# Proofs of Isogeny Knowledge and Application to Post-quantum One-Time Verifiable Random Function

Antonin Leroux

DGA, INRIA and LIX, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris

**Abstract.** In this paper, we introduce a new method to prove the knowledge of an isogeny of given degree between two supersingular elliptic curves. Our approach can be extended to verify the evaluation of the secret isogeny on some points of the domain. The main advantage of this new proof of knowledge is its compactness which is orders of magnitude better than existing proofs of isogeny knowledge. The principle of our method is to reveal some well-chosen endomorphisms and does not constitute a zero-knowledge proof. However, when the degree is a large prime, we can introduce a new hardness assumption upon which we build the first verifiable random function (VRF) based on isogenies. Our protocol can be seen as a generalization of the BLS-style classical construction from elliptic curves and achieves one-time pseudo-randomness in the random oracle model. We propose concrete parameters for this new scheme which reach post-quantum NIST-1 level of security. Our VRF has an overall cost (proof size, key size and output size) of roughly 1KB, which is shorter than all the other post-quantum instantiations based on lattices. In the process, we also develop several algorithmic tools to solve norm equations over quaternion orders that may be of independent interest.

## 1 Introduction

Isogeny-based cryptography has received an increasing amount of interest due to its presumed resistance to quantum computers. As the variety of primitive achievable from isogeny is expanding, new problems are arising. The problem of proving the knowledge of an isogeny or verify an isogeny's evaluation is appearing in various contexts such as SIDH [28] key validation, several signatures [43,17,6,29], VDFs [19] and the recent oblivious PRF protocol from [8]. The existing proof techniques are only working in the SIDH [28] or CSIDH [10] setting and are neither compact nor efficient as they are built from low-soundness subprotocols. The spirit of our approach rather follows the recent SQISign construction from [18], a very compact and relatively efficient signature scheme based on a high-soundness identification protocol. The principle behind SQISign is to reveal some well-chosen isogeny to prove the knowledge of the endomorphism ring of the public key curve. We extend this idea and propose to reveal some well-chosen endomorphisms to prove the knowledge of an isogeny between two curves.

While it has been shown [31,20,21] that computing random endomorphisms of supersingular curves breaks the security of any isogeny-based scheme, we argue that this is not necessarily the case if the endomorphisms are selected carefully. The key ingredient in SQISign is the generalized KLPT algorithm which allows the signer to compute a smooth isogeny between two given curves. Similarly, for our new proof technique we need to reveal endomorphisms of smooth norm inside a specific order. To that end, we develop a new algorithm to solve norm equations in a large class of quaternion orders. These types of equations and the necessity to find small solutions efficiently has appeared in several recent works [32,18,20,24,37,36,33]. We expand the range of known algorithms to a larger variety of quaternion orders.

Building upon our new proof technique, we introduce the first isogeny Verifiable Random Function (VRF) scheme. A verifiable random function is a way to generate authenticated randomness in a verifiable manner. This notion was introduced in [34] and has recently found concrete applications in blockchain consensus [12,25,16]. The two main VRF constructions are based on classical ECC (ECVRF [35]) and pairing-based cryptography (BLS-VRF [7]). Both are vulnerable to quantum computers. This weakness is a major concern for blockchain applications as it gives the possibility to a quantum attacker to rewrite history, thus violating one of the most important principles of blockchain. The impact of such an attack on long-term security is described more precisely in the introduction of [22].

VRFs are often constructed from unique signatures, and the additional uniqueness property is what makes the VRF construction difficult compared to digital signatures. Until this work, the only post-quantum VRFs were constructed from lattices, which proves to be quite a challenge given the inherently noisy nature of lattices. This explains why the resulting constructions have either very large size [26,42] or are restricted to a limited number of signatures under the same key [22]. For blockchain applications, the size is particularly crucial as VRF keys, outputs and proofs must fit into a block of fixed length. In this regard, the only practical protocol is the recent one introduced in [22].

These two constraints explain why isogenies are a good fit for VRFs as they provide the most-compact post-quantum protocols, and their algebraic nature makes exact and unique computation easier. Unfortunately, isogenies are also less flexible than lattices, and achieving a one-time construction already raises several technical challenges. We introduce different new ideas and algorithms to overcome these obstacles.

The main difficulty in translating existing Diffie-Hellman-based VRFs to the SIDH [28] or CSIDH [10] setting is the lack of an efficient hash into the set of supersingular curves (i.e. a way to produce a random supersingular elliptic curve without any additional information on it, such as its endomorphism ring) as can be done for elliptic curve points (see for instance [5]). We build a VRF scheme upon our compact proof of isogeny knowledge applied to a seemingly natural generalisation of the BLS-VRF protocol.

Our contributions can be summarized in the following manner:

– A new compact proof of isogeny knowledge and proof of evaluation by revealing a special suborder of the endomorphism ring.
– A new one-time VRF protocol from isogenies based on a new hardness assumption (Problem 1) in the random oracle model.
– New algorithms to find elements of smooth norm in these quaternion suborders.

This paper is organized as follows. Section 2 introduces preliminaries on VRFs and the Deuring correspondence. In Section 3, we outline a new proof method for isogeny knowledge and evaluation. Our VRF construction is introduced in Section 4 and analyzed in Section 5. In Section 6, we present new algorithms to find smooth elements in a large class of quaternion orders along with several other technical sub-algorithms and proofs. In Section 7 we look at parameters, size and efficiency for the proposed VRF construction. Finally, in Section 8, we sketch ideas to remove the one-time restriction and open some prospects for other applications of our new proof method.

*Acknowledgements* We thank Benjamin Smith and Luca De Feo for valuable feedback and proofreading an earlier version of this work.

## 2 Background material

We call *negligible* a function $f : \mathbb{Z}_{>0} \to \mathbb{R}_{>0}$ if it is asymptotically dominated by $O(x^{-n})$ for all $n > 0$. In the analysis of a probabilistic algorithm, we say that an event happens with *overwhelming probability* if its probability of failure is a negligible function of the length of the input.

In a distinguishing problem, the advantage of an attacker is the improvement in success probability over a random guess. The efficiency of a distinguisher can be estimated by its advantage. When all polynomial-time distinguishers have negligible advantage, we say that the problem is hard.

### 2.1 Verifiable Random Function

A Verifiable Random Function (VRF) is a way to generate authenticated randomness that can be verified. It constist of the following protocols:

– $\mathsf{SetUp}(1^\lambda)$, returns a set of public parameters $pp$ (see Definition 5).
– $\mathsf{KeyGen}(pp)$, returns a pair $(pk, sk)$ of public key and secret key from the public parameters.
– $\mathsf{VRFEval}(sk, x) = (v, \pi)$, takes the secret key $sk$ and an input $x$ and computes the output $y$ along with a proof $\pi$.
– $\mathsf{Verif}(pk, \pi, x, v)$ takes the VRF public key, proof, input and output and returns 0 or 1.

In this article, we construct a VRF satisfying the following properties:

– **Provability**: The verification always returns 1 on correctly generated proof and output from a given input.
– **One-time pseudo-randomness**: With one access to an oracle computing $\mathsf{VRFEval}(sk, x)$ for $x \neq x_0$, an adversary cannot distinguish between $\mathsf{VRFEval}(sk, x_0)$ and a random value (see Definition 1).
– **Uniqueness**: There does not exist a key and input and two pairs $(v_1, \pi_1)$ and $(v_2, \pi_2)$ with $v_1 \neq v_2$ both passing the verification (see Definition 2).
– **Unbiasability**: An adversary that can choose the key, cannot bias the output of the VRF when the input is uniformly random (see Definition 6).

The usual pseudo-randomness property allows the adversary to make a polynomial number of queries. The $k$-times variant was introduced recently in [22].

The uniqueness property can be relaxed in *computational uniqueness* (see Definition 3) where we assume that it is hard to find two (output,proof) pairs with different outputs passing the verification as opposed to *unconditional uniqueness* where there does not exist any such pairs (see Definition 2).

**Classical instantiation with ECC and natural extensions to isogenies** We outline briefly an efficient instantiation of VRF using classical ECC. More precise references can be found at [7,35]. As usual in ECC, the key pair is $(s, [s]P)$ for some point $P$ on a curve $E$. To evaluate the VRF, one uses a hash function $h : \{0,1\}^\star \to E(k)$ to hash into the curve and then output $[s]h(x)$ on input $x$. A proof can be computed with the usual tools of ECC. In BLS-VRF [7], pairings are used to verify the correctness of the computation.

The analog of this style of construction in the Diffie-Hellman settings of SIDH and CSIDH would require a way to hash into the set of supersingular curves. More precisely, the hash function into $E(k)$ allows one to generate a point $P$ in some group $G$ without knowing its discrete logarithm with respect to some generator. The generalization of this in the context of supersingular elliptic curves would be a way to generate a curve of unknown endomorphism ring or without knowing a path to some base curve $E_0$. This is a notoriously hard problem. For instance, it is not achieved by the CGL hash function [11].

Another way of generalizing these BLS-style protocols for isogenies is explored in [19,9] for verifiable delay functions and delay encryption, where the scalar multiplication by $s$ is replaced with evaluation through some secret isogeny. This is a framework that we also use in this work, as the VRF secret key is going to be an isogeny. However, the verification in [19,9] is not post-quantum secure, as it is based on pairings. We will make use of the new ideas presented in Section 3 to verify the correctness of isogeny evaluation in a post-quantum manner. However, there is a second challenge brought by the post-quantum setting: once the evaluation of a basis through the secret isogeny is known, computing the image of any point is as hard as breaking discrete logs. We don't really have a way to deal with this problem, and that is why our construction is only a one-time VRF. Potential fixes are proposed in Section 8.1 but several technical obstacles are still standing in the way of a non-restricted VRF from isogenies.

### 2.2 Elliptic curves, quaternion algebras and the Deuring correspondence

Below, we briefly expose the main features of the Deuring correspondence. For a more complete treatment of supersingular elliptic curves and quaternion algebras see [27,31,39].

The Deuring correspondence is an equivalence of categories between isogenies of supersingular elliptic curves and the left ideals over maximal order $\mathcal{O}$ of $\mathcal{B}_{p,\infty}$, inducing a bijection between conjugacy classes of supersingular $j$-invariants and maximal orders (up to equivalence) [31]. Moreover, this bijection is explicitly constructed as $E \to \mathrm{End}(E)$. Hence, given a supersingular curve $E_0$ with endomorphism ring $\mathcal{O}_0$, the pair $(E_1, \varphi)$, where $E_1$ is another supersingular elliptic curve and $\varphi : E_0 \to E_1$ is an isogeny, is sent to a left integral $\mathcal{O}_0$-ideal. The right order of this ideal is isomorphic to $\mathrm{End}(E_1)$. One way of realizing this correspondence is obtained through the kernel ideals defined in [40]. Given an integral left-$\mathcal{O}_0$-ideal I, we define the kernel of $I$ as the subgroup $E_0[I] = \{P \in E_0(\overline{\mathbb{F}}_{p^2}) : \alpha(P) = 0 \text{ for all } \alpha \in I\}$. To $I$, we associate the isogeny $\varphi_I : E_0 \to E_0/E_0[I]$. Conversely, given an isogeny $\varphi$, the corresponding *kernel ideal* is $I_\varphi = \{\alpha \in \mathcal{O}_0 \ : \ \alpha(P) = 0 \text{ for all } P \in \ker(\varphi)\}$. Sometimes, when the kernel of $\varphi$ is given as a group $G$, we also write $I_G$ for this ideal. When $G$ is cyclic and generated by a point $P$ we abuse notations by writing $I_P$. Two ideals $I, J$ are said to be *equivalent* if $I = J\beta$ for some $\beta \in B_{p,\infty}^{\times}$ and we write $I \sim J$.

