p^2}$.[3] This takes a time polynomial in the length of the input.

Then, as E_1 is isomorphic to $E^{\mathfrak{b}}$, which is in turn isomorphic to $E^{\mathfrak{a}}$, it only remains to compute the 2^e-isogeny F of dimension 2 to obtain an elliptic curve isomorphic to $E^{\mathfrak{a}}$. □

In terms of complexity, for a given integer n, 2^n-isogenies in dimension 2 are the most efficient higher dimensional isogenies one can hope for. Indeed, considering higher dimensions, or isogenies of degree divisible by an integer greater than 2, directly increases the total complexity. Using implementations from the current literature, computing the codomain of a 2^n-isogeny in dimension 2 takes only twice as long as for 2^n-isogenies in dimension 1, see [DMPR24, Table 4]. We refer the reader to this same paper for more details about the computation of such isogenies.

Unfortunately, this method is impractical due to the very low probability for u and v to be perfect squares as the following classical proposition suggests.

[3] We recall that it is possible to perform all the computation over $\mathbb{F}_p$, see [DEF$^+$25a] for more details.

Proposition 4.3.9. *Let n be a positive integer. An integer drawn uniformly at random in $[\![0, n]\!]$ is a perfect square with probability approximately $1/\sqrt{n}$.*

Notice that removing smooth factors from the integers u and v reduces their size and so improves their probability to be perfect squares.

The $\mathfrak{B}$-good case. We have seen that using isogenies in dimension 2 allows us to compute the action of ideals in the specific case where Equation (4.2) admits $\mathfrak{B}$-perfect solutions. At the cost of moving to higher dimension, one can relax this perfect-square condition to a sum-of-two-squares condition. To improve the probability to find suitable solutions u and v, we also allow them to be sum of two squares up to smooth factors in a set of primes $\mathfrak{B}$. We define such a pair (u, v) to be $\mathfrak{B}$-good.

Definition 4.3.10. *Let $\mathfrak{B}$ be a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$. We say that an integer $u \in \mathbb{Z}_{>0}$ is $\mathfrak{B}$-**good** if u can be written as $g_u(x_u^2 + y_u^2)$ with $x_u, y_u \in \mathbb{Z}_{\geq 0} \setminus \{(0, 0)\}$ and $g_u \in \mathbb{Z}_{>0}$ a product of primes in $\mathfrak{B}$. We say that a pair of integers (u, v) is $\mathfrak{B}$-good when u and v are $\mathfrak{B}$-good.*

Note that a $\mathfrak{B}$-good pair (u, v) is $\mathfrak{B}$-perfect when $y_u = y_v = 0$.

Proposition 4.3.11. *Let $\mathfrak{B}$ be a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$. Let (E, p, f) be a PEGASIS triplet and $\mathfrak{a}$ be a $\mathbb{Z}[(\sqrt{-p} + 1)/2]$-ideal. Given*

- *the elliptic curve $E/\mathbb{F}_p$,*
- *two ideals $\mathfrak{b} = \mathfrak{b}_e\mathfrak{b}_k$ and $\mathfrak{c} = \mathfrak{c}_e\mathfrak{c}_k$ of coprime norms equivalent to $\mathfrak{a}$ such that $\mathfrak{b}_e$ and $\mathfrak{c}_e$ are respectively the $\mathfrak{B}$-smooth parts of $\mathfrak{b}$ and $\mathfrak{c}$,*
- *an integer $e < f - 2$,*
- *a coprime $\mathfrak{B}$-good pair $(u, v) = (g_u(x_u^2 + y_u^2), g_v(x_v^2 + y_v^2))$ such that*

$$g_u(x_u^2 + y_u^2)\,\mathrm{N}(\mathfrak{b}_k) + g_v(x_v^2 + y_v^2)\,\mathrm{N}(\mathfrak{c}_k) = 2^e,$$

one can compute $j(E^{\mathfrak{a}})$ in time polynomial in the length of the input and in $\max(\mathfrak{B})$ plus the cost of computing a 2^e-isogeny of dimension 4.

Proof. The only difference with the $\mathfrak{B}$-perfect case concerns the forms of u and v. Hence, we start by stating the same objects from Proposition 4.3.6 as in the proof of Proposition 4.3.8. Let α be a generator of the principal ideal $\mathfrak{b}\bar{\mathfrak{c}}$ and λ be the isomorphism such that

$$\varphi_{\mathfrak{c}_e} \circ \iota(\alpha) \circ \hat{\varphi}_{\mathfrak{b}_e} = [\mathrm{N}(\bar{\mathfrak{c}}_e\mathfrak{b}_e)] \circ \lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ \varphi_{\mathfrak{b}_k}.$$

As before, we compute effective representations of the isogenies $\varphi_{E,\mathfrak{b}_e}$, $\varphi_{E,\mathfrak{c}_e}$ and of some g_u-isogeny $\varphi_u : E^{\mathfrak{b}_e} \to E_u$ and g_v-isogeny $\varphi_v : E^{\mathfrak{c}_e} \to E_v$ in time polynomial in the length of the input and in $\max(\mathfrak{B})$. However, to provide a suitable Kani diagram, we need to construct two isogenies of degree $u = g_u(x_u^2 + y_u^2)$ and $v = g_v(x_v^2 + y_v^2)$ respectively. Since $x_u^2 + y_u^2$ and $x_v^2 + y_v^2$ are no longer perfect squares, there is no scalar endomorphism with such degrees. Nevertheless, in dimension 2, it is possible to construct endomorphisms of degree $x^2 + y^2$, for any pair of integers (x, y). This relies on the fact that the matrix

$$\begin{pmatrix} [x] & -[y] \\ [y] & [x] \end{pmatrix}$$

defines an (x^2+y^2)-endomorphism over any product of two elliptic curves. Then, we can define $\Phi_u : (E^{\mathfrak{b}_e})^2 \to E_u^2$ to be the $g_u(x_u^2+y_u^2)$-isogeny given by the matrix

$$\begin{pmatrix} [x_u]\circ\varphi_u & -[y_u]\circ\varphi_u \\ [y_u]\circ\varphi_u & [x_u]\circ\varphi_u \end{pmatrix}$$

and $\Phi_v : (E^{\mathfrak{c}_e})^2 \to E_v^2$ to be the $g_v(x_v^2+y_v^2)$ isogeny given by the matrix

$$\begin{pmatrix} [x_v]\circ\varphi_v & -[y_v]\circ\varphi_v \\ [y_v]\circ\varphi_v & [x_v]\circ\varphi_v \end{pmatrix}.$$

By translating the perfect case Kani diagram to dimension 2 and substituting the isogeny $[x_u]\varphi_u$ (resp. $[x_v]\varphi_v$) by Φ_u (resp. Φ_v), we obtain the following $(g_u(x_u^2+y_u^2)\,\mathrm{N}(\mathfrak{b}_k), g_v(x_v^2+y_v^2)\,\mathrm{N}(\mathfrak{c}_k))$-isogeny diamond, again with the additional actions represented for global understanding,

$$\begin{tikzcd}
 & E^2 \arrow[d, "\varphi_{\mathfrak{b}_e}^{\times 2}"] & & \\
E_u^2 \arrow[r, "\tilde{\Phi}_u"] \arrow[ddd] & (E^{\mathfrak{b}_e})^2 \arrow[r, "\varphi_{\mathfrak{b}_k}^{\times 2}"] & E_1^2 \arrow[d, "(\lambda\circ\varphi_{\bar{\mathfrak{c}}_k})^{\times 2}"] & \\
 & & (E^{\mathfrak{c}_e})^2 \arrow[r, "\varphi_{\bar{\mathfrak{c}}_e}^{\times 2}"] \arrow[d, "\Phi_v"] & E^2 \\
 & & & \\
X \arrow[rr] & & E_v^2 & .
\end{tikzcd}$$

By Kani's Lemma, there is a 2^e-isogeny $F : E_u^2 \times E_v^2 \to E_1^2 \times X$ such that

$$\ker F = \{([u\,\mathrm{N}(\mathfrak{b}_k)](P), \Phi_v \circ (\lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ \varphi_{\mathfrak{b}_k})^{\times 2} \circ \tilde{\Phi}_u(P)) \text{ such that } P \in E_u^2[2^e]\}.$$

By construction, we have

$$(\lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ \varphi_{\mathfrak{b}_k})^{\times 2}(P) = [\mathrm{N}(\mathfrak{b}_e\mathfrak{c}_e)^{-1}](\varphi_{\mathfrak{c}_e} \circ \iota(\alpha) \circ \hat{\varphi}_{\mathfrak{b}_e})^{\times 2}(P), \forall P \in E_u^2[2^e].$$

Then, using the same simplifications as in the proof of Proposition 4.3.8, this kernel can be written as

$$\ker F = \{([\mathrm{N}(\mathfrak{b}_k)]\Phi_u \circ \varphi_{\mathfrak{b}_e}^{\times 2}(P), [\mathrm{N}(\mathfrak{c}_e)^{-1}]\Phi_v \circ (\varphi_{\mathfrak{c}_e} \circ \iota(\alpha))^{\times 2}(P)) \text{ such that } P \in E^2[2^e]\}.$$

As before, this kernel is defined over $\mathbb{F}_{p^2}$ and all involved isogenies are efficiently represented. Hence, it can be computed in time polynomial in the length of the input. Thus, to get the isomorphism class of $E^{\mathfrak{a}}$, all that remains is to compute F from its kernel. □

The main difference between computing an action from a $\mathfrak{B}$-good solution and from a $\mathfrak{B}$-perfect solution lies in the dimension of the isogeny between PPAVs that needs to be handled. The rest of the steps being mostly equivalent. While computing a 2^e-isogeny in dimension 2 is roughly twice as long as computing a 2^e-isogeny in dimension 1, computing such an isogeny in dimension 4 can take 20 times longer, see for instance [Dar24, Table 2]. Nevertheless, this drawback is acceptable as the probability to find such solutions is greatly improved.

Proposition 4.3.12 ([Lan09])**.** *Let n be a positive integer. Asymptotically, an integer drawn uniformly at random between 0 and n is sum of two squares with probability $\frac{C}{\sqrt{\log(n)}}$, with $C \approx 0.8$ the Landau-Ramanujan constant.*

Asymptotically, by Proposition 4.3.12 and Proposition 4.3.9, we see that, while the probability that a random integer is a perfect square is negligible, one could expect to find sums of two squares among a list of integers having a logarithmic number of elements. However, having a good proportion of sums of two squares does not mean that they are all accessible. In particular, one needs to detect them and, then, to compute the two squares.