The main properties of the Deuring correspondence are summarized in Table 1.

| | |
|---|---|
| Supersingular $j$-invariants over $\mathbb{F}_{p^2}$<br>$j(E)$ (up to Galois conjugacy) | Maximal orders in $B_{p,\infty}$<br>$\mathcal{O} \cong \mathrm{End}(E)$ (up to isomorpshim) |
| $(E_1, \varphi)$ with $\varphi : E \to E_1$ | $I_\varphi$ integral left $\mathcal{O}$-ideal and right $\mathcal{O}_1$-ideal |
| $\theta \in \mathrm{End}(E_0)$ | Principal ideal $\mathcal{O}\theta$ |
| $\deg(\varphi)$ | $n(I_\varphi)$ |
| $\hat{\varphi}$ | $\overline{I_\varphi}$ |
| $\varphi : E \to E_1, \psi : E \to E_1$ | Equivalent Ideals $I_\varphi \sim I_\psi$ |
| Supersingular $j$-invariants over $\mathbb{F}_{p^2}$ | $\mathrm{Cl}(\mathcal{O})$ |
| $\tau \circ \rho : E \to E_1 \to E_2$ | $I_{\tau\circ\rho} = I_\rho \cdot I_\tau$ |

**Table 1.** The Deuring correspondence, a summary from [18].

**Effective Deuring correspondence** For the concrete instantiation of our construction, we rely on the effective correspondence algorithms introduced in [18]. In particular, for the evaluation of our VRF, we will use the $\mathsf{IdealToIsogeny}_{\ell^\bullet}$ sub-algorithm to translate ideals of norm $\ell^f$ to the corresponding isogenies for some small prime $\ell$. This algorithm works by cutting the $\ell^f$-isogeny in smaller pieces of degree $\ell^{2e+\varepsilon}$ (for some small $e$ and $\varepsilon$) which can be translated into an isogeny using a good representation of the endomorphism ring of the successive

domains. This representation is obtained through an equivalent isogeny of degree $T^2$ coprime with $\ell$. This technique can be made efficient when both $\ell^e$ and $T$ divide $p^2-1$, which underlies the requirement on $p$ that we impose in Section 7.1.

## 3 Proof of isogeny knowledge and proof of evaluation

In this section, we provide a high-level description of our new method of proof. The goal is to provide some insight into the general principle of the proof. The reader solely interested in the VRF description can jump straight to Section 4.

We start with the proof of knowledge of an isogeny of given degree $D$ in Section 3.1, before looking at an adaptation of this method to verify the evaluation of an isogeny one some torsion points in Section 3.2.

### 3.1 Proof of $D$-isogeny knowledge

We target the following problem: given two supersingular curves $E_0$, $E$ and an integer $D$, prove the knowledge of an isogeny of degree $D$ between $E_0$ and $E$. The idea is to apply the method to the case where $E_0$ is a fixed curve with publicly known endomorphism ring and we want to retain some kind of secrecy on the endomorphism ring of $E$.

We base our proof system on an interesting property of endomorphism rings: given $\mathcal{O}_0 \cong \mathrm{End}(E_0)$ and $\mathcal{O} \cong \mathrm{End}(E)$, there is an embedding of $\mathfrak{O} = \mathbb{Z} + D\mathcal{O}_0$ in $\mathcal{O}$ if and only if there is an isogeny of degree $D$ connecting $E_0$ and $E$. The indirect implication comes easily from the map $[d] + [D]\alpha_0 \mapsto [d] + \varphi \circ \alpha_0 \circ \hat{\varphi}$ naturally defined from $\mathbb{Z} + D\mathcal{O}_0$ to $\mathcal{O}$ for any isogeny $\varphi : E_0 \to E$ of degree $D$. The other direction is more subtle, and we prove it in Lemma 9.

Thus, by exhibiting some endomorphisms of $\mathbb{Z} + D\mathcal{O}_0$ in $\mathcal{O}$, one can prove the existence of an isogeny of degree $D$ between $E_0$ and $E$. The main limitation of this principle is due to the fact that the embedding $\mathbb{Z} + D\mathcal{O}_0 \subset \mathcal{O}$ is closely related to $\varphi$ (as can be easily be seen from the concrete embedding map described above). In fact, this observation is at the heart of the attacks [36,33] on the SIDH key exchange and underlies the decryption process of the encryption scheme from [37]. Indeed, the knowledge of an element $\theta$ of $\mathbb{Z} + D\mathcal{O}_0$ is sufficient to recover the isogeny $\varphi$. Thus, when $D$-isogeny computations are efficient, our proof is equivalent to revealing the isogeny $\varphi$. Given this, one may wonder what the interest of our method is. The crucial point is that our proof system always has polynomial complexity in both $\log(p)$ and $\log(D)$ (see the size estimates of Section 6.3). On the other hand when $D$ is a prime number, the best algorithms to compute and evaluate $D$-isogenies run in $O(\sqrt{D})$ operations over the field of definition of the $D$-torsion (see [4]). Hence, there is an exponential gap between the two methods when $D$ is prime. In Section 4, we use our new proof method in this setting of large prime degree to construct a VRF.

### 3.2 Proving correctness of isogeny evaluation

From the method of Section 3.1, we can derive more powerful applications by specializing the endomorphisms of $\mathbb{Z}+D\mathcal{O}_0$ that we reveal. In particular, we can verify that a given cyclic subgroup $H$ is the image under $\varphi$ of another subgroup $G$. This follows from two standard results:

**Lemma 1.** *Let $E$ be a supersingular elliptic curve with endomorphism ring $\mathcal{O}$ and let $N$ be a prime. For $\alpha \in \mathcal{O}$ of norm coprime with $N$, exactly one of the following is true:*

1. *There are no cyclic subgroups of order $N$ stabilized by $\alpha$.*
2. *There is exactly one cyclic subgroup of order $N$ stabilized by $\alpha$, and $\mathrm{tr}(\alpha)^2 = 4n(\alpha) \mod N$.*
3. *There are exactly two cyclic subgroups of order $N$ stabilized by $\alpha$ and, $\alpha$ is contained in $\mathbb{Z}+I \smallsetminus \mathbb{Z}+N\mathcal{O}$ where $I$ is an integral left $\mathcal{O}$-ideal of norm $N$.*
4. *All cyclic subgroups of order $N$ are stabilized by $\alpha$, and $\alpha$ acts as a scalar on $E[N]$ and is contained in $\mathbb{Z}+N\mathrm{End}(E)$.*

*Proof.* If $G = \langle P \rangle$ is an eigenspace of $\alpha$, then there exists $\lambda \in \mathbb{Z}$ such that $\alpha(P) = [\lambda]P$. Thus, $(\alpha - \lambda)P = 0$ and $\alpha - \lambda$ is contained in $I_G$ by definition of kernel ideals and we have $\alpha \in \mathbb{Z} + I_G$. In terms of the number of eigenspaces, since $E[N] \cong \mathbb{Z}/N\mathbb{Z}^2$, it is quite clear that we have four possible situations: no eigenspaces, exactly one eigen space, exactly two distinct eigenspaces or all $E[N]$ depending on the degree and the number of solutions to the minimal polynomial. The last case where all cyclic subgroups of $E[N]$ are stabilized by $\alpha$ happens when the minimal polynomial has degree 1. In that case, $E[N]$ is in the kernel of $\alpha - \lambda$. The kernel ideal of $E[N]$ is $N\mathrm{End}(E)$ and so $\alpha \in \mathbb{Z} + N\mathrm{End}(E)$. When there is exactly one eigen space the minimal polynomial of $\alpha$ has degree 2 but only one roots. Since the minimal polynomial is $X^2 + \mathrm{tr}(\alpha)X + n(\alpha)$ we get that this situation happens when $4n(\alpha) = \mathrm{tr}(\alpha)^2 \mod N$. From there, it is easy to deduce that $\alpha = \mathrm{tr}(\alpha)/2 + \alpha_0$ and $\mathrm{tr}(\alpha_0) = n(\alpha_0) = 0 \mod N$.

**Lemma 2.** *Let $E_0, E$ be two supersingular ellitpic curves with respective endomorphism rings $\mathcal{O}_0, \mathcal{O}$ and let $\varphi : E_0 \to E$ be a $D$-isogeny, when $\alpha_0 \in \mathcal{O}_0$ satisifies $\alpha_0(G) = G$ for $G$ a cyclic subgroup of order $N$, then $\alpha = [d] + \varphi \circ \alpha_0 \circ \hat{\varphi} \in \mathcal{O}$ stabilizes $H = \varphi(G)$ for any $d \in \mathbb{Z}$.*

*Proof.* Let us write $P$ a generator of $G$ and $Q = \varphi(P)$. If $\alpha_0(G) = G$, there exists $\lambda \in \mathbb{Z}$ such that $\alpha_0(P) = [\lambda]P$. In that case, we have $\alpha(Q) = [d]Q + \varphi \circ \alpha_0 \circ \hat{\varphi}(\varphi(P)) = [d]Q + [D]\varphi((\alpha_0(P)) = [d]Q + [\lambda D]\varphi(P) = [d + \lambda D]Q$.

If we write $I_G$ for the kernel ideal corresponding to $G$, then the set of endomorphisms stabilizing $G$ is precisely $\mathbb{Z} + I_G$. From Lemma 2, we obtain that $\mathbb{Z} + D(\mathbb{Z} + I_G) \subset \mathbb{Z} + I_H$ when $H = \varphi(G)$. Additionally, by Lemma 1, when $\alpha_0 \in (\mathbb{Z} + I_G) \smallsetminus (\mathbb{Z} + N\mathcal{O}_0)$ and $\mathrm{tr}(\alpha_0)^2 \neq 4n(\alpha_0)$ we are in the case where $\alpha_0$ has exactly two eigenspaces. When $D$ is coprime with $N$ this is also the case for the endomorphisms $\alpha = [d] + \varphi \circ \alpha_0 \circ \hat{\varphi}$. Thus, given concrete endomorphisms

$\alpha \in \mathfrak{O} \setminus (\mathbb{Z} + DN\mathcal{O}_0)$, one can identify the two eigenspaces $H_1, H_2$ of $\alpha$ and determine (using the eigenvalues for instance) which one satisfies $H_i = \varphi(G)$.

More generally, we can do the same for any suborder $\mathfrak{O}_0 \subset \mathcal{O}_0$. Exploring the different kinds of suborders might lead to interesting new applications.

# 4 New post-quantum VRF from isogenies

We start by an informal description of our VRF in the next section before giving more details in Section 4.2.

## 4.1 VRF notations and global description

The notations introduced below are kept throughout the paper.

**Parameters** We start by taking a prime $p$. All elliptic curves considered are supersingular over $\mathbb{F}_{p^2}$. We do not give any constraint on the choice of $p$ now, but some requirements are going to appear (mainly for efficiency reasons). There are two additional prime numbers $D, N$ respectively the degree of the secret isogeny $\varphi$ and the order of the points sent through $\varphi$. We complete the set of parameters by a supersingular curve $E_0$ over $\mathbb{F}_{p^2}$ of known endomorphism ring $\mathcal{O}_0$ and several other parameters that we detail in Section 4.2.

**Keys** The secret key is an integral left $\mathcal{O}_0$-ideal $I$ of norm $D$, and the public key is a supersingular curve $E = E_0/E_0[I]$ together with a basis $(P_E, Q_E)$ of $E[N]$. This curve is $D$-isogenous to $E_0$ through the isogeny $\varphi$ corresponding to $I$.

**Evaluation Mechanism** On input $x$, we evaluate the VRF as follows: hash $x$ into two subgroups $G_1, G_2$ of order $N$ and compute $H_1 = \varphi(G_1), H_2 = \varphi(G_2)$. Then, hash these two groups into a final value $y$ that will be the output. In practice, we will represent these groups using projective points and the bijection $\Psi$ to avoid computing discrete logs over $E[N]$.

**Proof and Verification Protocol** Following Section 3.2, we prove correctness of the computation by revealing the embedding of the order $\mathfrak{O} = \mathbb{Z} + D((\mathbb{Z} + I_{G_1}) \cap (\mathbb{Z} + I_{G_2}))$ in $\mathrm{End}(E)$. The proof is constituted of a representation of a generating family (see Definition 4) of $\mathfrak{O}$. We represent these endomorphisms as isogenies (expressed as bitstrings as for the signature in [18]) and compute these isogenies from the quaternions using the `IdealToIsogeny` algorithm from [18]. As explained in Section 3.2, to verify the output it suffices to check that the embedding is correct and that the two subgroups $H_1$ and $H_2$ are stable under the elements of $\mathfrak{O}$. The important part for uniqueness is that all the curves admitting an embedding of $\mathfrak{O}$ are $D$-isogenous to $E_0$ (see Proposition 2).

*Remark 1.* One might wonder why we bother with two groups. Why not build the construction with only one subgroup $G$? The reason is simple: it would breach security. Indeed, doing the same construction with one subgroup would imply revealing the embedding of $\mathfrak{O} = \mathbb{Z} + D(\mathbb{Z} + I_G)$ and verifying that $\varphi(G)$ is stable under the endomorphisms of $\mathfrak{O}$. Unfortunately, this method reveals too much information. As pointed out in Section 3.2, most elements in $\mathfrak{O}$ have exactly two eigenspaces. One of those eigenspaces is always going to be $G$, by definition of $\mathfrak{O}$, but the second one will change for different endomorphisms of $\mathfrak{O}$. Hence, as soon as one endomorphism of $\mathfrak{O}$ is revealed to validate $\varphi(G)$, the adversary learns the image of another subgroup through $\varphi$ which is already enough to break pseudo-randomness as defined in Definition 1. That is why we must take a pair of distinct subgroups $G_1, G_2$ as input of our VRF.

### 4.2 Formal description

In this section, we give a formal description of the different protocols that compose our VRF. We leave some sub-protocols as black boxes for now and detail their descriptions in Section 6. We also omit the parameter generation; it will be discussed later in Section 7.1. For now, let us assume that there are four distinct primes $p, D, \ell, N$ such that $\ell^e N$ divides $p^2 - 1$, $N$ and $D$ respective sizes depend on $p$ and the level of security. There is also a curve $E_0$ over $\mathbb{F}_{p^2}$ of known endomorphism ring $\mathcal{O}_0$. The public parameters also include a basis $(P_0, Q_0)$ of $E_0[N]$ and the related kernel ideal $I_{P_0}$ together with an endomorphism $\iota \in \mathcal{O}_0$ such that $\iota(P_0) = Q_0$. We write $pp = (p, N, D, \ell, E_0, P_0, Q_0, I_{P_0}, \iota)$.

There is one small caveat to the construction outlined in Section 4.1: since $N$ is a large prime, computing discrete logarithms is very inefficient over the $N$ torsion. This fact makes hashing cyclic subgroups of order $N$ into a final output (as suggested in the description of the evaluation mechanism above) very difficult because there is no way to agree efficiently on one compact representation of the subgroup. We overcome this obstacle by making use of a bijection between the set of cyclic subgroups of order $N$ and the projective line of $\mathbb{Z}/N\mathbb{Z}$. We remind the reader that the projective line $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ is the set of pairs $(z : w) \in \mathbb{Z}/N\mathbb{Z}$ up to multiplication by a common scalar. We can make this bijection explicit by fixing a basis $P, Q$ of $E[N]$ and sending $(z : w)$ to $\langle [w]P + [z]Q \rangle$. Conversely, a group $G = \langle R \rangle$ is mapped to $(w : z)$ where $R = [w]P + [z]Q$. For any given basis $P, Q$, we write $\Psi_{P,Q}$ for this bijection.

Contrary to the set of cyclic subgroups, it is easy to hash out of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$. Indeed, any element $(w : z)$ admits the canonical representation: $(w/z : 1)$ if $z \neq 0$ and $(1 : 0)$ otherwise. A hash function from $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ to $\{0,1\}^{n_2(\lambda)}$ is obtained by extracting a bit-string from this representation and applying any standard hash function from $\{0,1\}^\star$ to $\{0,1\}^{n_2(\lambda)}$.

Thus, replacing cyclic subgroups by projective points with the bijection $\Psi$, our VRF construction produces a function from $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta$ to $\{0,1\}^{n_2(\lambda)}$ where $\Delta$ is the diagonal (pairs of the form $(x, x)$) subset of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2$ (the diagonal is removed for the security reasons explained in Remark 1). This VRF map can be seen as the composition of a permutation of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta$ with a

hash function $H : \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta \to \{0,1\}^{n_2(\lambda)}$. The core of our construction is really this permutation of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta$ that we obtain by composing the permutation on cyclic subgroups induced by $\varphi$ with the bijection $\Psi$.

We write $H : \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta \to \{0,1\}^{n_2(\lambda)}$ for the aforementioned hash function.

**Key generation** We describe the $\mathsf{KeyGen}(pp)$ algorithm:

1. Generate a random $\mathcal{O}_0$-ideal $I$ of norm $D$, corresponding to an isogeny $\varphi$ of degree $D$.
2. Compute $J \sim I$, an ideal of norm $\ell^f$.
3. Compute $\varphi_J : E_0 \to E$, the isogeny of degree $\ell^f$ corresponding to $J$.
4. Use $\varphi_J$ to compute $\varphi(P_0), \varphi(Q_0)$.
5. Sample a random matrix $B \in \mathrm{GL}_2(\mathbb{Z}/N\mathbb{Z})$ and set $P_E = [B_{1,1}]\varphi(P_0) + [B_{2,1}]\varphi(Q_0)$, $Q_E = [B_{1,2}]\varphi(P_0) + [B_{2,2}]\varphi(Q_0)$.
6. Output $(sk, pk) = (\langle I, J, \varphi_J, E, B\rangle, \langle E, P_E, Q_E\rangle)$.

**VRF evaluation** $\mathsf{VRFEval}(sk, (x_1, x_2))$:

1. Parse $sk$ as $I, J, \varphi_J, E, B$.
2. For $i \in \{1, 2\}$:
   (a) Compute $G_i = \Psi_{P_0,Q_0}(x_i)$.
   (b) Compute $I_{G_i} = \mathsf{ProjectivePointToIdeal}(I_{P_0}, \iota, x_i)$.
   (c) Select a representative $(x_{i,1} : x_{i,2})$ of $x_i$ and compute
$$\begin{bmatrix} w_i \\ z_i \end{bmatrix} = B^{-1} \begin{bmatrix} x_{i,1} \\ x_{i,2} \end{bmatrix}$$
3. Compute $\omega_1, \omega_2, \ldots, \omega_n = \mathsf{SmoothGen}_{\ell^\bullet}(D, I_{G_1}, I_{G_2})$.
4. Use $\mathsf{IdealToIsogenies}_{\ell^\bullet}$ to compute $\theta_1, \ldots, \theta_n$ as representatives of $\omega_1, \ldots, \omega_n$ in $\mathrm{End}(E)$.
5. Compute $y_i = (w_i : z_i)$ for $i \in \{1, 2\}$. Set $\pi = \theta_1, \ldots, \theta_n, y_1, y_2$.
6. output $H(y_1, y_2), \pi$.

**VRF verification** $\mathsf{Verif}(pk, \pi, (x_1, x_2), y)$:

1. Compute $G_1 = \Psi_{P_0,Q_0}(x_1)$, $G_2 = \Psi_{P_0,Q_0}(x_2)$.
2. Compute $I_{G_i} = \mathsf{ProjectivePointToIdeal}(N, I_{P_0}, \iota, x_i)$ for $i = 1, 2$.
3. Compute $\omega_1, \omega_2, \ldots, \omega_n = \mathsf{SmoothGen}(D, I_{G_1}, I_{G_2})$.
4. Parse $\pi$ as the representation of $n$ isogenies $\theta_1, \ldots, \theta_n$ of respective degrees $n(\omega_1), \ldots, n(\omega_n)$ and two elements $y_1, y_2$ of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$.
5. Verify that $y = H(y_1, y_2)$. If not, output 0.
6. Verify that each $\theta_j$ is an endomorphism of $E$. If not, output 0.
7. Perform $\mathsf{CheckTrace}_M(E, \theta_1, \ldots, \theta_n, \omega_1, \ldots, \omega_n)$ and obtain a bit $b$ as output. If $b = 0$, output 0.

8. Compute a representative $(w_i : z_i)$ for $y_i$ for $i = 1, 2$ and use it to compute $H_i = \Psi_{P_E,Q_E}(y_i)$ for each $i = 1, 2$ and verify that $\theta_j(H_i) = H_i$ for all $j \in [1, n]$ and $i \in [1, 2]$. If not, abort and output 0.
9. Output 1.

About the algorithms left as black boxes in the above description:

- $\mathsf{IdealToIsogeny}_{\ell^\bullet}$ is introduced in [18].
- $\mathsf{ProjectivePointToIdeal}$ takes an ideal $I_{P_0}$ of norm $N$ corresponding to the cyclic subgroup of order $N$ generated by a point $P_0$, an endomorphism $\iota$ of norm coprime with $N$ and a projective point $x = (w : z) \in \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ to output the kernel ideal generated by $[w]P_0 + [z]\iota(P_0)$. This sub-algorithm is detailed in Section 6.1.
- $\mathsf{SmoothGen}$ takes two ideals $I_1, I_2$ of norm $N$ and a prime $D$, and outputs a *generating family* (see Definition 4) $\omega_1, \ldots, \omega_n$ of $\mathfrak{O} = \mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$. We postpone the description of this algorithm to Section 6.3.
- $\mathsf{CheckTrace}_M$ is parametrized by a value $M$. In Section 6.4, we introduce precisely this algorithm. In Section 7.1 we compute bounds on the size of $M$ for which our VRF reaches different levels of uniqueness.

*Remark 2.* It is possible to validate the public keys. There are two things to verify: $E$ is supersingular, $P_E, Q_E$ is a basis of $E[N]$. Checking that a curve is supersingular can be done by computing its number of points. Then, we can check that $P_E, Q_E \in E[N]$ by computing the scalar multiplication by $N$. Finally, to make sure that $P_E, Q_E$ is a basis, it suffices to compute the Weil pairing and verify that the result is not trivial. All these operations are standard in ECC and can be done efficiently.

## 5 Security Analysis

In this section, we treat the pseudo-randomness and uniqueness security properties. Provability and unbiasability are easier to obtain and are treated in Appendix B.

### 5.1 Pseudo-randomness

We state here more formally the one-time pseudo-randomness problem.

**Definition 1.** *Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ be a polynomial time adversary playing the following experiment:*

1. $pp \leftarrow \mathsf{ParamGen}(1^\lambda)$
2. $(pk, sk) \leftarrow \mathsf{KeyGen}(pp)$.
3. $(x, st_1) \leftarrow \mathcal{A}_1(pk)$.
4. $(v, \pi) = \mathsf{VRFEval}(sk, x)$.
5. $(x_0, st_2) \leftarrow \mathcal{A}_2(v, \pi, st_1)$.
6. $(v_0, \pi_0) \leftarrow \mathsf{VRFEval}(sk, x_0)$.

7. $v_1 \xleftarrow{\$} \{0,1\}^{n_2(\lambda)}$.
8. $b \xleftarrow{\$} \{0,1\}$.
9. $b' \leftarrow \mathcal{A}_3(v_b, st_2)$.

*The VRF is one-time pseudo-random if*

$$\Pr(b = b' \wedge x_0 \neq x) \leq 1/2 + \text{negl}(\lambda)$$

The pseudo-randomness property of our VRF is based on the hardness of Problem 1 that we introduce below. Before getting into the concrete formulation of the problem, we start with a small result that will prove useful for the proof of Proposition 1. It motivates the fact that our VRF is only one-time.

**Lemma 3.** *Let $E, E_0$ two supersingular elliptic curves such that there exists $\varphi : E_0 \to E$ of degree $D$ coprime with $N$, another prime. Let $G_1, G_2, G_3$ be three different cyclic subgroups in $E_0[N]$. Given $\mathcal{D}$, an algorithm to solve discrete logarithms in $E[N]$, and $H_1, H_2, H_3 \subset E[N]$ such that $\varphi(G_i) = H_i$ for $i = 1, 2, 3$, there exists a polynomial-time algorithm to compute $\varphi(G)$ for any $G \subset E_0[N]$.*

*Proof.* Let $P_i, Q_i$ be the respective generators of $G_i, H_i$ for $i = 1, 2, 3$. We know there exists $\lambda_i$ such that $\varphi(P_i) = [\lambda_i]Q_i$. The two points $Q_1, Q_2$ form a basis of $E[N]$. With $\mathcal{D}$ we can find $\mu_1, \mu_2$ such that $Q_3 = [\mu_1]Q_1 + [\mu_2]Q_2$. Doing the same on $E_0$, we obtain $P_3 = [\nu_1]P_1 + [\nu_2]P_2$. Then, we get that $\lambda_i/\lambda_3 = \mu_i/\nu_i$ for $i = 1, 2$ ($\nu_i \neq 0$ since $G_3 \neq G_1, G_3 \neq G_2$). Thus, we know the values $\lambda_1, \lambda_2$ up to a scalar, which is enough the compute the image of subgroups of order $N$. Given $R = [\eta_1]P_1 + [\eta_2]P_2$, we can easily verify that $\varphi(\langle R \rangle) = \langle [\lambda_1/\lambda_3]Q_1 + [\lambda_2/\lambda_3]Q_2 \rangle$. Apart from the computation of the coefficients $\mu_i, \nu_i, \eta_i$ for all $i$, all the operations can be made in polynomial time in $\log(D)$, $\log(N)$.