Actually, it is harder to check if a pair (u, v) is $\mathfrak{B}$-good than if it is $\mathfrak{B}$-perfect. For both cases, one factors out the $\mathfrak{B}$-smooth part of u, so we have

$$u = g_u u',$$

with $g_u \in \mathfrak{B}$ and $u' \in \mathbb{Z}_{>0}$. Then for $\mathfrak{B}$-perfectness, we directly compute the square root of the remainder u' in order to check if it is a perfect square. While for $\mathfrak{B}$-goodness, we need to factorise the integer u' in order to verify whether it fulfills the following well-known characterisation for sum of two squares. To the best of our knowledge, there is currently no better criterion.

Proposition 4.3.13 ([Dud08, Section 18. Theorem 1])**.** *An integer $n \in \mathbb{Z}_{>0}$ with prime factorisation $\prod_{i=0}^{r} \ell_i^{e_i} = n$ is not a sum of two squares if and only if there exists an index $i \in [\![1, r]\!]$ such that $e_i \equiv 1 \mod 2$ and $\ell_i \equiv 3 \mod 4$.*

Remark 4.3.14. *It is worth noticing that factoring out primes congruent to* 1 *modulo* 4 *does not improve the probability that an integer is a sum of two squares, while factoring out primes congruent to* 3 *modulo* 4 *does. This is why, in practice, the primes in the set* $\mathfrak{B}$ *are chosen congruent to* 3 *modulo* 4*.*

If the integer u' verifies this condition, using Cornacchia's algorithm [Cor07], one computes the two squares efficiently from its factorisation. Hence, this method is efficient if and only if factoring u' can be done efficiently. For instance, when

$$u' = g'_u \ell_u,$$

where g'_u is B'-smooth, for some small integer B', and ℓ_u is a prime, one can efficiently factor out the smooth part and test the primality of the cofactor. Note that, by the prime number theorem, the probability that u'/g'_u is a prime is around $1/\log(u'/g'_u)$. Then, using this process, from a list of possible u with a polylogarithmic number of elements, one should be able to find a u expressible as a $g_u(x^2 + y^2)$. This is the specific process used in [DEF$^+$25a], with B' up to 10^4 or 10^5 depending on the size of p.

The $\mathfrak{B}$-semigood case. It is further possible to relax the $\mathfrak{B}$-good condition to improve the probability of finding (u, v) such that the isogenies of degree u and v can be efficiently computed in dimension 2. We name this weaker condition $\mathfrak{B}$-semigoodness.

Definition 4.3.15. *Let $\mathfrak{B}$ be a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$. We say that an integer $u \in \mathbb{Z}_{>0}$ is $\mathfrak{B}$-**semigood** if u can be written as $g_u(ax_u^2 + by_u^2)$ with $x_u, y_u \in \mathbb{Z}_{>0}$, and $g_u, a, b \in \mathbb{Z}_{>0}$ are products of primes in $\mathfrak{B}$. A pair (u, v) is $\mathfrak{B}$-semigood if both u and v are $\mathfrak{B}$-semigood for the* same *a and b.*

This is a generalisation of the $\mathfrak{B}$-good integers, since a $\mathfrak{B}$-semigood integer u is $\mathfrak{B}$-good when $a = b = 1$.

Remark 4.3.16. *For simplicity, we assume that a and b are the same for u and v. However, it should be possible to perform similar computations when $u = g_u(ax_u^2 + by_u^2)$ and $v = g_v(cx_v^2 + dy_v^2)$, further increasing the likelihood of finding such a pair (u, v).*

Proposition 4.3.17. *Let $\mathfrak{B}$ be a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$. Let (E, p, f) be a* `PEGASIS` *triplet and $\mathfrak{a}$ be a $\mathbb{Z}[(\sqrt{-p}+1)/2]$-ideal. Given*

- *the elliptic curve $E/\mathbb{F}_p$,*
- *two ideals $\mathfrak{b} = \mathfrak{b}_e\mathfrak{b}_k$ and $\mathfrak{c} = \mathfrak{c}_e\mathfrak{c}_k$ of coprime norms equivalent to $\mathfrak{a}$ such that $\mathfrak{b}_e$ and $\mathfrak{c}_e$ are respectively the $\mathfrak{B}$-smooth parts of $\mathfrak{b}$ and $\mathfrak{c}$,*
- *an integer $e < f - 2$,*
- *a coprime $\mathfrak{B}$-semigood pair $(u, v) = (g_u(ax_u^2 + by_u^2), g_v(ax_v^2 + by_v^2))$ such that*

$$g_u(ax_u^2 + by_u^2)\,\mathrm{N}(\mathfrak{b}_k) + g_v(ax_v^2 + by_v^2)\,\mathrm{N}(\mathfrak{c}_k) = 2^e,$$

one can compute $j(E^{\mathfrak{a}})$ in time polynomial in the length of the input and in $\max(\mathfrak{B})$ *plus the cost of computing a 2^e-isogeny of dimension 4.*

Proof. This situation differs from Proposition 4.3.11 only by the presence of the integers a and b. While it may seem to be a minor change, the impact on the construction is not negligible.

An important new step is to build $\mathbb{Z}[\frac{\sqrt{-p}+1}{2}]$-ideals of norm a and b. Their actions on $\mathbb{Z}[\frac{\sqrt{-p}+1}{2}]$-oriented elliptic curves provide efficiently represented isogenies of degree a and b, whence the condition for a and b to split in $\mathbb{Z}[\frac{\sqrt{-p}+1}{2}]$. We compute such ideals as follows.

For $c \in \{a, b\}$, given its factorisation $\prod_{i=1}^n \ell_i$, we compute the $\mathbb{Z}[\frac{\sqrt{-p}+1}{2}]$-ideal

$$I_c := \prod_{i=1}^{n} \langle \ell_i, \frac{s_\ell + \sqrt{-p}}{2} \rangle,$$

where s_ℓ is a square root of Δ modulo 4ℓ. Factoring c and computing I_c can be done in time polynomial in $\max(\mathfrak{B})$, $\log c$, and $\log p$. This ideal is invertible and has norm c by a direct application of [BP89, Proposition 8.6.4]. We denote by E' the curve obtained by acting on E with the ideal $I_a \cdot I_b$.

Once again, we consider a generator α of the principal ideal $\mathfrak{b}\bar{\mathfrak{c}}$ and apply Proposition 4.3.6 to obtain an isomorphism λ such that

$$\varphi_{\mathfrak{c}_e} \circ \iota(\alpha) \circ \hat{\varphi}_{\mathfrak{b}_e} = [\mathrm{N}(\bar{\mathfrak{c}}_e\mathfrak{b}_e)] \circ \lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ \varphi_{\mathfrak{b}_k}. \tag{4.5}$$

We use the notation of the proposition, summarised in the following sequence of isogenies

$$E \xrightarrow{\varphi_{\mathfrak{b}_e}} E^{\mathfrak{b}_e} \xrightarrow{\varphi_{\mathfrak{b}_k}} E_1 \xrightarrow{\lambda\circ\varphi_{\bar{\mathfrak{c}}_k}} E^{\mathfrak{c}_e} \xrightarrow{\hat{\varphi}_{\mathfrak{c}_e}} E.$$

In this case, we also apply this proposition to (E', p, f). Therefore, we obtain an isomorphism λ' such that

$$\varphi'_{\mathfrak{c}_e} \circ \iota'(\alpha) \circ \hat{\varphi}'_{\mathfrak{b}_e} = [\mathrm{N}(\bar{\mathfrak{c}}_e\mathfrak{b}_e)] \circ \lambda' \circ \varphi'_{\bar{\mathfrak{c}}_k} \circ \varphi'_{\mathfrak{b}_k}, \tag{4.6}$$

with the following sequence of isogenies

$$E' \xrightarrow{\varphi'_{\mathfrak{b}_e}} E'^{\mathfrak{b}_e} \xrightarrow{\varphi'_{\mathfrak{b}_k}} E'_1 \xrightarrow{\lambda'\circ\varphi'_{\bar{\mathfrak{c}}_k}} E'^{\mathfrak{c}_e} \xrightarrow{\hat{\varphi}'_{\mathfrak{c}_e}} E'.$$

We compute efficient representations of $\varphi_{\mathfrak{b}_e}$, $\varphi_{\mathfrak{c}_e}, \varphi'_{\mathfrak{b}_e}$ and $\varphi'_{\mathfrak{c}_e}$ in time polynomial in $\log \mathrm{N}(\mathfrak{b}_e), \log \mathrm{N}(\mathfrak{c}_e), \log p$ and $\max(\mathfrak{B})$. In addition, we consider the following 2-dimensional diagonal isogenies

$$\begin{aligned}
\Phi_\alpha &:= \begin{pmatrix} \iota(\alpha) & 0 \\ 0 & \iota'(\alpha) \end{pmatrix} : E \times E' \to E \times E', \\
\Phi_{\mathfrak{b}_e} &:= \begin{pmatrix} \varphi_{\mathfrak{b}_e} & 0 \\ 0 & \varphi'_{\mathfrak{b}_e} \end{pmatrix} : E \times E' \to E^{\mathfrak{b}_e} \times E'^{\mathfrak{b}_e}, \\
\Phi_{\mathfrak{b}_k} &:= \begin{pmatrix} \varphi_{\mathfrak{b}_k} & 0 \\ 0 & \varphi'_{\mathfrak{b}_k} \end{pmatrix} : E^{\mathfrak{b}_e} \times E'^{\mathfrak{b}_e} \to E_1 \times E'_1, \\
\Phi_{\bar{\mathfrak{c}}_k} &:= \begin{pmatrix} \lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ & 0 \\ 0 & \lambda' \circ \varphi'_{\bar{\mathfrak{c}}_k} \end{pmatrix} : E_1 \times E'_1 \to E^{\mathfrak{c}_e} \times E'^{\mathfrak{c}_e}, \text{ and} \\
\Phi_{\mathfrak{c}_e} &:= \begin{pmatrix} \varphi_{\mathfrak{c}_e} & 0 \\ 0 & \varphi'_{\mathfrak{c}_e} \end{pmatrix} : E^{\mathfrak{c}_e} \times E'^{\mathfrak{c}_e} \to E \times E.
\end{aligned}$$

By Equation (4.5) and Equation (4.6), we have the equality

$$\Phi_{\mathfrak{c}_e} \circ \Phi_\alpha \circ \tilde{\Phi}_{\mathfrak{b}_e} = [\mathrm{N}(\mathfrak{c}_e \mathfrak{b}_e)] \circ \Phi_{\bar{\mathfrak{c}}_k} \circ \Phi_{\mathfrak{b}_k}. \tag{4.7}$$