A crucial point for the hardness of Problem 1 below is that it does not seem possible to replicate the proof of Lemma 3 when the image of only two subgroups through $\varphi$ is revealed.

*Problem 1.* Let $E$ be a supersingular elliptic curve such that there exists $\varphi : E_0 \to E$ of degree $D$ and $P_E, Q_E$ a random basis of $E[N]$. The problem is for an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ to win the following game :

1. $pp = (*, P_0, Q_0) \leftarrow \mathsf{ParamGen}(1^\lambda)$
2. $((E, P_E, Q_E), sk) \leftarrow \mathsf{KeyGen}(pp)$.
3. $(x, st_1) \leftarrow \mathcal{A}_1(pk)$.
4. $(v, \pi) = \mathsf{VRFEval}(sk, x)$.
5. $(x_0, st_2) \leftarrow \mathcal{A}_2(v, \pi, st_1)$.
6. $(y_0) \leftarrow \Psi^{-1}_{P_E, Q_E}(\varphi(\Psi_{P_0, Q_0}(x_0)))$.
7. $y_1 \xleftarrow{\$} \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$.
8. $b \xleftarrow{\$} \{0,1\}$.
9. $b' \leftarrow \mathcal{A}_3(y_b, st_2)$.

The problem is to obtain $b' = b$ with non-negligible advantage when $x = (x_1, x_2)$ and $x_0 \neq x_1, x_0 \neq x_2$.

The hardness of Problem 1 underlies the pseudo-randomness of our VRF when the DLP can be solved in $E[N]$. Since we place ourselves in a quantum setting, we can assume the existence of such an algorithm.

**Proposition 1.** *Assuming the existence of an algorithm $\mathcal{D}$ to efficiently solve the DLP in $E_0[N]$ and $E[N]$, our VRF construction is pseudo-random under the hardness of Problem 1 in the random oracle model.*

*Proof.* Given the existence of $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ we are going to describe an adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3)$ breaking Problem 1. $\mathcal{B}_1$ mimicks the behavior of $\mathcal{A}_1$. Upon, receiving $v, \pi, st_1$, $\mathcal{B}_2$ computes $x_0, x_0', st_2 = \mathcal{A}_2(v, \pi, st_1)$ and outputs $x_0, st_2$. Upon receiving $y_b$, $\mathcal{B}_3$ uses $\mathcal{D}$ and $y_b, z_1, z_2, x_0, x_1, x_2$ (where $v = z_1, z_2$, and $x = x_1, x_2$) to compute $y_b'$ from $x_0'$ (as explained in the proof of Lemma 3). Then, $\mathcal{B}_3$ sets $v_b = H(y_b, y_b')$ and outputs $\mathcal{A}_3(v_b, st_2)$. By design, when $\varphi(\Psi_{P_0,Q_0}(x_i)) = \Psi_{P_E,Q_E}(y_i)$ for all $i \in \{0, 1, 2\}$, we have $\varphi(\Psi_{P_0,Q_0}(x_0')) = \Psi_{P_E,Q_E}(y_b')$ which proves that $v_b$ is a valid VRF output for $(x_0, x_0')$ when $b = 0$. Whereas, when $y_b$ is a random element in $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$, the value $H(y_b, y_b')$ is distributed as a random element in $\{0,1\}^{n_2(\lambda)}$ in the random oracle model.

### Anaysis of Problem 1

*About key recovery:* It is a well-established fact that revealing non-trivial (i.e. non-scalars) endomorphisms of an elliptic curve is basically equivalent to revealing its endomorphism ring. Once the knowledge of the endomorphism ring is leaked, an adversary is able to perform very powerful algorithms over the quaternions which usually allows one to break the standard isogeny problems. This kind of result has been the focus of an extensive line of work in isogeny-based cryptography [31,20,3,21]. In particular, the general method to compute the endomorphism ring of a given curve is to gather several endomorphisms until they generate an order that is either the desired maximal order, or an order that is contained in only a few maximal orders (thus making it possible to enumerate the solutions). However, this method cannot be applied in our case. Indeed, we reveal endomorphisms that are contained in the suborder $\mathbb{Z} + D\mathcal{O}_0$ which is expressly contained in an exponential number of maximal orders. Moreover, the adversary already has the knowledge that $\mathbb{Z} + D\mathcal{O}_0$ is a suborder of $\mathrm{End}(E)$: the embedding is a consequence of the existence of $\varphi : E_0 \to E$ of degree $D$, a fact well-known to an adversary trying to break Problem 1.

However, we do more than just revealing the existence of this suborder. Indeed, we reveal a concrete embedding of a suborder of $\mathbb{Z} + D\mathcal{O}_0$ in $\mathrm{End}(E)$. This is done with the endomorphisms $\theta_1, \ldots, \theta_n$. Revealing this embedding may seem a troublesome thing to do at first glance. Indeed, the torsion points attacks [36,33] on SIDH are obtained precisely by computing one of the endomorphisms of $\mathbb{Z} + D\mathcal{O}_0$. In these attacks, the knowledge of such endomorphisms is enough

to compute the secret isogeny $\varphi$. However, an important part of making this attack work in polynomial time is that $D$ is smooth. In our case, $D$ is a prime number which makes the computation of isogenies of degree $D$ hard without the knowledge of the endomorphism ring. Thus, even if the knowledge $\theta_i$ is enough to uniquely define $\varphi$, the best algorithms to perform this computation have exponential complexity.

*Concrete Pseudo-Randomness Problem:* The above reasoning justifies why we believe that recovering the secret isogeny $\varphi$ is hard. But this is far from enough to argue that our VRF is pseudo-random. Of course, recovering the secret key is one way to break pseudo-randomness but it is definitely not the only one. In particular, reformulating the problem through the bijections $\Psi_{P_0,Q_0}$ and $\Psi_{P_E,Q_E}$, it is clear that the task at hand is to distinguish between the random subgroup $\Psi_{Q_E,P_E}(y_0)$ and the image through $\varphi$ of the subgroup $G_0 = \Psi_{P_0,Q_0}(x_0)$. This computation must remain difficult, even after revealing the image through $\phi$ of two subgroups $G_1$, $G_2$ (computed as $G_i = \Psi_{P_0,Q_0}(x_i)$ for $x = (x_1, x_2)$). A first easy remark is that when $x_0 \neq x_1, x_2$ we have $G_0 \neq G_1, G_2$ which means that we cannot extract the answer directly. However, a generator of $G_0$ can always be expressed as a linear combination of generators of $G_1$ and $G_2$. The question is: can the adversary exploit this decomposition to break the problem ?

An important ingredient seems to be the ability to break the DLP in $E[N]$. As we are claiming post-quantum security, we can just assume that the adversary has access to a DLP oracle $\mathcal{D}$. In particular, given a basis of $E[N]$ and a third point $R$, the adversary can find the coordinates of $R$ with respect to this basis. Thus, taking generators $P_i$ of $G_i$ for $i = 0, 1, 2$, an adversary can express $P_0$ as a linear combination of $P_1$ and $P_2$. Writing $H_i = \Psi_{P_E,Q_E}(z_i)$ for $i = 1, 2$ when $v = z_1, z_2$ and $H_0 = \Psi_{P_E,Q_E}(y_b)$, we can also find generators $Q_0, Q_1, Q_2$ of $H_0, H_1, H_2$ and write similar decompositions. Let us introduce more formal notation. Let $\mu_i$ be the scalars such that $[\mu_i]Q_i = \varphi(P_i)$ for $i = 1, 2$. Since the $P_i$ and $Q_i$ are computed independently as generators of the two subgroups, it is clear that these scalars are random. Recovering these scalars trivially allows the adversary to solve the problem since the decomposition $P_0 = [a_1]P_1 + [a_2]P_2$ gives $\langle\varphi(Q_0)\rangle = \langle[a_1\mu_1]Q_1 + [a_2\mu_2]Q_2\rangle$. However, it does not seem easy to recover these scalars. The Weil pairing allows the adversary to extract the value of $\mu_1\mu_2$, but no more than that. This information does not seem to be enough to solve the distinguishing problem. Indeed, for any cyclic subgroup $H \subset E[N]$, it is easy to verify that we can find values $\mu_1(H), \mu_2(H)$ satisfying the multiplicative condition and such that $H = \varphi(G_0)$ if $[\mu_i(H)]Q_i = \varphi(P_i)$ for $i = 1, 2$.

Without further information, it seems hard to solve the distinguishing problem with good probability. However, the story is not over in our case. We have a lot of additional information, as the prover reveals the endomorphisms $\theta_1, \ldots, \theta_n$. We explained that the degree $D$ being a large prime prevents the adversary from exploiting the knowledge of $\theta_1, \cdots, \theta_n$ to compute directly $\hat{\varphi}$. However, the $\theta_i$ are related to $\varphi$ and it is possible to evaluate them. We want to verify that this does not help to solve Problem 1. Indeed, for each $j \in [1, n]$, there exist $d_j \in \mathbb{Z}$ and $\alpha_j \in (\mathbb{Z} + I_{G_1}) \cap \mathbb{Z} + (I_{G_2})$ such that $\theta_j = [d_j] + \varphi \circ \alpha_j \circ \hat{\varphi}$. Thus, the evaluation of $\theta_j$

on points of $E[N]$ is related to the evaluation of $\hat{\varphi}$ on $E[N]$. However, we are going to show that the evaluation of each $\theta_j$ on any point of $E[N]$ is independent of the values $\mu_1, \mu_2$ that we introduced above. This proves that evaluating the $\theta_j$ on $E[N]$ cannot help to recover $\mu_1, \mu_2$. The main fact (which stems from Lemmas 2 and 10) is that if $\theta_j \in \mathbb{Z}+D((\mathbb{Z}+I_{G_1})\cap(\mathbb{Z}+I_{G_2}))\setminus\mathbb{Z}+DN\mathcal{O}_0$, then each $\alpha_i$ has two eigenspaces in $E_0[N]$, which are exactly $G_1, G_2$. Then, there exist eigenvalues $\lambda_{i,j}$ such that $\alpha_j(P_i) = [\lambda_{i,j}]P_i$ and $\theta_j(Q_i) = [d_j + D\lambda_{i,j}]Q_i$ for $i = 1, 2$ and $j \in [n]$. If we take any point $R \in E[N]$, then we can express it as $[b_1]Q_1 + [b_2]Q_2$. A simple computation shows that $\theta_j(R) = [d_j]R + [D]([b_1\lambda_{1,j}]Q_1 + [b_2\lambda_{2,j}]Q_2)$. As we announced, this expression is completely independent of $\mu_1, \mu_2$.

*Remark 3.* Finally, we highlight that it seems important to take $N$ prime to ensure the hardness of Problem 1. We are going to explain a way to break the problem when $N = N_1N_2$ with $N_1$ coprime with $N_2$ (a similar method can be applied when $N$ is a prime power). If we take $P_2$ a point of order $N_2$ and $P_1, Q_1$ a basis of $E_0[N_1]$, then the points $R = P_1 + P_2$ and $S = P_2 + Q_1$ do not generate the same subgroups of order $N$, but they satisfy $\langle[N_1]R\rangle = \langle[N_1]S\rangle$. Pushing these subgroups under a $D$-isogeny $\varphi$ will conserve this property. This gives us a way to construct distinct subgroups whose image under $\varphi$ satisfies a specific property. Of course, two random subgroups will have a very low probability of satisfying this same property, so a distinguisher can be easily obtained from this idea. Fixing this problem would imply imposing some limitations on the choices of VRF input, and it is not clear how one would do that in a clean way. In any case, it appears both simpler and more secure to take a prime $N$.

### 5.2 Uniqueness

We have two flavours of uniqueness: unconditional and computational. The first one is harder to reach, and that is why we are going to present two versions of our VRF, providing a tradeoff between security and efficiency of the verification.

We start by introducing the relevant security definitions.

**Definition 2.** *A VRF is said to satisfy unconditional full uniqueness when no values $pk, v, v', x, \pi, \pi'$ can satisfy $\mathsf{Verif}(pk, \pi, x, v) = 1$ and $\mathsf{Verif}(pk, \pi', x, v') = 1$ with $v \neq v'$.*

**Definition 3.** *Let $pp \leftarrow \mathsf{ParamGen}(1^\lambda)$. A VRF is said to satisfy computational full uniqueness if for every polynomial-time adversary $\mathcal{A}$, $(x, pk, v, v', \pi, \pi') \leftarrow \mathcal{A}(pp)$ we have:*

$$Pr(\mathsf{Verif}(pk, \pi, x, v) = \mathsf{Verif}(pk, \pi', x, v') = 1 \wedge v \neq v') \leq \mathrm{negl}(\lambda).$$

The key part behind uniqueness is the following result, whose proof we postpone until Appendix A.

**Proposition 2.** *Take $I_{G_1}$ and $I_{G_2}$ two $\mathcal{O}_0$-ideals of prime norm $N$, corresponding to the kernel ideals of two subgroups $G_1, G_2$ of order $N$ in $E_0[N]$. Let $D$ be a*

*prime number different from $N$. Given a supersingular curve $E$ not isomorphic to $E_0$, if there exists an embedding of $\mathfrak{O} = \mathbb{Z} + D((\mathbb{Z} + I_{G_1}) \cap (\mathbb{Z} + I_{G_2}))$ in* $\mathrm{End}(E)$ *and there exist two different subgroups $H_1, H_2 \subset E[N]$ stable under any endomorphism of $\mathfrak{O}$, there exists an isogeny $\varphi$ of degree $D$ between $E_0$ and $E$, and $H_i = \varphi(G_i)$ for $i = 1, 2$.*

Proposition 2 suggests that the verifier must be able to check that the endomorphisms given as proof are elements of $\mathfrak{O}$. As showed in Lemma 4, it suffices to check some traces and norms for that. Norms are easy but traces are harder. To do that we rely on the $\mathsf{CheckTrace}_M$ protocol. This algorithm verifies correctness of the traces $\mod M$ by evaluating $\mathrm{tr}(\theta) = \theta + \hat{\theta}$ on $E[M]$. We will see in Lemma 5 that if we take $M$ big enough, then we obtain unconditional uniqueness.

**Unconditional Uniqueness** We start with the definition of a generating family.

**Definition 4.** *A generating family $\theta_1, \cdots, \theta_n$ for an order $\mathfrak{O}$ is a set of elements in $\mathfrak{O}$ such that any element $\rho \in \mathfrak{O}$ can be written as a linear combination of 1 and products $\prod_{j \in \mathcal{I}} \theta_j$ for all $\mathcal{I} \subset [1, n]$. In that case, we write $\mathfrak{O} = \mathsf{Order}(\theta_1, \ldots, \theta_n)$.*

The following lemma shows that unconditional uniqueness can be obtained by checking the norm and traces of at most $2^n$ endomorphisms when given a generating family of size $n$.

**Lemma 4.** *Two orders $\mathcal{O}_1 = \mathsf{Order}(\theta_1, \ldots, \theta_n)$ and $\mathcal{O}_2 = \mathsf{Order}(\omega_1, \ldots, \omega_n)$ of rank 4 in a quaternion algebra are isomorphic if $\mathrm{tr}(\prod_{j \in \mathcal{I}} \theta_j) = \mathrm{tr}(\prod_{j \in \mathcal{I}} \omega_j)$ for all $\mathcal{I} \subset [1, n]$.*

*Proof.* An isomorphism of quaternion orders is a bijection $\alpha : \mathcal{O}_1 \to \mathcal{O}_2$ such that for all $\theta \in \mathcal{O}_1$, $n(\alpha(\theta)) = n(\theta)$ and $\mathrm{tr}(\alpha(\theta)) = \mathrm{tr}(\theta)$. We label $\theta'_1, \ldots, \theta'_m$ (resp. $\omega'_1, \ldots, \omega'_m$) with $m = 2^n$ the set of multi-products obtained from $\theta_1, \ldots, \theta_n$ (resp. $\omega_1, \ldots, \omega_n$). By the definition of a generating family, any element $\alpha \in \mathcal{O}_1$ (resp. $\mathcal{O}_2$) can be written as a linear combination of $\theta'_1, \ldots, \theta'_m$ (resp. $\omega'_1, \ldots, \omega'_m$). We claim that the map $\alpha : \sum_{i=1}^m x_i \theta'_i \mapsto \sum_{i=1}^m x_i \omega'_i$ is an isomorphism of quaternion orders. It is easy to verify that this map is bijective and linear. It remains to check that it preserves the trace and the norm.

The trace being linear, its clear that $\mathrm{tr}(\alpha(\theta)) = \mathrm{tr}(\theta)$ for all $\theta \in \mathcal{O}_1$. For any $\theta = \sum_{i=1}^m x_i \theta'_i$, we have $n(\theta) = \sum_{1 \le i < j \le m} x_i x_j \mathrm{tr}(\theta'_i \hat{\theta}'_j) + \frac{1}{2} \sum_{i=1}^m x_i^2 \mathrm{tr}(\theta'_i \hat{\theta}'_i)$. Thus, we need to prove that we have equality of traces for all $\theta'_i \hat{\theta}'_j$ and $\omega'_i \hat{\omega}'_j$. Since $\mathrm{tr}(ab) = \mathrm{tr}(ba) = \mathrm{tr}(\hat{a}\hat{b})$ and $2\mathrm{tr}(a)\mathrm{tr}(b) = \mathrm{tr}(ab) + \mathrm{tr}(\hat{a}b)$ for all $a, b \in B_{p,\infty}$, it suffices to verify the equality $\mathrm{tr}(\prod_{j \in \mathcal{I}} \theta_j) = \mathrm{tr}(\prod_{j \in \mathcal{I}} \omega_j)$ to get the desired result. This also proves that we have equality of norms between $\theta$ and $\alpha(\theta)$.

**Lemma 5.** *Given any $\theta \in \mathrm{End}(E)$, if $\mathrm{tr}(\theta) = t \mod M$ for $M > 4\sqrt{n(\theta)}$ and $|t| \le M/2$, then $\mathrm{tr}(\theta) = t$.*

*Proof.* Over $B_{p,\infty}$, the norm form is $n : (x, y, z, w) \mapsto x^2+qy^2+pz^2+qpw^2$ where $q > 0, p > 0$. Since $\mathrm{tr} : (x, y, z, w) \mapsto 2x$, we can easily verify that $\mathrm{tr}(\theta)^2 < 4n(\theta)$. This gives a bound of $2\sqrt{n(\theta)}$ on the absolute value of $\mathrm{tr}(\theta)$. The result follows.

Combining our three lemmas, we deduce that if $M$ is bigger than $2\sqrt{n(\theta_j)^n}$ for all $j \in [n]$, then we obtain unconditional uniqueness by checking $O(2^n)$ traces mod $M$.

**Theorem 1.** *Assuming that* SmoothGen *outputs a generating family whose elements have norm smaller than* $M^{2/n}/4$ *and the parameters* $E_0, p, D$ *are such that there does not exist a curve* $E$ *with two distinct* $D$*-isogenies between* $E_0$ *and* $E$*, the VRF scheme introduced in Section 4.2 satisfies unconditional uniqueness.*

*Proof.* By Lemma 4 and Lemma 5, the verification performed by applying $\mathsf{CheckTrace}_M$ implies that $\mathbb{Z} + D((\mathbb{Z} + I_{G_1}) \cap (\mathbb{Z} + I_{G_2}))$ is embedded in $\mathrm{End}(E)$. Then, the next step in Verif guarantees that $H_1$ and $H_2$ are eigenspaces for each $\theta_j$. This proves that these two subgroups are the respective images of $G_1, G_2$ through an isogeny $\varphi$ of degree $D$ by Proposition 2. Since by hypothesis there exists at most one isogeny of degree $D$ between $E_0$ and $E$, there can be only one correct pair of subgroups $H_1, H_2$ passing the verification, and one correct output $v = H(\Psi^{-1}_{P_E,Q_E}(H_1), \Psi^{-1}_{P_E,Q_E}(H_2))$.

**Computational Uniqueness** In Section 6.3 we show that we can take $n = 3$ and prove an heuristic upper bound on the norm of the elements given in output of SmoothGen. The minimal value of $M$ to reach the inequality given in Theorem 1 is quite high, and checking the traces modulo this $M$ will not prove very efficient. An idea is to check equality of traces modulo a smaller integer to gain efficiency. In this case, we cannot show unconditional uniqueness, and that is why we introduce a new problem upon which we base the computational uniqueness of our construction.

*Problem 2.* Let $p, N, D, E_0, P_0, Q_0, M$ be the parameters of the VRF. The problem is to find $E, P_E, Q_E$ and $G_1, G_2, H_1, H_2$ and $\theta_1, \ldots, \theta_n \in \mathrm{End}(E)$ such that if $\omega_1, \ldots, \omega_n = \mathsf{SmoothGen}(\mathbb{Z}+D((\mathbb{Z}+I_{G_1})\cap(\mathbb{Z}+I_{G_2})))$, then $\theta_j$ is an endomorphism of $E$ with $n(\theta_j) = n(\omega_j)$ and $\mathrm{tr}(\prod_{j\in\mathcal{I}} \theta_j) = \mathrm{tr}(\prod_{j\in\mathcal{I}} \omega_j)$ for all $\mathcal{I} \subset [1, n]$ and $H_1, H_2$ are common eigenspaces of $\theta_j$ for all $j \in [1, n]$ and there does not exists $\varphi$ of degree $D$ such that $H_i = \varphi(G_i)$ for $i = 1, 2$.

**Theorem 2.** *Under the hardness of Problem 2 for the parameter* $M$*, the VRF scheme introduced in Section 4.2 satisfies computational uniqueness if the parameters* $E_0, p, D$ *are such that there does not exists a curve* $E$ *with two distinct* $D$*-isogenies between* $E_0$ *and* $E$*.*

*Proof.* To prove Theorem 2, we are going to show how to solve Problem 2 using an algorithm $\mathcal{A}$ that can break uniqueness. It is easy to see that the two problems have the same input: the parameters of the VRF. Thus, we can feed an instance of Problem 2 to $\mathcal{A}$ and obtain an output. We parse this output as

$x, E, P_E, Q_E, v, v', \pi, \pi'$. We parse $x$ as $x_1, x_2$ and $\pi$ (resp. $\pi'$) as $\theta_1, \ldots, \theta_n, y_1, y_2$ (resp. $\theta'_1, \ldots, \theta'_n, y'_1, y'_2$). Since $v \neq v'$, and there are no two distinct isogenies of degree $D$ between $E_0$ and $E$, we can assume wlog that $y_1$ is such that there are no isogenies of degree $D$ sending $\Psi_{P_0,Q_0}(x_1)$ to $\Psi_{P_E,Q_E}(y_1)$. By construction, if $\mathsf{Verif}(pk, \pi, x, v) = 1$, then the tuple $E, P_E, Q_E), \Psi_{P_0,Q_0}(x_1), \Psi_{P_0,Q_0}(x_2), \Psi_{P_E,Q_E}(y_1), \Psi_{P_E,Q_E}(y_2)$, $\theta_1, \ldots, \theta_n$ is a correct output to Problem 2. In practice, we don't know which one among $(y_1, y_2)$ and $(y'_1, y'_2)$ will provide a correct output. Thus, by selecting a random one among the two, we obtain an algorithm with a success probability of at least $1/2$.

In practice, we propose to base the computational uniqueness of our VRF under the hardness of Problem 2 when $M = (p^2 - 1)/2$. This value appears to be a good compromise between efficiency and security.

**Analysis of Problem 2** First, we would like to highlight that the hardness of Problem 2 is a type of assumption quite unusual in isogeny-based cryptography. Contrary to Problem 1 (which is new but remains related to computation and evaluation of isogenies, two very classical problems), the hardness of Problem 2 is related to the resolution of some set of quadratic equations.