Let us now turn to the computation of isogenies of degree u and v. First, we consider the two diagrams

$$\begin{tikzcd}
E_b^{\mathfrak{b}_e} \arrow[r, "{[x_u]\lambda_1 \circ \varphi'_{1,a}}"] & E_{ab}^{\mathfrak{b}_e} \\
E^{\mathfrak{b}_e} \arrow[u, "{[y_u]\varphi_{1,b}}"] \arrow[r, "{[x_u]\varphi_{1,a}}"] & E_a^{\mathfrak{b}_e} \arrow[u, "{[y_u]\varphi'_{1,b}}"']
\end{tikzcd}
\qquad
\begin{tikzcd}
E_b^{\mathfrak{c}_e} \arrow[r, "{[x_v]\lambda_2 \circ \varphi'_{2,a}}"] & E_{ab}^{\mathfrak{c}_e} \\
E^{\mathfrak{c}_e} \arrow[u, "{[y_v]\varphi_{2,b}}"] \arrow[r, "{[x_v]\varphi_{2,a}}"] & E_a^{\mathfrak{c}_e} \arrow[u, "{[y_v]\varphi'_{2,b}}"']
\end{tikzcd}$$

where $\varphi_{1,a}, \varphi'_{1,a}, \varphi_{2,a}, \varphi'_{2,a}$ (resp. $\varphi_{1,b}, \varphi'_{1,b}, \varphi_{2,b}, \varphi'_{2,b}$) are given by the action of I_a (resp. I_b) and λ_1, λ_2 are the isomorphisms making the diagrams commute. Such isomorphisms exist by Proposition 1.4.7 and can be computed efficiently by exhaustive search.

We also compute two isomorphisms $\lambda_{\mathfrak{b}_e} : E_{ab}^{\mathfrak{b}_e} \to E'^{\mathfrak{b}_e}$ and $\lambda_{\mathfrak{c}_e} : E_{ab}^{\mathfrak{c}_e} \to E'^{\mathfrak{c}_e}$. Again, they exist by Proposition 1.4.7 and can be computed by exhaustive search.

Combining these isomorphisms with the diagrams provides an $(ax_u^2 + by_u^2)$-isogeny $\Phi_1 : E^{\mathfrak{b}_e} \times E'^{\mathfrak{b}_e} \to E_a^{\mathfrak{b}_e} \times E_b^{\mathfrak{b}_e}$ and an $(ax_v^2 + by_v^2)$-isogeny $\Phi_2 : E^{\mathfrak{c}_e} \times E'^{\mathfrak{c}_e} \to E_a^{\mathfrak{c}_e} \times E_b^{\mathfrak{c}_e}$ explicitly given by the matrices

$$\begin{pmatrix} [x_u]\varphi_{1,a} & [y_u]\hat{\varphi}'_{1,b} \circ \hat{\lambda}_{\mathfrak{b}_e} \\ -[y_u]\varphi_{1,b} & [x_u]\hat{\varphi}'_{1,a} \circ \hat{\lambda}_1 \circ \hat{\lambda}_{\mathfrak{b}_e} \end{pmatrix} \text{ and } \begin{pmatrix} [x_v]\varphi_{2,a} & [y_v]\hat{\varphi}'_{2,b} \circ \hat{\lambda}_{\mathfrak{c}_e} \\ -[y_v]\varphi_{2,b} & [x_v]\hat{\varphi}'_{2,a} \circ \hat{\lambda}_2 \circ \hat{\lambda}_{\mathfrak{c}_e} \end{pmatrix}.$$

Finally, we compute two g_u-isogenies $\varphi_u : E_a^{\mathfrak{b}_e} \to E_u$ and $\varphi'_u : E_b^{\mathfrak{b}_e} \to E'_u$ as well as two g_v-isogenies $\varphi_v : E_a^{\mathfrak{c}_e} \to E_v$ and $\varphi'_v : E_b^{\mathfrak{c}_e} \to E'_v$ in time polynomial in $\log g_u, \log g_v, \log p$ and in $\max(\mathfrak{B})$. They provide a g_u-isogeny $\Phi'_1 : E_a^{\mathfrak{b}_e} \times E_b^{\mathfrak{b}_e} \to E_u \times E'_u$ and a g_v-isogeny $\Phi'_2 : E_a^{\mathfrak{c}_e} \times E_b^{\mathfrak{c}_e} \to E_v \times E'_v$ by considering the diagonal matrices

$$\begin{pmatrix} \varphi_u & 0 \\ 0 & \varphi'_u \end{pmatrix} \text{ and } \begin{pmatrix} \varphi_v & 0 \\ 0 & \varphi'_v \end{pmatrix}.$$

Hence, by composing them, we obtain isogenies with the desired degrees. Let us denote by Φ_u the $g_u(ax_u^2 + by_u^2)$-isogeny $\Phi'_1 \circ \Phi_1 : E^{\mathfrak{b}_e} \times E'^{\mathfrak{b}_e} \to E_u \times E'_u$ and by Φ_v the $g_v(ax_v^2 + by_v^2)$-isogeny $\Phi'_2 \circ \Phi_2 : E^{\mathfrak{c}_e} \times E'^{\mathfrak{c}_e} \to E_v \times E'_v$.

We can now state the appropriate Kani diagram in dimension 4

$$\begin{tikzcd}
& E \times E' \arrow[d, "\Phi_{\mathfrak{b}_e}"] & & \\
E_u \times E'_u \arrow[r, "\tilde{\Phi}_u"] \arrow[ddd] & E^{\mathfrak{b}_e} \times E'^{\mathfrak{b}_e} \arrow[r, "\Phi_{\mathfrak{b}_k}"] & E^{\mathfrak{b}} \times E'^{\mathfrak{b}} \arrow[d, "\Phi_{\bar{\mathfrak{c}}_k}"] & \\
& & E^{\mathfrak{c}_e} \times E'^{\mathfrak{c}_e} \arrow[r, "\Phi_{\bar{\mathfrak{c}}_e}"] \arrow[dd, "\Phi_v"] & E \times E' \\
& & & \\
X \arrow[rr] & & E_v \times E'_v & .
\end{tikzcd}$$

By Kani's Lemma, there is a 2^e-isogeny $F : E_u \times E'_u \times E_v \times E'_v \to E^{\mathfrak{b}} \times E'b \times X$ such that

$$\ker F = \{([u\,\mathrm{N}(\mathfrak{b}_k)](P), \Phi_v \circ \Phi_{\bar{\mathfrak{c}}_k} \circ \Phi_{\mathfrak{b}_k} \circ \tilde{\Phi}_u(P) \text{ such that } P \in (E_u \times E'_u)[2^e]\}.$$

By Equation (4.7) and using the same simplifications as for the other cases, we have

$$\ker F = \{([\mathrm{N}(\mathfrak{b}_k)]\Phi_u \circ \Phi_{\mathfrak{b}_e}(P), [\mathrm{N}(\mathfrak{c}_e)^{-1}]\Phi_v \circ \Phi_{\mathfrak{c}_e} \circ \Phi_\alpha(P) \text{ such that } P \in E \times E'[2^e]\}.$$

Hence, this kernel is computable in time polynomial in the length of the input. Finally, one recovers an elliptic curve isomorphic to $E^{\mathfrak{a}}$ by computing the isogeny F from its kernel. □

The advantages and drawbacks of the $\mathfrak{B}$-semigood approach compared to the $\mathfrak{B}$-good one, are less clear than in previous comparisons.

On the one hand, handling $\mathfrak{B}$-semigood has a higher computational cost. While in both cases, a single 2^e-isogeny in dimension 4 must be computed, in the $\mathfrak{B}$-semigood case, obtaining the kernel of this 4-dimensional isogeny requires more prior steps. The main additional costs are

- computing the ideals I_a and I_b and their actions over four different curves each,
- computing the action of the ideals $\mathfrak{b}_e$ and $\mathfrak{c}_e$ on a second curve.

Furthermore, verifying whether a candidate solution (u, v) is suitable is also more costly, due to the multiple possible forms.

On the other hand, one may expect that the probability of an integer being representable as $x^2 + by^2$, with $b \in \{1, 2, 3, 7\}$, is around three times greater than the probability of being representable by a sum of two squares. The table of experimental results in Figure 4.2 supports this estimate.

Integer length	32 bits	64 bits	128 bits
Success rate when $b = 1$	0.11	0.08	0.05
Success rate when $b \in \{1, 2, 3, 7\}$	0.29	0.22	0.16

Figure 4.2: Success rates for representing an integer as $x^2 + by^2$, with $x, y \in \mathbb{N}$, depending on the possible values for b. Each rate is computed from 10^5 integers drawn uniformly at random.

Let us briefly explain why the set $\{2, 3, 7\}$ is an interesting choice for $\mathfrak{B}$ regarding b.

We first focus on forms $x^2 + by^2$ where b is prime. Our discussion relies mainly on Proposition 4.3.18 and on the fact that an integer whose prime factors are all represented by such a quadratic form can itself be represented by this form — this is a direct corollary of norm complete multiplicity in $\mathbb{Z}[\sqrt{-b}]$.

Proposition 4.3.18. *[Cox13, Corollary 2.6] Let b be an integer and p be an odd prime such that $\gcd(b,p) = 1$. Then p is represented by a primitive form of discriminant $-4b$ if and only if its Legendre symbol $(-b/p)$ is equal to 1.*

Note that the quadratic form $x^2 + by^2$ has discriminant $-4b$. Hence, the best choices for b to maximise the probability that primes are represented by $x^2 + by^2$ are those for which the class group of discriminant $-4b$ has cardinality 1. Indeed, by Proposition 4.3.18, if $(b/p) = 1$, then p is represented by a primitive form of discriminant $4b$. When there is a unique class in this group, we have that $x^2 + by^2$ is equivalent to this primitive form. Therefore, $x^2 + by^2$ also represents p.

Since an integer has a probability roughly of one-half of being a quadratic residue modulo a prime p, we expect that $x^2 + by^2$ represents p with probability one-half. In addition, because the set of even discriminants of class number 1 is $\{-4, -8, -12, -16, -28\}$, there are only 4 such forms interesting to us: $x^2 + y^2$, $x^2 + 2y^2$, $x^2 + 3y^2$ and $x^2 + 7y^2$. We cover all of them with the set of primes $\{2, 3, 7\}$.