Problem 2 is difficult to analyze. Indeed, in Theorem 1 we give an upper bound on the value of $M$ for which there are no solutions to the problem. However, it is not clear what is the optimal such value. It may be that when $M = (p^2 - 1)/2$, as we intend to take, the problem is already unsolvable. However, since we were unable to prove that, the conservative approach is to assume that there may be some solutions. In that case, finding a solution amounts to finding an input $x = (x_1, x_2)$ and an order in $\mathrm{End}(E)$ satisfying some constraint on norm and traces depending on $x$, but that is not constructed as $\mathbb{Z} + D((\mathbb{Z} + I_{G_1}) \cap (\mathbb{Z} + I_{G_2}))$ where $G_i = \Psi_{P_0,Q_0}(x_i)$. The trace and norm equations can be seen as quadratic equations that can be solved mod $M$, but since we also need equality of the norms over $\mathbb{Z}$, it is not clear whether there are solutions and if they are easy to find. The usual tools used to solve equations over quaternion orders (for instance in [32,18]) are not sufficient to address our problem.

Let us look at the simple example where $n = 2$. Then, the order is $\mathfrak{O} = \mathsf{Order}(\theta_1, \theta_2) = \langle 1, \theta_1, \theta_2, \theta_1\theta_2 \rangle$. The goal is to find $\theta_1, \theta_2$ with a precise constraint on their norm, and a constraint mod $M$ for the three traces $\mathrm{tr}(\theta_1), \mathrm{tr}(\theta_2), \mathrm{tr}(\theta_1\theta_2)$. While it is easy to find $\theta_1$ and $\theta_2$ with the correct norm and trace, it seems difficult to ensure the additional constraint on $\mathrm{tr}(\theta_1\theta_2)$. Let us look at that constraint when $\theta_1 = a + ib + jc + kd$ and $\theta_2 = e + if + jg + kh$, then $\mathrm{tr}(\theta_1\theta_2) = ae - (qbf + p(cg + qdh))$. Thus, the problem is: given $n_1, n_2, t_1, t_2, t_3, M$ find $a, b, c, d, e, f, g, h$ such that $a^2 + qb^2 + pc^2 + qpd^2 = n_1, e^2 + qf^2 + pg^2 + pqh^2 = n_2$ and $2a = t_1 \mod M, 2e = t_2 \mod M$ and $ae - (qbf + p(cg + qdh)) = t_3 \mod M$. This appears to be hard when $M$ is big enough for the constraints mod $M$ to have a good probability to be respected by luck. In practice, as explained in Section 6.3, we take $n = 3$ and $\mathfrak{O}$ has an even more complicated structure which only increases the number of equations to be verified, as highlighted in Lemma 4.

*Remark 4.* Additionally, we highlight that progress toward solving the kind of equations above, would probably allow us to devise an algorithm SmoothGen finding solutions of smaller norm, which would make Problem 2 more difficult.

## 6 Sub-algorithms over the quaternion algebra

In this section, we fill the blanks left in Section 4.2, and dive into the more complicated sub-algorithms of our VRF construction. We provide precise descriptions of the algorithms ProjectivePointToIdeal, SmoothGen, and $\mathsf{CheckTrace}_M$ in Sections 6.1, 6.3 and 6.4 respectively.

Following the classical approach in the literature [32,18], we take $B_{p,\infty}$ to be the quaternion order generated by $1, i, j, k$ where $i^2 = -q$, $j^2 = -p$ and $k = ij = -ji$ for some small integer $q$ (when $p = 3 \mod 4$ we can take $q = 1$). Then, we assume that $\mathcal{O}_0 \subset B_{p,\infty}$ is a special extremal order containing a suborder with orthogonal basis $\langle 1, \omega, j, \omega j \rangle$ where $\mathbb{Z}[\omega] \subset \mathbb{Q}[i]$ is a quadratic order of small discriminant.

### 6.1 Kernel ideal computation from projective point

Here, we describe the algorithm ProjectivePointToIdeal that is used in both VRFEval and Verif to compute the two kernel ideals $I_{G_1}, I_{G_2}$.

This procedure is not entirely trivial from existing techniques. Even though computing kernel ideals is now standard in isogeny-based cryptography [24,18,20], it is not an efficient operation in the generic case due to the necessity to compute some discrete logarithms. This is prohibitive in our case where the order is the large prime number $N$. To overcome this obstacle, we use the ideal $I_{P_0}$ and the endomorphism $\iota$ given as public parameters of the scheme. We define $\mathsf{ProjectivePointToIdeal}(N, I_{P_0}, \iota, (w : z)) = [\mathcal{O}_0(w + z\iota)]_* I_{P_0}$ using the ideal push-forward notation $[K]_* J$ introduced in [18].

When $P = [w]P_0 + [z]Q_0$ and $Q_0 = \iota(P_0)$, we obtain $P = (w + z\iota)(P_0)$. Then, by definition of the push-forward ideal, we have $I_P = [\mathcal{O}_0(w + z\iota)]_* I_{P_0}$ and it can be easily computed from $(w : z), \iota, I_{P_0}$ using the formulas described in [18].

Given the explanations above, one might wonder how to efficiently generate the ideal $I_{P_0}$. The answer is that there are some choices of $P_0$ where the ideal can be computed easily. This is the case when $P_0$ is an eigenvector of the Frobenius morphism. As we explain in Section 7.1, our choice of $E_0$ allows us to select exactly such a $P_0$.

### 6.2 Algorithms from previous works

In Section 6.3 below, we introduce the algorithm SmoothGen that produces a generating family of smooth norm in some special class of quaternion orders. The resolution of norm equations in quaternion ideals and orders has been the focus of [32,18,37] (respectively targeting ideals, Eichler orders and orders of the

form $\mathbb{Z} + D\mathcal{O}_0$). The purpose of Algorithm 2 below is to extend to the case $\mathbb{Z} + D\mathfrak{O}_0$ where $\mathfrak{O}_0$ can cover a large class of orders (see Remark 6).

In Section 6.3, we rely upon several algorithms existing in the literature. The full version of [18] is a good reference for all these algorithms. We briefly recall their purpose.

- $\mathsf{EquivalentPrimeIdeal}(I)$, given a left $\mathcal{O}_0$-ideal $I$, finds an equivalent left $\mathcal{O}_0$-ideal of prime norm.
- $\mathsf{EichlerModConstraint}(I, \gamma)$, given an ideal $I$ of norm $N$, and $\gamma \in \mathcal{O}_0$ of norm $n$ coprime with $N$, finds $(C_0 : D_0) \in \mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$ such that $\mu_0 = j(C_0 + \omega D_0)$ satisfies $\gamma\mu_0 \in \mathbb{Z} + I$.
- $\mathsf{StrongApproximation}_F(N, C_0, D_0)$, given a prime $N$ and $C_0, D_0 \in \mathbb{Z}$, finds $\mu = \lambda\mu_0 + N\mu_1 \in \mathcal{O}_0$ of norm dividing $F$, with $\mu_0 = j(C_0 + \omega D_0)$. We write $\mathsf{StrongApproximation}_{\ell^\bullet}$ when the expected norm is a power of $\ell$.

*Remark 5.* The $\mathsf{StrongApproximation}_{\ell^\bullet}$ algorithm was originally introduced for a prime number $N$ in [32]. When we add the additional constraint that $\ell$ is not a square mod $N$, it can be shown with heuristic arguments to succeed in polynomial time with overwhelming probability. Without this residuosity constraint the success probability is $1/2$. We can easily extend $\mathsf{StrongApproximation}$ to the case of composite $N$ (and this is the version that we use for Algorithm 1) if we allow the success probability to decrease. The case where $N$ has two large primes divisors is treated in [18], and they show that the success probability is $1/4$. In general, it is easy to see that the success probability is $1/2^k$ where $k$ is the number of distinct prime divisors of $N$. Below, we are going to use the algorithm with $N$ having three large prime divisors. When all the prime factors $N_i$ are such that $\ell$ is not a square mod $N_i$, the probability can be increased to $1/2^{k-1}$ as is done in [32].

### 6.3 Computing a smooth generating family

In this section, we describe how to perform the $\mathsf{SmoothGen}$ protocol. The overall idea is to generate several random elements and hope they form a generating family. Thus, we mainly focus on how to sample an element of smooth norm with some randomization. We discuss how many such elements we need to sample in the end of this section.

The goal is a randomized algorithm to solve norm equations in an order of the form $\mathfrak{O} = \mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$, where $I_1$ and $I_2$ are integral $\mathcal{O}_0$-ideals of norm $N$. In the next paragraph, we introduce Algorithm 1 to solve a norm equation in $\mathbb{Z} + DI$ for $I$ some integral $\mathcal{O}_0$-ideal. From there, we derive Algorithm 2 which applies Algorithm 1 to solve norm equations in $\mathfrak{O}$.

**Solving norm equations in $\mathbb{Z} + DI$.** The recent paper [18] introduced a method to solve norm equations in orders of the form $\mathbb{Z} + I$ for some ideal $I$ of norm $N$. We briefly present their approach before explaining how to modify it in order to obtain a method to solve norm equations in $\mathbb{Z} + DI$ for a large prime

$D$. Given an element $\gamma \in \mathcal{O}_0$ of norm coprime with $N$, the idea of the algorithm from [18] is that there always exists $C_0, D_0$ such that $\gamma j(C_0 + D_0\omega) \in \mathbb{Z} + I$. After that, it suffices to do a *strong approximation* ([32,24,18]) mod $N$ to find $\mu_0 \in \mathcal{O}_0$ so that $j(C_0 + D_0\omega) + N\mu_0$ has smooth norm. If $\gamma$ has also smooth norm, then it can be shown that $\gamma(j(C_0 + D_0\omega) + N\mu_0)$ is an element of $\mathbb{Z} + I$ of smooth norm.

In our case, if $\gamma$ is such that there exists a solution $C_1, D_1$ with $\gamma j(C_1 + D_1\omega) \in \mathbb{Z} + DI$, then we can follow the same method and do the strong approximation mod $ND$ to find an element $j(C_1 + D_1\omega) + ND\mu_1$ of the good norm in $\mathbb{Z} + DI$. However, unlike for $\mathbb{Z} + I$, this is not always possible when working in $\mathbb{Z} + DI$. To ensure that we are in this good situation, we need an additional condition on $\gamma$. Fortunately this condition is exactly what allow us to find $\gamma$ of smooth norm. We give a precise statement in the following proposition:

**Proposition 3.** *Let $I$ be an integral left $\mathcal{O}_0$-ideal of norm $N$ and let $D$ be a distinct prime number. If $\gamma \in \mathcal{O}_0$ can be written as $j(C_2 + \omega D_2) + D\mu_2$ with $\mu_2 \in \mathcal{O}_0$ and $\gamma$ has norm coprime with $N$, then there exists $C_1, D_1 \in \mathbb{Z}$ such that $\gamma j(C_1 + \omega D_1) \in \mathbb{Z} + DI$.*

*Proof.* If $\gamma$ has norm coprime with $N$, we know from [18] that there exists $C_0, D_0$ such that $\gamma j(C_0 + \omega D_0) \in \mathbb{Z} + I$. Then, if we set $C_2' = -D_2' C_2 (D_2)^{-1}$ mod $D$ for any $D_2'$, it is easy to verify that $\gamma j(C_2' + \omega D_2') \in \mathbb{Z} + D\mathcal{O}_0$. Hence, if $C_1, D_1$ satisfies $C_1, D_1 = C_0, D_0 \mod N$, $C_1, D_1 = C_2', D_2' \mod D$ we have that $\gamma j(C_1 + \omega D_1) \in \mathbb{Z} + DI$. By the CRT, we know we can find such $C_1, D_1$.

**Algorithm 1** $\mathsf{ExtendedEichlerNormEquation}_{\ell^\bullet}(D, I)$

**Require:** $I$ a left $\mathcal{O}_0$-ideal of norm $N$ coprime with $D$.
**Ensure:** $\beta \in \mathbb{Z} + DI$ of norm $\ell^e$.

1: Select a random class $(C_2 : D_2) \in \mathbb{P}^1(\mathbb{Z}/D\mathbb{Z})$.
2: Compute $\mu_2 = \mathsf{StrongApproximation}_{\ell^\bullet}(D, C_2, D_2))$ and set $\gamma = j(C_2 + \omega D_2) + D\mu_2$. If the computation fails, go back to Step 1.
3: Compute $(C_0 : D_0) = \mathsf{EichlerModConstraint}(\gamma, I)$.
4: Sample a random $D_2'$ in $\mathbb{Z}/N_2\mathbb{Z}$, compute $C_2' = -D_2' C_2 (D_2)^{-1} \mod D$.
5: Compute $C_1 = \mathsf{CRT}_{N,D}(C_0, C_2')$, $D_1 = \mathsf{CRT}_{N,D}(D_0, D_2')$.
6: Compute $\mu_1 = \mathsf{StrongApproximation}_{\ell^\bullet}(ND, C_1, D_1))$. If it fails, go back to step 1.
7: **return** $\beta = (j(C_2 + \omega D_2) + D\mu_2)(j(C1 + \omega D_1) + ND\mu_1)$.