The second-best choices for b are those for which the corresponding class group has cardinality 2, such as 5 or 13. The probability that a prime is representable by $x^2 + by^2$ is then halved for such b. However, from [Cox13], we have the following equivalences

$$\begin{aligned} p &= x^2 + y^2 && \text{iff } p \equiv 1 \mod 4, \\ p &= x^2 + 5y^2 && \text{iff } p \equiv 1, 9 \mod 20, \\ p &= x^2 + 13y^2 && \text{iff } p \equiv 1, 9, 17, 25, 29, 49 \mod 52. \end{aligned}$$

Hence, if p is represented by $x^2 + 5y^2$ or $x^2 + 13y^2$, it is also represented by $x^2 + y^2$. Nevertheless, adding the primes 5 and 13 to $\mathfrak{B}$ still improves the likelihood of representing a prime since they allow us to represent primes by the forms $x^2 + 10y^2$ or $x^2 + 15y^2$, for instance. In this work, we have chosen to focus on forms of the form $x^2 + by^2$ with b prime for simplicity, so we do not consider them. The other reason to avoid the forms $ax^2 + by^2$ where $a \neq 1$ or where b is a composite number is that most of them lead to class groups of cardinality greater than 3 and experiments have shown that adding a quadratic form from a class group with cardinality 3 or more has only a negligible impact on the ability to represent a prime. Additionally, the next prime after 13 that yields a class group of cardinality 2 is 37. Therefore, to achieve the most significant improvement with the smallest set of primes, it seems reasonable to consider only primes up to 7.

The $\mathfrak{B}$-arbitrary case. Let us now discuss how to compute actions from arbitrary solutions (u, v) to Equation (4.3). Most of the process can be carried out similarly to Proposition 4.3.11. The main difficulty lies in the construction of u-isogenies and v-isogenies when the integers u and v are neither squares nor sum of two squares. One natural idea is to use the fact that any integer can be written as a sum of 4 squares, hence it is possible to construct an endomorphism in dimension 8 of any given degree, due to Zarhin's trick [Zar74], as we did in Section 2.2. This approach is suitable for theoretical results. However, for practical purposes, moving to dimension 8 is too costly.

The method adopted here is to adapt a construction from [NO24, `QFESTA`] to compute isogenies of prescribed degree. In that paper, the authors build isogenies of degree n from endomorphisms of degree $n(2^e - n)$ via the computation of an isogeny in dimension 2. To first compute endomorphisms of degree $n(2^e - n)$, they rely on heuristic algorithms, for instance [DLLW23, FullRepresentInteger], to represent $n(2^e - n)$ by a norm form on some endomorphism ring.

Here, it is not possible to directly use such algorithms, as the endomorphism ring of the elliptic curves is a priori unknown. Nevertheless, as we shall see, the orientations provide enough information to construct endomorphisms of degree $n(2^e - n)$ in dimension 2. Unfortunately, to apply the `QFESTA` trick from an endomorphism in dimension 2, one needs to perform a computation in dimension 4. Furthermore, once the 2-dimensional isogenies of degree u and v are obtained, one still needs to compute an isogeny in dimension 4, as in the $\mathfrak{B}$-good case, to get the image of the action. Thus, this approach is definitely the most expensive in terms of complexity. It is also more heuristic than the previous ones.

We now turn to the computation of isogenies of degree u and v in dimension 2 under a heuristic assumption.

Proposition 4.3.19. *Let (E, p, f) be a* ***PEGASIS*** *triplet, $M \in \mathbb{Z}_{>0}$ an integer and $q(x, y, z, t) = (x^2 + y^2) + p(z^2 + t^2)$ be a quadratic form. From a tuple $(x_0, y_0, z_0, t_0) \in \mathbb{Z}^4$ such that $q(x_0, y_0, z_0, t_0) = M$, one can compute an efficient representation of an M-endomorphism of E^2 in polynomial time in $\log p$ and $\log M$.*

Proof. We first consider the form $q_0(x, z) = x^2 + pz^2$. This is a norm form over $\mathbb{Z}[\pi] \subset \mathrm{End}_{F_p}(E)$, since for any endomorphism $\theta = x + z\pi \in \mathbb{Z}[\pi]$, we have $\deg(\theta) = q_0(x, z)$. Hence, the form q represents sum of the degree of pairs of endomorphisms in $\mathbb{Z}[\pi]$, as

$$q(x, y, z, t) = (x^2 + y^2) + p(z^2 + t^2) = q_0(x, z) + q_0(y, t).$$

Given a tuple (x_0, y_0, z_0, t_0) representing M, one can define two endomorphisms $\gamma_0 = x + z\pi$ and $\gamma_1 = y + t\pi$ such that $\deg \gamma_0 + \deg \gamma_1 = M$. Then the matrix

$$\begin{pmatrix} \gamma_0 & \hat{\gamma}_1 \\ -\gamma_1 & \hat{\gamma}_0 \end{pmatrix}$$

defines an endomorphism of E^2 of degree M. Indeed, we have

$$\begin{pmatrix} \gamma_0 & \hat{\gamma}_1 \\ -\gamma_1 & \hat{\gamma}_0 \end{pmatrix} \begin{pmatrix} \hat{\gamma}_0 & -\hat{\gamma}_1 \\ \gamma_1 & \gamma_0 \end{pmatrix} = \begin{pmatrix} \deg \gamma_0 + \deg \gamma_1 & -\gamma_0\hat{\gamma}_1 + \hat{\gamma}_1\gamma_0 \\ -\gamma_1\hat{\gamma}_0 + \hat{\gamma}_0\gamma_1 & \deg \gamma_0 + \deg \gamma_1 \end{pmatrix} = \begin{pmatrix} M & 0 \\ 0 & M \end{pmatrix}.$$

The last equality comes from the commutativity in $\mathbb{Z}[\pi]$.

This endomorphism is efficiently represented as its matrix coefficients are compositions of scalar multiplications and Frobenius endomorphisms. □

To the best of our knowledge, the best approaches to find a representation of a given integer by a norm form are heuristics.

Heuristic 4.3.20. *Let p be a prime and $M \geq p \log(p)$. There is an algorithm that takes as input p and M and computes a tuple of integers (x, y, t, z) such that*

$$x^2 + y^2 + p(t^2 + z^2) = M \tag{4.8}$$

in time polynomial in $\log(p)$.

Proof. As its heuristic nature suggests, we are unfortunately unable to provide a proof for this result. Morally, the heuristic algorithm, such as [DLLW23, Algorithm 1], used to find integers x, y, t, z such that a given integer $M \geq p \log p$ satisfies Equation (4.8) is the following:

1. sample random integers t and z,

2. use Cornacchia's algorithm [Cor07] to find x, y such that $x^2 + y^2 = M - p(t^2 + z^2)$,

3. if it fails, go back to (1),
4. return (x, y, t, z).

Since $M \geq p \log p$, there should be at least $\sqrt{\log p}$ possible pairs (t, z). By Proposition 4.3.12, we expect to find among them a pair (t, z) such that $M - p(t^2 + z^2)$ is a sum of two squares. □

Proposition 4.3.21. *Under Heuristic 4.3.20, given a* `PEGASIS`*-triplet (E, p, f) and an odd integer u approximately equal to $\sqrt{p}$, one can compute a u-isogeny of E^2 in time polynomial in $\log p$ plus a computation of a 2^{f-2}-isogeny in dimension 4.*

Proof. Let $M = u(2^{f-2} - u)$. By assumption on u, we have $M > p \log p$. Then by Heuristic 4.3.20, one can compute a tuple (x, y, z, t) such that

$$x^2 + y^2 + p(z^2 + t^2) = M,$$

in polynomial time. Using Proposition 4.3.19, one can define an M-endomorphism γ of E^2 in time polynomial in $\log p$. Then, the fact that $M = u(2^{f-2} - u)$ allows us to adapt the idea from [NO24] to build an isogeny of degree u. Note that u and $2^{f-2} - u$ are coprime by parity of u.

By Proposition 1.6.10, there exist a PPAV A_1 in dimension 2 together with an u-isogeny $\varphi_1 : E^2 \to A_1$ and an $(2^{f-2} - u)$-isogeny $\varphi_2 : A_1 \to E^2$ such that the M-endomorphism γ of E^2 is equal to their composition $\varphi_2 \circ \varphi_1$. This provides the following Kani diagram

$$\begin{tikzcd} E^2 \arrow[r, "\varphi_1"] \arrow[d] & A_1 \arrow[d, "\varphi_2"] \\ X \arrow[r] & E^2 \end{tikzcd} .$$

Thus, by Lemma 1.6.9, there is an 2^{f-2}-isogeny $F : E^4 \to A_1 \times X$ embedding $\varphi_1 : E^2 \to A_1$, such that its kernel is given by

$$\begin{aligned} \ker F &= \{([u](P), \varphi_2 \circ \varphi_1(P)) \text{ such that } P \in E^2[2^{f-2}]\} \\ &= \{([u](P), \gamma(P)) \text{ such that } P \in E^2[2^{f-2}]\}. \end{aligned}$$

□

Remark 4.3.22. *In practice, the* `PEGASIS` *algorithm finds pairs (u, v) such that $u, v \approx \sqrt{p}$, hence our assumption on u in Proposition 4.3.21. Moreover, u is odd since $(u \,\mathrm{N}(\mathfrak{b}), v \,\mathrm{N}(\mathfrak{c})) = 1$ and $u \,\mathrm{N}(\mathfrak{b}) + v \,\mathrm{N}(\mathfrak{c})$ is a power of two.*

We can now turn to the computation of the action when the solution is arbitrary.

Proposition 4.3.23. *Let $\mathfrak{B}$ be a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$. Let (E, p, f) be a* `PEGASIS` *triplet and $\mathfrak{a}$ be a $\mathbb{Z}[(\sqrt{-p} + 1)/2]$-ideal. Given*

- *the elliptic curve $E/\mathbb{F}_p$,*
- *two ideals $\mathfrak{b} = \mathfrak{b}_e \mathfrak{b}_k$ and $\mathfrak{c} = \mathfrak{c}_e \mathfrak{c}_k$ of coprime norms equivalent to $\mathfrak{a}$ such that $\mathfrak{b}_e$ and $\mathfrak{c}_e$ are respectively the $\mathfrak{B}$-smooth parts of $\mathfrak{b}$ and $\mathfrak{c}$,*
- *an integer $e < f - 2$,*
- *a coprime pair $\mathfrak{B}$-arbitrary (u, v) with $u, v \approx \sqrt{p}$ such that $u \,\mathrm{N}(\mathfrak{b}_k) + v \,\mathrm{N}(\mathfrak{c}_k) = 2^e$*

under Heuristic 4.3.20, one can compute $j(E^{\mathfrak{a}})$ in time polynomial in the length of the input and in $\max(\mathfrak{B})$ *plus the cost of computing in dimension* 4 *a 2^e-isogeny and two 2^{f-2}-isogenies.*

Proof. The construction is the same as in the proof of Proposition 4.3.11, hence we introduce the same objects. The only difference being the method to construct isogenies of degree u and v. Let α be a generator of the principal ideal $\bar{\mathfrak{c}}\mathfrak{b}$. By Proposition 4.3.17, there is an isomorphism λ, such that

$$[\mathrm{N}(\bar{\mathfrak{c}}_e\mathfrak{b}_e)]\lambda \circ \varphi_{\bar{\mathfrak{c}}_k} \circ \varphi_{\mathfrak{b}_k} = \varphi_{\mathfrak{c}} \circ \iota(\alpha) \circ \hat{\varphi}_{\mathfrak{b}_e}.$$

By Proposition 4.3.21, one can compute an u-isogeny $\Phi_u : (E^{\mathfrak{b}_e})^2 \to A_u$ and an v-isogeny $\Phi_v : (E^{\mathfrak{c}_e})^2 \to A_v$. This step runs in time polynomial in $\log p$ plus the computation of two 2^{f-2}-isogenies in dimension 4.

Finally, constructing the same Kani diagram as in the proof of Proposition 4.3.11 ensures the existence of an 2^e-isogeny $F : A_u \times A_v \to E^{\mathfrak{b}^2} \times X$ such that

$$\ker F = \{([\mathrm{N}(\mathfrak{b}_k)]\Phi_u \circ \varphi_{\mathfrak{b}_e}^{\times 2}(P), [\mathrm{N}(\mathfrak{c}_e)^{-1}]\Phi_v \circ (\varphi_{\mathfrak{c}_\mathfrak{c}}\iota(\alpha))^{\times 2}(P) \text{ such that } P \in E^2[2^e]\}.$$

Once again, this kernel is computable in time polynomial in the length of the input and in $\max(\mathfrak{B})$. Thus, at the cost of computing this 2^e-isogeny in dimension 4, one obtains the codomain $E^{\mathfrak{b}}$. Note that the considered diagram differs only by the codomains of Φ_u and Φ_v, A_u and A_v, which are now generic PPAV of dimension 2 instead of the squares of elliptic curves E_u^2 and E_v^2. □

The overall algorithm

Before presenting the adopted strategy for the implementation of `PEGASIS` regarding the possible forms for (u, v), we provide a high-level description of the algorithm in which it is integrated.

Let (E, p, f) be a `PEGASIS` triplet, $\mathfrak{B}$ a set of primes splitting in $\mathbb{Q}(\sqrt{-p})$ and $\mathfrak{a}$ a $\mathbb{Z}[\frac{\sqrt{-p}+1}{2}]$-ideal. To compute $[\mathfrak{a}] \star E$, we proceed as follows.

1. Compute an ordered list S of short ideals in $[\mathfrak{a}]$ and factor out their $\mathfrak{B}$-smooth part,
2. Select in S two ideals $\mathfrak{b} = \mathfrak{b}_e\mathfrak{b}_k$ and $\mathfrak{c} = \mathfrak{c}_e\mathfrak{c}_k$ of coprime norms,
3. Enumerate the solutions $(u, v) \in \mathbb{Z}_{>0}$ of $u\,\mathrm{N}(\mathfrak{b}_k) + v\,\mathrm{N}(\mathfrak{c}_k) = 2^e$, for $e \leq f - 2$,
4. If a pair (u, v) has the desired form, compute the action, otherwise choose a different pair $(\mathfrak{b}, \mathfrak{c})$.

As seen in the proof of Proposition 1.4.9, the first step is equivalent to finding small elements in $\mathfrak{a}$. This can be done efficiently thanks to its rank 2 lattice structure.

The third step can also be done efficiently, for instance, by describing the set of solutions explicitly. It is well-known that the set of solutions of a non-homogeneous linear Diophantine equation, such as

$$u\,\mathrm{N}(\mathfrak{b}_k) + v\,\mathrm{N}(\mathfrak{c}_k) = 2^e,$$

can be parametrised by an integer t as

$$(u, v) = (u_0 + t\,\mathrm{N}(\mathfrak{c}_k), v_0 - t\,\mathrm{N}(\mathfrak{b}_k)),$$

where (u_0, v_0) is a particular solution. Note that, by choosing the pair of ideals in order to minimise the product of their norms, we maximise the number of solutions (u, v).

Remark 4.3.24. *It is possible to rerandomise the input ideal $\mathfrak{a}$ by multiplying it by small splitting ideals. Hence, it seems highly improbable not to be able to compute an action.*

An important point is that depending on the kind of solutions we are searching for, we might need to consider many different pairs $(\mathfrak{b}, \mathfrak{c})$ as well as check many solutions (u, v). Hence, solving this norm equation is really not negligible and should also be taken into account when choosing an optimal strategy.

In the previous subsections, we studied how to compute ideal actions depending on the provided solutions of the `CLAPOTI` equation. We now summarise the advantages and drawbacks of the different types of solutions — to motivate the adopted strategy. We then discuss the conjectured security of `PEGASIS`. We recall that part our analysis is heuristic.

Perfect, (semi)good or arbitrary? We begin by comparing the probability to find the different kind of solutions, as well as the number of higher dimensional isogenies to compute in order to evaluate the action.

Form of (u, v)	Higher dimensional computation	Estimate of the probability
$\mathfrak{B}$-perfect	1 isogeny in dimension 2	$1/\operatorname{poly}(p)$
$\mathfrak{B}$-(semi)good	1 isogeny in dimension 4	$1/\operatorname{polylog}(p)$
$\mathfrak{B}$-arbitrary	3 isogenies in dimension 4	1

Figure 4.3: Comparison for $\mathfrak{B}$-perfect, $\mathfrak{B}$-(semi)good and $\mathfrak{B}$-arbitrary (u, v) regarding their probability to be encountered as well as the dimension and number of higher dimensional isogenies to compute.

We merge the semigood and good cases into a single row, due to their similarities. Determining when one is advantageous over the other — that is, when the higher probability of finding a $\mathfrak{B}$-semigood solution outweighs its additional computational cost — requires further investigation.

From Figure 4.3, one might conclude that the $\mathfrak{B}$-good case is the most suitable for practical purposes. It seems like a suitable trade-off between complexity — regarding the computation of higher dimensional isogenies — and probability to encounter such solutions. Nevertheless, it is important not to neglect the other steps when aiming for a scalable scheme.

For large p, the required number of solutions (u, v) to be tested before finding a $\mathfrak{B}$-good pair, might be substantial. In the following table, we see that the global step of solving the norm equation tends to take more than 10% of the total runtime.

Moreover, the second step — computing the kernel of the 4-dimensional isogeny — should also cost less for $\mathfrak{B}$-arbitrary (u, v) than for $\mathfrak{B}$-good ones. Finally, as the isogeny computation is quasi-linear in its length, the computation of an u-isogeny in dimension 4 should be at least twice as fast as the Step 3 — computing the final 4-dimensional isogeny.

Hence, for the highest level of security, adopting a hybrid pair (u, v), where u is $\mathfrak{B}$-good and v is $\mathfrak{B}$-arbitrary, might be a better solution than considering a $\mathfrak{B}$-good pair. So far, the Sage implementation of `PEGASIS` relies only on $\mathfrak{B}$-good pairs (u, v). Yet, additional experiments have shown that this hybrid strategy should be more efficient for p of at least around 4000 bits. Hence, asymptotically, computing the action from the first encountered pair (u, v), using three 4-dimensional isogenies, should be more efficient than the other strategies.

Prime size (bits)	Prime	Time (s)				Rerand.
		Step 1	Step 2	Step 3	Total	
508	$3 \cdot 11 \cdot 2^{503} - 1$	0.0969	0.477	0.960	1.53	0.17
1008	$3 \cdot 5 \cdot 2^{1004} - 1$	0.212	1.16	2.84	4.21	0.07
1554	$3^2 \cdot 2^{1551} - 1$	1.19	2.85	6.49	10.5	1.53
2031	$3 \cdot 17 \cdot 2^{2026} - 1$	1.68	8.34	11.3	21.3	0.70
4089	$3^2 \cdot 7 \cdot 2^{4084} - 1$	15.6	52.8	53.5	122	0.41

Table 4.2: Time our SageMath 10.5 PEGASIS implementation takes to evaluate one group action at different prime sizes using the $\mathfrak{B}$-good strategy, measured in wall-clock seconds on an Intel Core i5-1235U CPU with maximal clock speed of 4.0 GHz. The last column indicates the number of ideal-class rerandomizations required to find a solution in Step 1. These results are averaged over 100 runs. Step 1 is the time used to solve the norm equation, Step 2 is the time used to derive the kernel of the dimension 4 isogeny, and Step 3 is the time used to compute the dimension 4 isogeny. This table is borrowed from [DEF$^+$25a].

What should be the set $\mathfrak{B}$? Another key consideration when designing PEGASIS concerns the set of primes $\mathfrak{B}$. According to the above discussion, for primes p up to 4000 bits, searching for $\mathfrak{B}$-good pairs should be the most efficient strategy. Furthermore, Remark 4.3.14 suggests that primes congruent to 3 modulo 4 should be preferred. But how many should be included?

This leads to a trade-off: adding more primes to factor out makes solving the CLAPOTI equation easier — since it increases the number of candidate solutions and the likelihood that one will be suitable — but on the other hand, it slows down the computation of the $\mathfrak{B}$-smooth isogenies — since they are possibly more of them and of higher norm. Nevertheless, to improve the number of candidate solutions, one can also rely on the rerandomisation process, see Remark 4.3.24.

At the end, the following set of parameters has been chosen to minimise the first two steps of PEGASIS. In particular, all the runtime results presented in the different tables are using the corresponding parameters from Figure 4.4.

Parameter set	f	c	$\mathfrak{B}$
500	503	33	2, 3, 7, 11, 13
1000	1004	15	2, 3, 5, 7, 11
1500	1551	9	2, 3, 5, 11
2000	2026	51	2, 3, 7, 11, 17
4000	4084	63	2, 3, 7, 11, 17, 19

Figure 4.4: Parameter sets used in our implementation. The prime p is of the form $p = c2^f - 1$ and $\mathfrak{B}$ is the set of small split primes used. This table comes from [DEF$^+$25a].