**Proposition 4.** *When $N$ has a constant number $k$ of prime divisors and is coprime with $D$, Algorithm 1 terminates in probabilistic polynomial time and outputs an element of $\mathbb{Z} + DI$ of norm $\ell^e$ for some integer $e$.*

*Proof.* As mentioned in Remark 5, the algorithm $\mathsf{StrongApproximation}_{\ell^\bullet}(D, \cdot)$ finds a solution of norm $\ell^{e_2}$ with probability $1/2$ in heuristic polynomial time. This probability can even be brought to 1 when $\ell$ is not a quadratic residue mod

$D$. As a result of Proposition 3, EichlerModConstraint always succeeds in finding a solution $(C_0 : D_0)$. Then, as pointed out in Remark 5, when $k$ is constant the strong approximation mod $ND$ succeeds with constant probability. Assuming that a new choice of $(C_2 : D_2)$ randomizes $(C_1 : D_1)$ sufficiently we can show that a solution can be found with overwhelming probability after a logarithmic number of repetitions. This proves the algorithm's termination.

For correctness, we can verify easily that $j(C_2+D_2\omega)j(C_2'+\omega D_2') \in \mathbb{Z}+D\mathcal{O}_0$. Since $\beta - j(C_2 + D_2\omega)j(C_2' + \omega D_2') \in D\mathcal{O}_0$ this proves that $\beta \in \mathbb{Z} + D\mathcal{O}_0$. By the correctness of EichlerModConstraint and the fact that $N\mathcal{O}_0$ is contained in $I$ we can also show that $\beta \in \mathbb{Z} + I$. Hence, $\beta \in (\mathbb{Z} + D\mathcal{O}_0) \cap (\mathbb{Z} + I) = \mathbb{Z} + DI$.

The estimates provided in [18] allow us to predict that we can find a solution $\beta$ of norm $\ell^e$ where $e \sim 2\log_\ell(p)+6\log_\ell(D)+3\log_\ell(N)$. This comes from the fact that a strong approximation mod $N'$ can find solutions of norm approximately equal to $pN'^3$.

**Solving norm equations over $\mathfrak{O}$** Algorithm 1 is not enough to solve our problem as the order $\mathfrak{O} = \mathbb{Z}+D((\mathbb{Z}+I_1)\cap(\mathbb{Z}+I_2))$ cannot be directly expressed in the form $\mathbb{Z}+DI$ where $I$ is a left integral $\mathcal{O}_0$-ideal. Nonetheless, it is possible to sample ideals $J$ such that $\mathbb{Z} + DJ \subset \mathfrak{O}$, thus allowing us to circumvent this limitation.

Our method to find this ideal $J$ comes from the decomposition $\mathfrak{O} = \mathbb{Z} + D\overline{I_1}I_2 = \mathbb{Z}+D((\mathbb{Z}+I_1)\cap(\mathbb{Z}+I_2))$ that we already used in the proof of Lemma 3. The left order of the ideal $\overline{I_1}I_2$ is not $\mathcal{O}_0$, but if we take $J_1 \sim I_1$ we obtain an $\mathcal{O}_0$-ideal as $J = J_1\overline{I_1}I_2$. Then, we can apply the above algorithm to solve norm equations in $\mathbb{Z} + JD \subset \mathfrak{O}$. In reality, the previous ideal $J$ is not well-defined because $\mathcal{O}_R(J_1) \neq \mathcal{O}_R(I_1)$, but it conveys the idea. The precise formulation is used in Algorithm 2 and is proven in Lemma 6.

**Algorithm 2** SpecialOrderNormEquation$_{\ell^\bullet}(I_1, I_2, D)$

**Require:** $I_1, I_2$ two distinct left $\mathcal{O}_0$-ideals of norm $N$ coprime with $D$.
**Ensure:** $\beta \in \mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$ of norm $\ell^e$.
1: Compute $J_1 =$ EquivalentPrimeIdeal($\mathsf{I}_1$) and $\alpha_1$ such that $J_1 = I_1\alpha_1$.
2: Compute $\beta =$ ExtendedEichlerNormEquation$_{\ell^\bullet}(D, J)$ on $J = J_1\alpha_1^{-1}\overline{I_1}I_2\alpha_1$.
3: **return** $\alpha_1\beta\alpha_1^{-1}$.

The correctness of SpecialOrderNormEquation relies on the following result:

**Lemma 6.** *Given $I_1, I_2$ two left $\mathcal{O}_0$-ideals of same norm and $J_1 = I_1\alpha_1$ for some $\alpha_1 \in B_{p,\infty}^\times$, if $\beta \in \mathbb{Z} + DJ$ where $J$ is the integral left $\mathcal{O}_0$-ideal defined as $J_1\alpha_1^{-1}\overline{I_1}I_2\alpha_1$, then $\alpha_1\beta\alpha_1 p^{-1} \in \mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$.*

*Proof.* If we take $J_1 = I_1\alpha_1$, then $\mathcal{O}_R(J_1) = \alpha_1^{-1}\mathcal{O}_R(I_1)\alpha_1$ which is why the product $J = J_1\alpha_1^{-1}\overline{I_1}I_2\alpha_1$ is well-defined. It is clear that $\mathcal{O}_L(J) = \mathcal{O}_0$. Finally, since $J \subset \alpha_1^{-1}\overline{I_1}I_2\alpha_1$ when $\beta \in \mathbb{Z}+DJ$, the conjugate $\alpha_1\beta\alpha_1^{-1} \in \mathbb{Z}+D\overline{I_1}I_2 = \mathfrak{O}$.

**Proposition 5.** *When the norm $N$ of $I_1$ and $I_2$ is coprime to $D$, the algorithm* SpecialOrderNormEquation *terminates in heuristic polynomial time and compute $\beta \in \mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$ of norm $\ell^e$.*

*Proof.* With overwhelming probability, the norm of $J_1$ will be coprime with $D$. As shown in Lemma 6, the input $J$ satsifies the condition of Proposition 4 and so the algorithm terminates in heuristic polynomial time. Correctness follows from Lemma 6 and it is easy to verify that $\alpha_1^{-1}\beta\alpha_1$ has same norm as $\beta$, which proves that the norm of $\beta$ is a power of $\ell$.

The norm of the ideal $J$ is $N_1N^2$ where $N_1 = n(J_1)$. Bounds given in [32] allow us to argue that $NN_1 = O(p)$, an estimate quite accurate in practice. By the length estimate on the size of the solution of ExtendedEichlerNormEquation given above, we see that we can find solutions of norm $\ell^e$ where $e \sim 5\log(p) + 6\log(D) + 3\log(N)$.

*Remark 6.* In Algorithm 2, we treat the special case of $\mathfrak{O} = \mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$ as this is what is required for our VRF application. However, we can derive an algorithm to solve norm equations in a very large class of orders from Algorithm 1. Indeed, any quaternion order $\mathcal{O}$ can be decomposed as $\mathcal{O} = \mathbb{Z} + \mathfrak{f}\mathrm{Gor}(\mathcal{O})$ where $\mathrm{Gor}(\mathcal{O})$ is the Gorenstein closure of $\mathcal{O}$ (and is a Gorenstein order) and $\mathfrak{f} \in \mathbb{Z}$ is the conductor. More details on the topic of Gorenstein orders can be found in [39]. It is possible to design a generalization of SpecialOrderNormEquation which can solve norm equations in any quaternion order as soon as its Gorenstein closure are Eichler orders (i.e orders that can be expressed as an intersection of orders of the form $\mathbb{Z} + I$).

**Finding a basis of $\mathfrak{O}$.** The idea to find a basis is just to repeat the above algorithm for several $J_1$ (if we keep the same, then $J_1$ we obtain a basis of $\mathbb{Z} + D\alpha_1 J_1 \alpha_1^{-1}\overline{I_1}I_2$, which a strict suborder of $\mathfrak{O}$) until we have enough elements to make a generating family. Experimental results show that taking three such elements is already enough.

*Conjecture 1.* If $\theta_1, \theta_2, \theta_3$ are random outputs of $\mathsf{SpecialOrderNormEquation}(\mathsf{I_1}, \mathsf{I_2}, \mathsf{D})$, then $\mathfrak{O} = \mathsf{Order}(\theta_1, \theta_2, \theta_3)$ with good probability.

**Algorithm 3** $\mathsf{SmoothGen}_{\ell^\bullet}(I_1, I_2, D)$

**Require:** $I_1, I_2$ two distinct left $\mathcal{O}_0$-ideals of norm $N$ coprime with $D$.
**Ensure:** A generating family $\theta_1, \ldots, \theta_n$ for $\mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$ where each $\theta_j$ has norm $\ell^{e_j}$.
1: Set $L = \emptyset$ and $\mathfrak{O} = \mathbb{Z} + D((\mathbb{Z} + I_1) \cap (\mathbb{Z} + I_2))$.
2: Repeat $\alpha = \mathsf{SpecialOrderNormEquation}_{\ell^\bullet}(I_1, I_2, D)$ and $L = L \cup \{\alpha\}$ until there exists $\theta_1, \theta_2, \theta_3 \in L$ such that $\mathfrak{O} = \mathsf{Order}(\theta_1, \theta_2, \theta_3)$.
3: **return** $\theta_1, \theta_2, \theta_3$.

**Proposition 6.** *When the norm $N$ of $I_1$ and $I_2$ is coprime to $D$ and assuming Conjecture 1,* SmoothGen *terminates in probabilistic polynomial time and outputs a generating family of $\mathfrak{O}$.*

*Proof.* By Conjecture 1, we need only to repeat a polynomial number of times the algorithm SpecialOrderNormEquation which terminates in polynomial time by Proposition 5. By the termination condition, the output is a generating family of $\mathfrak{O}$.

**A deterministic algorithm for computing the generating family** For the VRF, we actually need SmoothGen to be a deterministic algorithm. As explained in [18], the sub-algorithm StrongApproximation can be made deterministic. With that in mind, it is easy to see that the two sources of randomness in SmoothGen come from the first steps of SpecialOrderNormEquation and ExtendedEichlerNormEquation, respectively. The EquivalentPrimeIdeal algorithm used in SpecialOrderNormEquation can be modified to run in a deterministic manner but we need to use a randomized version to obtain a generating family of $\mathfrak{O}$. The deterministic variant is obtained by defining an ordering on solutions and selecting the smallest one with respect to that ordering. With that idea, we can rerandomize consecutive executions by selecting solutions in increasing order. The same can be done in the random choice of $(C_2 : D_2)$ in the first step of ExtendedEichlerNormEquation by fixing an ordering on $\mathbb{P}^1(\mathbb{Z}/D\mathbb{Z})$.

### 6.4 Checking traces

In this section, we present an algorithm $\mathsf{CheckTrace}_M$ such that our VRF will achieve unconditional or computational uniqueness depending on the choice of $M$.

Computing the trace of an endomorphism is a well-studied problem, as it is the primary tool of the point counting algorithms such as SEA [38]. For our application the task is even simpler as we merely have to verify the correctness of the alleged trace value and not compute it.

The goal of $\mathsf{CheckTrace}_M$ is to verify the value of the traces mod $M$. We achieve this verification by using the formula $\mathrm{tr}(\theta) = \theta + \hat{\theta}$. Thus, it suffices to evaluate $\theta$ and $\hat{\theta}$ on a basis of the $M$-torsion, and then verify the relation. In particular, we do not need $M$ to be smooth.

The above protocol runs in polynomial time when $E[M]$ is defined in a field extension of degree $O(\log(p))$.

## 7 Parameters and Efficiency Analysis

In this section we discuss the choice of parameters to instantiate our VRF scheme at a given level of security $\lambda$ with the best possible efficiency. Then, we propose a concrete set of parameters for $\lambda = 128$ and NIST level 1 post-quantum security and assess the features of our construction.