Security. As mentioned in Section 1.4, there is ongoing discussion about the quantum security of CSIDH in the literature. Even though the setup of PEGASIS is not the same as the one of CSIDH, assuming that their security is similar is the best approximation one can currently do without further investigations.

In the current state of the art, the size of the prime p required to achieve NIST-I level security with CSIDH ranges from 512 bits to 4096 bits. To the best of our knowledge, PEGASIS is currently the only EGA in isogeny-based cryptography handling such large characteristics. Thus, it is currently the only group action of the field possibly matching the NIST-I security conditions for post-quantum applications.

Conclusion

We end this thesis by summarising our contributions, highlighting their context and potential improvements. We also outline several open questions suggesting future research directions.

The division algorithm. In Chapter 2, we expanded the list of powerful tools offered by higher dimensional isogenies: we provided an efficient general division algorithm for isogenies, adapted from a prior result by Damien Robert. Before the `SIDH` attacks, efficient division of isogenies was only feasible for power-smooth divisors. Our rigorous and detailed complexity analysis of this unconditional algorithm is essential for the rest of our work. It is a crucial ingredient to study the complexity of oriented problems and the equivalence between fundamental problems, and it is also relevant to other advances in the literature. We hope this contribution will have other applications as the field continues to develop.

Nevertheless, making the algorithm unconditional comes at the cost of its effectiveness. While it runs in polynomial time, it requires to perform computations in dimension 8, which is highly impractical. This leads to the following question:

Question 1. *Is it possible to provide efficient unconditional* `IsogenyDivision` *algorithm in dimension 2 or 4 with a rigorous complexity analysis?*

This question extends to the `IsogenyInterpolation` and `IdealToIsogeny` algorithms.

In any case, higher dimensional isogenies and principally polarised abelian varieties are now common objects in isogeny-based cryptography. They have been the source of several breakthroughs and significant improvements in the field. Therefore, continuing to investigate their applications remains an important research direction.

The relations between fundamental problems. Another contribution in this thesis is the unconditional equivalence between the hard problems of isogeny-based cryptography — Isogeny, EndRing, MaxOrder, OneEnd and HomModule— proven in Section 3.2. It closes the discussion, started at the early stage of the field, about the connections between these problems; at least regarding the minimal assumptions required to show their equivalence. Nevertheless, some important questions remain, such as the place of ℓ-IsogenyPath:

Question 2. *Can the ℓ-*IsogenyPath *problem reduce to the* Isogeny *problem without assuming the generalised Riemann hypothesis?*

Moreover, since the reductions presented in this work do not aim for efficiency, there is still room for improvement. Providing practical reductions between these problems might result in interesting applications, just as the reduction from Isogeny to MaxOrder did. For instance, one can think of our reduction from OneEnd to MaxOrder which develops new ways of exploiting information from the quaternion world.

Finally, the higher-dimensional concepts developed following the SIDH attacks do not only provide tools for isogeny-based cryptography in dimension 1, but also open the possibility of cryptography based on principally polarised abelian varieties in higher dimensions. This is a whole new paradigm to explore. Hence, we should ask ourselves:

Question 3. *What are the hard problems to consider in higher dimensions and how are they related?*

Answering this question might also have direct consequences on isogeny-based cryptography in dimension 1.

The worst-case to average-case reductions. In Section 3.4, we provided worst-case to average-case reductions for fundamental problems. Having such clean reductions is an important strength of isogeny-based cryptography compared to other post-quantum candidates. For instance, in lattice-based cryptography, the reductions are not as general as the isogeny-based ones: they depend on approximations factors and on the numerous parameters of the different problems. Prior to this work, several reductions were considered folklore rather than formally proven. Our contribution is to provide rigorous proofs, ready-to-use theorems, but also to highlight their current limitations.

In particular, the generalised Riemann hypothesis is still necessary in two cases. First, reductions involving the ℓ-IsogenyPath problem require this assumption. It seems unlikely that one could possibly solve this issue in the worst-case to average-case scenario without answering Question 2. Second, as discussed in Section 3.4, the method developed to reduce OneEnd to MaxOrder in Proposition 3.2.7 cannot be applied in the worst-case to average-case reduction, leading to the question:

Question 4. *Can we provide a reduction to the* MaxOrder *problem in the average case without assuming the generalised Riemann hypothesis?*

One could also extend this work to oriented problems. From the results summarised in Figure 4.1, it should be possible to provide several worst-case to average-case reductions. However, some reductions are proven under the generalised Riemann hypothesis — even the effective algorithms to sample random ideals [CJS14, Theorem 2.1], which are crucial to randomise instances, are only proven under **GRH** — while others are quantum reductions. Hence, there might be improvements to be made in this direction.

The complexity analysis of oriented problems In Section 4.2, we proved a rigorous complexity analysis of the $\mathfrak{O}$-Vectorisation problem and solved the Primitivisation problem, which yielded the first rigorous resolution of the EndRing problem given a non-trivial endomorphism. This contributes to a better understanding of the security foundations of oriented isogeny-based cryptography.

Possible directions for improving this result include removing the dependence on **GRH** or the factorisation step. Additionally, we recall that the concrete complexity of quantum algorithm solving problems such as $\mathfrak{O}$-Vectorisation remains under active discussion. Hence, investigating more deeply their complexity is an important research line.

It might also be fruitful to explore the other oriented problems and their relations with the new higher dimensional tools in mind.

The PEGASIS practical EGA. The final contribution we mention here is the introduction of the practical effective group action PEGASIS, which opens several directions for future research. Thanks to the choice of CSIDH-like parameters optimised to solve the norm equation from the CLAPOTI algorithm, it outperforms the current state of the art. Its proof-of-concept implementation in Sage is 100 times faster than `KLaPoTi`, the only other EGA without precomputation, also considering its proof-of-concept implementation in Sage. This first `PEGASIS` implementation is also more efficient than the EGAs using precomputation; which are not scalable to higher security levels. A more low level implementation would obtain an even larger efficiency gain.

Let us outline three different directions to pursue in future work:

- Exploring more deeply the theoretical foundations of the scheme. A better understanding of the norm equation and how to exploit its solutions might lead to improved algorithms.
- Providing a low-level optimised implementation to evaluate how promising the scheme truly is. A concrete objective will be to achieve efficiency comparable to the state-of-the-art implementations of the `CSIDH` framework.
- Exploring advanced cryptographic constructions. For instance, we can compare the advantages of digital signatures based on `PEGASIS` with those from the `SQIsign`-family. The former is expected to be less efficient, but more suitable for advanced schemes, such as threshold signatures, due to its group action framework.

Bibliography

[ABE$^+$24] Bill Allombert, Jean-François Biasse, Jonathan Komada Eriksen, Péter Kutas, Chris Leonardi, Aurel Page, Renate Scheidler, and Márton Tot Bagi. PEARL-SCALLOP: Parameter extension applicable in real-life SCALLOP. Cryptology ePrint Archive, Report 2024/1744, 2024.

[ACD$^+$24] Sarah Arpin, James Clements, Pierrick Dartois, Jonathan Komada Eriksen, Péter Kutas, and Benjamin Wesolowski. Finding orientations of supersingular elliptic curves and quaternion orders. *Designs, Codes and Cryptography*, 92(11):3447–3493, 2024.

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

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

[ADMP20] Navid Alamati, Luca De Feo, Hart Montgomery, and Sikhar Patranabis. Cryptographic group actions and applications. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2020, Part II*, volume 12492 of *Lecture Notes in Computer Science*, pages 411–439, Daejeon, South Korea, December 7–11, 2020. Springer, Cham, Switzerland.

[Bac90] Eric Bach. Explicit bounds for primality testing and related problems. *Mathematics of Computation*, 55(191):355–380, 1990.

[BCC$^+$23] Andrea Basso, Giulio Codogni, Deirdre Connolly, Luca De Feo, Tako Boris Fouotsa, Guido Maria Lido, Travis Morrison, Lorenz Panny, Sikhar Patranabis, and Benjamin Wesolowski. Supersingular curves you can trust. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023, Part II*, volume 14005 of *Lecture Notes in Computer Science*, pages 405–437, Lyon, France, April 23–27, 2023. Springer, Cham, Switzerland.

[BCNE$^+$19] Efrat Bank, Catalina Camacho-Navarro, Kirsten Eisenträger, Travis Morrison, and Jennifer Park. Cycles in the supersingular l-isogeny graph and corresponding endomorphisms. In Jennifer S. Balakrishnan, Amanda Folsom, Matilde Lalín, and Michelle Manes, editors, *Research Directions in Number Theory*, pages 41–66, Cham, 2019. Springer International Publishing.

[BDD$^+$24] Andrea Basso, Pierrick Dartois, Luca De Feo, Antonin Leroux, Luciano Maino, Giacomo Pope, Damien Robert, and Benjamin Wesolowski. SQIsign2D-West - the fast, the small, and the safer. In Kai-Min Chung and

Yu Sasaki, editors, *Advances in Cryptology – ASIACRYPT 2024, Part III*, volume 15486 of *Lecture Notes in Computer Science*, pages 339–370, Kolkata, India, December 9–13, 2024. Springer, Singapore, Singapore.

[BDGJ20] Colin Boyd, Gareth T. Davies, Kristian Gjøsteen, and Yao Jiang. Fast and secure updatable encryption. In Daniele Micciancio and Thomas Ristenpart, editors, *Advances in Cryptology – CRYPTO 2020, Part I*, volume 12170 of *Lecture Notes in Computer Science*, pages 464–493, Santa Barbara, CA, USA, August 17–21, 2020. Springer, Cham, Switzerland.

[BJS14] Jean-François Biasse, David Jao, and Anirudh Sankar. A quantum algorithm for computing isogenies between supersingular elliptic curves. In Willi Meier and Debdeep Mukhopadhyay, editors, *Progress in Cryptology - INDOCRYPT 2014: 15th International Conference in Cryptology in India*, volume 8885 of *Lecture Notes in Computer Science*, pages 428–442, New Delhi, India, December 14–17, 2014. Springer, Cham, Switzerland.

[BKM$^+$21] Andrea Basso, Péter Kutas, Simon-Philipp Merz, Christophe Petit, and Antonio Sanso. Cryptanalysis of an oblivious PRF from supersingular isogenies. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2021, Part I*, volume 13090 of *Lecture Notes in Computer Science*, pages 160–184, Singapore, December 6–10, 2021. Springer, Cham, Switzerland.

[BKV19] Ward Beullens, Thorsten Kleinjung, and Frederik Vercauteren. CSI-FiSh: Efficient isogeny based signatures through class group computations. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology – ASIACRYPT 2019, Part I*, volume 11921 of *Lecture Notes in Computer Science*, pages 227–247, Kobe, Japan, December 8–12, 2019. Springer, Cham, Switzerland.

[BM25] Andrea Basso and Luciano Maino. POKÉ: A compact and efficient PKE from higher-dimensional isogenies. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology – EUROCRYPT 2025, Part II*, volume 15602 of *Lecture Notes in Computer Science*, pages 94–123, Madrid, Spain, May 4–8, 2025. Springer, Cham, Switzerland.

[BP89] Johannes Buchmann and Michael Pohst. Computing a lattice basis from a system of generating vectors. In *Eurocal'87: European Conference on Computer Algebra Leipzig, GDR, June 2–5, 1987 Proceedings*, pages 54–63. Springer, 1989.

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

[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 Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 893–902. SIAM, 2016.

[BY91] Gilles Brassard and Moti Yung. One-way group actions. In Alfred J. Menezes and Scott A. Vanstone, editors, *Advances in Cryptology – CRYPTO'90*, volume 537 of *Lecture Notes in Computer Science*, pages 94–107, Santa Barbara, CA, USA, August 11–15, 1991. Springer Berlin Heidelberg, Germany.

[Cas96] John William Scott Cassels. *An introduction to the geometry of numbers.* Springer Science & Business Media, 1996.

[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*, pages 111–129, Paris, France, April 15–17, 2020. Springer, Cham, Switzerland.

[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, Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 423–447, Lyon, France, April 23–27, 2023. Springer, Cham, Switzerland.

[CGL06] Denis Charles, Eyal Goren, and Kristin Lauter. Cryptographic hash functions from expander graphs. Cryptology ePrint Archive, Report 2006/021, 2006.

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

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

[CKMZ22] Tímea Csahók, Péter Kutas, Mickaël Montessinos, and Gergely Zábrádi. Explicit isomorphisms of quaternion algebras over quadratic global fields. *Research in Number Theory*, 8(4):77, 2022.

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

[CLM$^+$18] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes. CSIDH: An efficient post-quantum commutative group action. In Thomas Peyrin and Steven Galbraith, editors, *Advances in Cryptology – ASIACRYPT 2018, Part III*, volume 11274 of *Lecture Notes in Computer Science*, pages 395–427, Brisbane, Queensland, Australia, December 2–6, 2018. Springer, Cham, Switzerland.

[CLP24] Mingjie Chen, Antonin Leroux, and Lorenz Panny. SCALLOP-HD: Group action from 2-dimensional isogenies. In Qiang Tang and Vanessa Teague, editors, *PKC 2024: 27th International Conference on Theory and Practice of Public Key Cryptography, Part III*, volume 14603 of *Lecture Notes in Computer Science*, pages 190–216, Sydney, NSW, Australia, April 15–17, 2024. Springer, Cham, Switzerland.

[Cor07] Giuseppe Cornacchia. *Su di un metodo per la risoluzione in numeri interi dell'equazione...* Giornale di Matematiche di Battaglini, 1907.

[Cou06] Jean-Marc Couveignes. Hard homogeneous spaces. Cryptology ePrint Archive, Report 2006/291, 2006.

[Cox13] David A. Cox. *Primes of the Form $x^2 + ny^2$: Fermat, Class Field Theory, and Complex Multiplication.* Wiley, 2nd edition edition, 2013.

[CP25] Mingjie Chen and Christophe Petit. Computing the endomorphism ring of a supersingular elliptic curve from a full rank suborder. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology – EUROCRYPT 2025, Part VI*, volume 15606 of *Lecture Notes in Computer Science*, pages 446–474, Madrid, Spain, May 4–8, 2025. Springer, Cham, Switzerland.

[CSCJR22] Jorge Chávez-Saab, Jesús-Javier Chi-Domínguez, Samuel Jaques, and Francisco Rodríguez-Henríquez. The SQALE of CSIDH: sublinear Vélu quantum-resistant isogeny action with low exponents. *Journal of Cryptographic Engineering*, 12(3):349–368, September 2022.

[Dar24] Pierrick Dartois. Fast computation of 2-isogenies in dimension 4 and cryptographic applications. Cryptology ePrint Archive, Report 2024/1180, 2024.

[DD22] Pierrick Dartois and Luca De Feo. On the security of OSIDH. In Goichiro Hanaoka, Junji Shikata, and Yohei Watanabe, editors, *PKC 2022: 25th International Conference on Theory and Practice of Public Key Cryptography, Part I*, volume 13177 of *Lecture Notes in Computer Science*, pages 52–81, Virtual Event, March 8–11, 2022. Springer, Cham, Switzerland.

[DDF$^+$21] Luca De Feo, Cyprien Delpech de Saint Guilhem, Tako Boris Fouotsa, Péter Kutas, Antonin Leroux, Christophe Petit, Javier Silva, and Benjamin Wesolowski. Séta: Supersingular encryption from torsion attacks. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2021, Part IV*, volume 13093 of *Lecture Notes in Computer Science*, pages 249–278, Singapore, December 6–10, 2021. Springer, Cham, Switzerland.

[DEF$^+$25a] Pierrick Dartois, Jonathan Komada Eriksen, Tako Boris Fouotsa, Arthur Herlédan Le Merdy, Riccardo Invernizzi, Damien Robert, Ryan Rueger, Frederik Vercauteren, and Benjamin Wesolowski. PEGASIS: Practical effective class group action using 4-dimensional isogenies. Cryptology ePrint Archive, Paper 2025/401, 2025. Accepted in the conference Advances in Cryptology – CRYPTO 2025 of the IACR.

[DEF$^+$25b] Pierrick Dartois, Jonathan Komada Eriksen, Tako Boris Fouotsa, Arthur Herlédan Le Merdy, Riccardo Invernizzi, Damien Robert, Ryan Rueger, Frederik Vercauteren, and Benjamin Wesolowski. PEGASIS: Practical effective class group action using 4-dimensional isogenies. Cryptology ePrint Archive, Report 2025/401, 2025.

[Deu41] Max Deuring. Die typen der multiplikatorenringe elliptischer funktionenkörper. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 14:197–272, 1941.

[DF90] Yvo Desmedt and Yair Frankel. Threshold cryptosystems. In Gilles Brassard, editor, *Advances in Cryptology – CRYPTO'89*, volume 435 of *Lecture Notes in Computer Science*, pages 307–315, Santa Barbara, CA, USA, August 20–24, 1990. Springer, New York, USA.

[DFK$^+$23] Luca De Feo, Tako Boris Fouotsa, Péter Kutas, Antonin Leroux, Simon-Philipp Merz, Lorenz Panny, and Benjamin Wesolowski. SCALLOP: Scaling the CSI-FiSh. In Alexandra Boldyreva and Vladimir Kolesnikov, editors, *PKC 2023: 26th International Conference on Theory and Practice of Public Key Cryptography, Part I*, volume 13940 of *Lecture Notes in Computer Science*, pages 345–375, Atlanta, GA, USA, May 7–10, 2023. Springer, Cham, Switzerland.

[DG16] Christina Delfs and Steven D. Galbraith. Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. *Designs, Codes and Cryptography*, 78(2):425–440, 2016.

[DG19] Luca De Feo and Steven D. Galbraith. SeaSign: Compact isogeny signatures from class group actions. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019, Part III*, volume 11478 of *Lecture Notes in Computer Science*, pages 759–789, Darmstadt, Germany, May 19–23, 2019. Springer, Cham, Switzerland.

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[DKL$^+$20] Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: Compact post-quantum signatures from quaternions and isogenies. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2020, Part I*, volume 12491 of *Lecture Notes in Computer Science*, pages 64–93, Daejeon, South Korea, December 7–11, 2020. Springer, Cham, Switzerland.

[DLLW23] Luca De Feo, Antonin Leroux, Patrick Longa, and Benjamin Wesolowski. New algorithms for the Deuring correspondence - towards practical and secure SQISign signatures. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023, Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 659–690, Lyon, France, April 23–27, 2023. Springer, Cham, Switzerland.

[DLRW24] Pierrick Dartois, Antonin Leroux, Damien Robert, and Benjamin Wesolowski. SQIsignHD: New dimensions in cryptography. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology – EUROCRYPT 2024, Part I*, volume 14651 of *Lecture Notes in Computer Science*, pages 3–32, Zurich, Switzerland, May 26–30, 2024. Springer, Cham, Switzerland.

[DMPR24] Pierrick Dartois, Luciano Maino, Giacomo Pope, and Damien Robert. An algorithmic approach to (2, 2)-isogenies in the theta model and applications to isogeny-based cryptography. In Kai-Min Chung and Yu Sasaki, editors, *Advances in Cryptology – ASIACRYPT 2024, Part III*, volume 15486 of *Lecture Notes in Computer Science*, pages 304–338, Kolkata, India, December 9–13, 2024. Springer, Singapore, Singapore.

[Dud08] Underwood Dudley. *Elementary number theory.* Courier Corporation, 2008.

[EHL$^+$18] Kirsten Eisenträger, Sean Hallgren, Kristin E. 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, Part III*, volume 10822 of *Lecture Notes in Computer Science*, pages 329–368, Tel Aviv, Israel, April 29 – May 3, 2018. Springer, Cham, Switzerland.

[EL24] Jonathan Komada Eriksen and Antonin Leroux. Computing orientations from the endomorphism ring of supersingular curves and applications. *IACR Communications in Cryptology (CiC)*, 1(3):5, 2024.

[ElG84] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology – CRYPTO'84*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18, Santa Barbara, CA, USA, August 19–23, 1984. Springer Berlin Heidelberg, Germany.

[ES24] Kirsten Eisenträger and Gabrielle Scullard. Connecting kani's lemma and path-finding in the bruhat-tits tree to compute supersingular endomorphism rings. *CoRR*, abs/2402.05059, 2024.

[GLM24] Steven D. Galbraith, Yi-Fu Lai, and Hart Montgomery. A simpler and more efficient reduction of DLog to CDH for abelian group actions. In Qiang Tang and Vanessa Teague, editors, *PKC 2024: 27th International Conference on Theory and Practice of Public Key Cryptography, Part III*, volume 14603 of *Lecture Notes in Computer Science*, pages 36–60, Sydney, NSW, Australia, April 15–17, 2024. Springer, Cham, Switzerland.

[GPS17] Steven D. Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017, Part I*, volume 10624 of *Lecture Notes in Computer Science*, pages 3–33, Hong Kong, China, December 3–7, 2017. Springer, Cham, Switzerland.

[GPSV21] Steven Galbraith, Lorenz Panny, Benjamin Smith, and Frederik Vercauteren. Quantum Equivalence of the DLP and CDHP for Group Actions. *Mathematical Cryptology*, 1(1):40–44, 2021.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery.

[Har13] Robin Hartshorne. *Algebraic geometry*, volume 52. Springer Science & Business Media, 2013.

[HM89] James L Hafner and Kevin S McCurley. A rigorous subexponential algorithm for computation of class groups. *Journal of the American mathematical society*, 2(4):837–850, 1989.

[HW25a] Arthur Herlédan Le Merdy and Benjamin Wesolowski. The supersingular endomorphism ring problem given one endomorphism. *IACR Communications in Cryptology*, 2(1), 2025.

[HW25b] Arthur Herlédan Le Merdy and Benjamin Wesolowski. Unconditional foundations for supersingular isogeny-based cryptography. Cryptology ePrint Archive, Paper 2025/271, 2025.

[JD11] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Bo-Yin Yang, editor, *Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011*, pages 19–34, Tapei, Taiwan, November 29 – December 2 2011. Springer Berlin Heidelberg, Germany.

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

[KLLQ23] Shuichi Katsumata, Yi-Fu Lai, Jason T. LeGrow, and Ling Qin. CSI-Otter: Isogeny-based (partially) blind signatures from the class group action with a twist. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology – CRYPTO 2023, Part III*, volume 14083 of *Lecture Notes in Computer Science*, pages 729–761, Santa Barbara, CA, USA, August 20–24, 2023. Springer, Cham, Switzerland.

[KLPT14] David Kohel, Kristin Lauter, Christophe Petit, and Jean-Pierre Tignol. On the quaternion-isogeny path problem. *LMS Journal of Computation and Mathematics*, 17(A):418–432, 2014.

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

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

[KV10] Markus Kirschmer and John Voight. Algorithmic Enumeration of Ideal Classes for Quaternion Orders. *SIAM Journal on Computing*, 39(5):1714–1747, 2010.

[Lan09] Edmund Landau. *Über die Einteilung der positiven ganzen Zahlen in vier Klassen nach der Mindestzahl der zu ihrer additiven Zusammensetzung erforderlichen Quadrate.* B. G. Teubner, 1909.

[Ler22] Antonin Leroux. *Quaternion Algebra and isogeny-based cryptography.* PhD Thesis, PhD thesis, Ecole doctorale de l'Institut Polytechnique de Paris, 2022.

[Ler25] Antonin Leroux. Verifiable random function from the deuring correspondence and higher dimensional isogenies. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology – EUROCRYPT 2025, Part VII*, volume 15607 of *Lecture Notes in Computer Science*, pages 167–194, Madrid, Spain, May 4–8, 2025. Springer, Cham, Switzerland.

[LO77] Jeffrey C Lagarias and Andrew M Odlyzko. Effective versions of the chebotarev density theorem. In *Algebraic number fields: L-functions and Galois properties (Proc. Sympos., Univ. Durham, Durham, 1975)*, volume 7, pages 409–464, 1977.

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

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

[Mam24] Maher Mamah. The supersingular isogeny path and endomorphism ring problems: Unconditional reductions. Cryptology ePrint Archive, Report 2024/1569, 2024.

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

[MMP$^+$23] Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, and Benjamin Wesolowski. A direct key recovery attack on SIDH. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023, Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 448–471, Lyon, France, April 23–27, 2023. Springer, Cham, Switzerland.

[MOT20] Tomoki Moriya, Hiroshi Onuki, and Tsuyoshi Takagi. SiGamal: A supersingular isogeny-based PKE and its application to a PRF. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2020, Part II*, volume 12492 of *Lecture Notes in Computer Science*, pages 551–580, Daejeon, South Korea, December 7–11, 2020. Springer, Cham, Switzerland.

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

[MZ22] Hart Montgomery and Mark Zhandry. Full quantum equivalence of group action DLog and CDH, and more. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022, Part I*, volume 13791 of *Lecture Notes in Computer Science*, pages 3–32, Taipei, Taiwan, December 5–9, 2022. Springer, Cham, Switzerland.

[NIS16] NIST. National institute of standards and technology, December 2016. Post-Quantum Cryptography Standardization, `https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Post-Quantum-Cryptography-Standardization`.

[NO24] Kohei Nakagawa and Hiroshi Onuki. QFESTA: Efficient algorithms and parameters for FESTA using quaternion algebras. In Leonid Reyzin and Douglas Stebila, editors, *Advances in Cryptology – CRYPTO 2024, Part V*, volume 14924 of *Lecture Notes in Computer Science*, pages 75–106, Santa Barbara, CA, USA, August 18–22, 2024. Springer, Cham, Switzerland.

[NS09] Phong Q Nguyen and Damien Stehlé. Low-dimensional lattice basis reduction revisited. *ACM Transactions on algorithms (TALG)*, 5(4):1–48, 2009. Publisher: ACM New York, NY, USA.

[ON24] Hiroshi Onuki and Kohei Nakagawa. Ideal-to-isogeny algorithm using 2-dimensional isogenies and its application to SQIsign. In Kai-Min Chung and Yu Sasaki, editors, *Advances in Cryptology – ASIACRYPT 2024, Part III*, volume 15486 of *Lecture Notes in Computer Science*, pages 243–271, Kolkata, India, December 9–13, 2024. Springer, Singapore, Singapore.

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

[Piz80] Arnold K Pizer. An algorithm for computing modular forms on $\Gamma_0(N)$. *Journal of Algebra*, 64(2):340–390, 1980.

[Piz90] Arnold K Pizer. Ramanujan graphs and hecke operators. *Bulletin of the American Mathematical Society*, 23(1):127–137, 1990.

[Pom87] Carl Pomerance. Fast, rigorous factorization and discrete logarithm algorithms. In *Discrete algorithms and complexity (Kyoto, 1986)*, volume 15 of *Perspect. Comput.*, pages 119–143. Academic Press, Boston, MA, 1987.

[PPS24] Lorenz Panny, Christophe Petit, and Miha Stopar. KLaPoTi: An asymptotically efficient isogeny group action from 2-dimensional isogenies. Cryptology ePrint Archive, Report 2024/1844, 2024.

[PR23] Aurel Page and Damien Robert. Introducing clapoti(s): Evaluating the isogeny class group action in polynomial time. Cryptology ePrint Archive, Report 2023/1766, 2023.

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

[PW24] Aurel Page and Benjamin Wesolowski. The supersingular endomorphism ring and one endomorphism problems are equivalent. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology – EUROCRYPT 2024, Part VI*, volume 14656 of *Lecture Notes in Computer Science*, pages 388–417, Zurich, Switzerland, May 26–30, 2024. Springer, Cham, Switzerland.

[RA05] Jorge L. Ramírez Alfonsín. *The Diophantine Frobenius Problem.* Oxford University Press, 12 2005.

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

[Rob22a] Damien Robert. Evaluating isogenies in polylogarithmic time. Cryptology ePrint Archive, Report 2022/1068, 2022.

[Rob22b] Damien Robert. Some applications of higher dimensional isogenies to elliptic curves (overview of results). Cryptology ePrint Archive, Report 2022/1704, 2022.

[Rob23a] Damien Robert. Breaking SIDH in polynomial time. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023, Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 472–503, Lyon, France, April 23–27, 2023. Springer, Cham, Switzerland.

[Rob23b] Damien Robert. The geometric interpretation of the tate pairing and its applications. Cryptology ePrint Archive, Report 2023/177, 2023.

[Rob24] Damien Robert. On the efficient representation of isogenies (a survey). Cryptology ePrint Archive, Report 2024/1071, 2024.

[RS06] Alexander Rostovtsev and Anton Stolbunov. Public-Key Cryptosystem Based On Isogenies. Cryptology ePrint Archive, Report 2006/145, 2006.

[Ró92] Lajos Rónyai. Algorithmic properties of maximal orders in simple algebras over Q. *Computational Complexity*, 2(3):225–243, September 1992.

[Sch90] Claus-Peter Schnorr. Efficient identification and signatures for smart cards. In Gilles Brassard, editor, *Advances in Cryptology – CRYPTO'89*, volume 435 of *Lecture Notes in Computer Science*, pages 239–252, Santa Barbara, CA, USA, August 20–24, 1990. Springer, New York, USA.

[Sch95] René Schoof. Counting points on elliptic curves over finite fields. *Journal de théorie des nombres de Bordeaux*, 7(1):219–254, 1995.

[Sho94] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th Annual Symposium on Foundations of Computer Science*, pages 124–134, Santa Fe, NM, USA, November 20–22, 1994. IEEE Computer Society Press.

[Sho97] Victor Shoup. Lower bounds for discrete logarithms and related problems. In Walter Fumy, editor, *Advances in Cryptology – EUROCRYPT'97*, volume 1233 of *Lecture Notes in Computer Science*, pages 256–266, Konstanz, Germany, May 11–15, 1997. Springer Berlin Heidelberg, Germany.

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

[Sil09] Joseph H. Silverman. *The Arithmetic of Elliptic Curves*, volume 106 of *Graduate Texts in Mathematics*. Springer New York, New York, NY, 2009.

[vdW56] Bartel Leendert van der Waerden. Die Reduktionstheorie der positiven quadratischen Formen. *Acta Mathematica*, 96(1):265–309, 1956. Publisher: Kluwer Academic Publishers Dordrecht.

[Voi13] John Voight. Identifying the Matrix Ring: Algorithms for Quaternion Algebras and Quadratic Forms. In Krishnaswami Alladi, Manjul Bhargava, David Savitt, and Pham Huu Tiep, editors, *Quadratic and Higher Degree Forms*, pages 255–298. Springer New York, New York, NY, 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.

[Wat69] William C. Waterhouse. Abelian varieties over finite fields. *Annales scientifiques de l'École Normale Supérieure*, Ser. 4, 2(4):521–560, 1969.

[Wes22a] Benjamin Wesolowski. Orientations and the supersingular endomorphism ring problem. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology – EUROCRYPT 2022, Part III*, volume 13277 of *Lecture Notes in Computer Science*, pages 345–371, Trondheim, Norway, May 30 – June 3, 2022. Springer, Cham, Switzerland.

[Wes22b] Benjamin Wesolowski. The supersingular isogeny path and endomorphism ring problems are equivalent. In *62nd Annual Symposium on Foundations of Computer Science*, pages 1100–1111, Denver, CO, USA, February 7–10, 2022. IEEE Computer Society Press.

[Wes24] Benjamin Wesolowski. *Random Walks in Number-theoretic Cryptology*. HDR Thesis, ENS Lyon, 2024.

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