**Algorithm 4** $\mathsf{CheckTrace}_M(E, \theta_1, \ldots, \theta_n, \omega_1, \ldots, \omega_n)$

**Require:** $\theta_1, \ldots, \theta_n$, $n$ endomorphisms of $E$ and $n$ elements of $B_{p,\infty}$ $\omega_1, \ldots, \omega_n$.
**Ensure:** A bit $b$ equal to 1 if and only if $\mathrm{tr}(\theta_i) = \mathrm{tr}(\omega_i) \mod M$ for all $i \in [1, n]$.
1: Compute $P, Q$ a basis of $E[M]$ over the appropriate field extension. Set $b = 1$.
2: **for** $\mathcal{I} \subset [1, n]$ **do**
3: Set $\theta_{\mathcal{I}} = \prod_{j \in I} \theta_j$ and $\omega_{\mathcal{I}} = \prod_{j \in I} \omega_j$.
4: Verify $\theta_{\mathcal{I}}(R) + \hat{\theta}_{\mathcal{I}}(R) = [\mathrm{tr}(\omega_{\mathcal{I}})]R$ for $R \in \{P, Q\}$. If not, set $b = 0$.
5: **end for**
6: **return** $b$.

### 7.1 Choice of parameters

**Choosing the prime $p$.** Generic attacks against the endomorphism ring computation problem imply that we must take $\log(p) \geq 2\lambda$ and this is the only real requirement for security. However, for efficiency's sake we need a prime of a very special form. Indeed, we must be able to apply the algorithms from [18] to compute the proof $\pi$ of our VRF. Thus, we look for $p$ such that $p^2 - 1 = \ell^e T f_+ f_-$ where $\ell$ is a small prime, $T$ is smooth and coprime with $\ell$, $T^2 \sim p^3$, and $f_\pm$ divides $p \pm 1$. Then, we choose the parameter $N$ as a prime divisor of the remaining factor $f_+$ (for reasons explained at the end of this section when we treat the generation of the additional public parameters $P_0, I_{P_0}$). Hence we have the bound $\log N \leq \lambda$ when $\log p \sim 2\lambda$. As a consequence, for obvious reasons, we cannot have an output space of size bigger than $2 \log N \leq 2\lambda$.

**The choice of $E_0$** The base curve $E_0$ must also be chosen very carefully. For efficiency, the endomorphism ring of $E_0$ must be a special-extremal order as defined in [32] (otherwise we cannot apply the algorithm introduced in Section 6.3). This imply two things: $E_0$ is defined over $\mathbb{F}_p$, and $\mathrm{End}(E_0)$ contains a quadratic suborder $R$ of small discriminant that is orthogonal to $j$. Several examples of such maximal orders are given in [32]. The second constraint concerns both $E_0$ and $D$ and is necessary to obtain uniqueness. As stated in Theorems 1 and 2, there must not be any pair of distinct $D$-isogenies between $E_0$ and any other curve $E$. In fact, we can state a very concrete condition on $E_0$ and $D$ to ensure that.

**Proposition 7.** *Let $E_0$ be a special extremal curve as described above. Let us write $R$ for the quadratic order of small discriminant $\Delta$ embedded in $\mathcal{O}_0$. If $\mathrm{End}(E_0)$ does not contain any non-trivial endomorphism of norm 1, $D$ is inert in $R$ and $D^2 < p/\Delta$, then there cannot be two distinct isogenies of degree $D$ between $E_0$ and a curve $E$.*

*Proof.* Two distinct $D$-isogenies between $E_0$ and $E$ would imply the existence of a non-trivial endomorphism $\alpha$ of norm $D^2$ in $E_0$. There are two possibilities: either $\alpha \in R$ or $\alpha \notin R$. Since $D$ is inert in $R$ and $R$ does not contain any element of norm 1 apart from $\pm 1$, the only endomorphisms of norm $D^2$ are the

trivial scalars $\pm D$. Thus, $\alpha$ must not be contained in $R$, and $\mathbb{Z}[\alpha]$ is a distinct quadratic order embedded in $\mathrm{End}(E_0)$. By a classical theorem from Kaneko [30], we know that $|\Delta\Delta'| \geq 4p$ where $\Delta' = \mathrm{disc}\ \mathbb{Z}[\alpha]$. The formula $\Delta' = \mathrm{tr}(\alpha)^2 - 4n(\alpha)$ gives the bound $|\Delta'| \leq 4D^2$. Injecting this inequality into the bound above we obtain that $D^2$ must be bigger than $p/\Delta$. This contradicts our assumption on $D$, and proves that there cannot be any non-trivial endomorphisms of norm $D^2$ in $\mathrm{End}(E_0)$.

**The prime $D$.** The choice of prime $D$ is only mildly dependent on the value of $p$. As such, there are a lot of possibilities for $D$ when $p$ has been fixed. The constraints on this prime are mainly derived from security requirements. First, by Proposition 7 and Theorems 1 and 2, uniqueness requires that $D < \sqrt{p/\Delta}$. Also, for security against key recovery attacks, we need $\log D \geq \lambda$. Indeed, since $D$ is prime, the brute-force attack to find the ideal of the secret isogeny $\varphi$ has complexity $O(D)$ (see the analysis of key recovery in [18]). The two bounds above suggest $D$ must have exactly $\lambda$ bits (since we are going to choose $\log p$ as close as possible to the lower-bound $2\lambda$). However, as pointed out in Section 5.1, this is not the only requirement. Indeed, for pseudo-randomness we need also that computing an isogeny of degree $D$ is hard. The recent work [4] has introduced a method to compute an isogeny $\varphi$ of large prime degree $D$ in $O(\sqrt{D})$ operations over the field of definition of $\ker \varphi$. Thus, we cannot merely rely on the size of $D$ to ensure the hardness of Problem 1 (as pointed out above, we must have $\log D = \lambda$, which would give an attack in $O(2^{\lambda/2})$). Thus, the only other solution is to choose $D$ such that $\ker \varphi$ is defined over a field extension of very high degree. In particular, let us write $k_D$ for the smallest integer such that there exists a supersingular curve $E$ over $\mathbb{F}_{p^{k_D}}$ with $E[D] \subset E(\mathbb{F}_{p^{k_D}})$. If we can ensure that $\log k_D \sim \lambda$, then the complexity to compute an isogeny of degree $D$ should be bigger than $2^\lambda$. The following result can be derived from [40, Theorem 4.1] and gives us a method to choose such a $D$.

**Lemma 7.** *If $p = 1 \mod 3$ and $E$ is a supersingular elliptic curve over $\mathbb{F}_{p^2}$ then:*

- *$k$ is odd and $\#E(\mathbb{F}_{p^k}) = p^k + 1$.*
- *$k$ is even and $\#E(\mathbb{F}_{p^k}) = p^k + 1$ or $\#E(\mathbb{F}_{p^k}) = (p^{k/2} \pm 1)^2$.*

For $\ker \varphi$ to be defined over $\mathbb{F}_{p^k}$ we must have $D | \#E(\mathbb{F}_{p^k})$. Thus, $k_D$ is (possibly up to a factor 2), the smallest integer $k$ such that $p^k = 1 \mod D$. The multiplicative group $\mathbb{Z}/D\mathbb{Z}^\times$ has $D-1$ elements. If $D = 2D' + 1$ where $D'$ is a also prime number then, unless $p^2 = 1 \mod D$, we have that $k_D \geq D'$. More generally, if $D-1$ is equal to a large prime $D'$ multiplied by a few small factors, then with good probability we will have $k_D \geq D'$. To summarize: we need a prime $D$ of exactly $\lambda$-bits such that $D$ is inert in the quadratic order $R$ and that $D'$, the biggest prime factor of $D-1$, is approximately equal to $2^\lambda$. Under standard results on prime distribution, we can find such a $D$ in polynomial time.

**The parameter $M$** We indicate two choices of $M$, one for unconditional uniqueness and one for efficient verification under computational uniqueness. We label $M_u$ the former, and $M_c$ the latter. A lower bound on $M_u$ is dictated by Theorem 1. The norms of the endomorphisms whose traces we need to check must be smaller than $(1/4)M^{2/3}$. From Section 6.3 and Lemma 4, we need to verify the traces of: $\theta_1, \theta_2, \theta_3, \theta_1\theta_2, \theta_1\theta_3, \theta_2\theta_3, \theta_1\theta_2\theta_3$ where $\theta_1, \theta_2, \theta_3$ are obtained from the SpecialOrderNormEquation algorithm. Estimates from Section 6.3 predict that we can find with good probability endomorphisms of norm $\ell^e$ where $e \sim 5\log(p) + 6\log(D) + 3\log(N)$ and the choices of $p, D, N$ give $e \sim 16\lambda + 3\log(N)$. This is the best we can say in full generality, as the value of $\log N$ cannot be predicted for sure. From there, we can derive that $\log M_u$ must be bigger than $24\lambda + 9/2\log(N)$. At most, we will have $\log(N) \sim \lambda$, thus giving a range of $[24\lambda, 57/2\lambda]$ for $\log M_u$. For the concrete value $M_u$, we recommend to find the smallest field extension $F_{p^k}$ such that there exists an integer $M_u$ of size above the desired bound with $E[M_u] \subset E(\mathbb{F}_{p_k})$.

For efficiency, we choose $M_c = \mathrm{lcm}(p-1, p+1) = (p^2-1)/2$. Indeed, we can find isomorphic models $E$ and $E'$ over $\mathbb{F}_{p^2}$ such that $\#E(\mathbb{F}_{p^2}) = (p-1)^2$ and $\#E'(\mathbb{F}_{p^2}) = (p+1)^2$ (this idea is used in both [18,14]). Thus, we can compute the traces mod $M_c$ by evaluating the endomorphisms twice over $\mathbb{F}_{p^2}$. Since $\log M_c = 4\lambda$, this seems like an acceptable compromise between security and efficiency.

**Computation of remaining public parameters** Now that we have specified the choices of all the integral parameters, we need to explain how to compute the remaining public parameters of our scheme. In particular, we need to find a basis $P_0, Q_0$ of $E_0[N]$, an endomorphism $\iota$ such that $\iota(P_0) = Q_0$ and the kernel ideal $I_{P_0}$. As we outlined in Section 6.1, this operation is not trivial as $N$ is a large prime number. This is where the specific choice of $N$ will come into play. Since $N$ divides $p+1$, there exists a subgroup $\langle P_0 \rangle$ of order $N$ in $E_0(\mathbb{F}_p)$ and it can be easily computed. These points are left invariant by the Frobenius endomorphism $\pi$ of $E_0$. Thus, $\langle P_0 \rangle \subset \ker \pi - 1$ and $I_{P_0} = \mathcal{O}_0\langle j-1, N\rangle$. For $\iota$, any element of $\mathrm{End}(E_0)$ not sending $P_0$ to $\langle P_0 \rangle$ can be used (it suffices to take any $\iota \in \mathcal{O}_0 \setminus (\mathbb{Z} + I_{P_0})$ and from there $Q_0$ can be easily computed.

### 7.2 Concrete values for $\lambda = 128$ with efficiency and size estimates.

**Example parameters** We now describe concrete parameters for $\lambda = 128$ to reach NIST level-1 post-quantum security. We follow [18] and take the 256-bits prime $p$ to be:

$$\begin{aligned} p+1 &= 2^{33} \cdot 5^{21} \cdot 7^2 \cdot 11 \cdot 31 \cdot 83 \cdot 107 \cdot 137 \cdot 751 \cdot 827 \cdot 3691 \cdot 4019 \cdot 6983 \\ &\quad \cdot 517434778561 \cdot 26602537156291\,, \\ p-1 &= 2 \cdot 3^{53} \cdot 43 \cdot 103^2 \cdot 109 \cdot 199 \cdot 227 \cdot 419 \cdot 491 \cdot 569 \cdot 631 \cdot 677 \cdot 857 \cdot 859 \\ &\quad \cdot 883 \cdot 1019 \cdot 1171 \cdot 1879 \cdot 2713 \cdot 4283\,. \end{aligned}$$

More details on the search of primes of this form are given in [18,14,15]. Then, we can take $\ell = 2$ and the smooth integer $T$ to be the product of all odd divisors of $(p-1)(p+1)$ that are smaller than 6983.

Since $p^2 - 1 = 2^{34} \cdot T \cdot 517434778561 \cdot 26602537156291$ and $517434778561 \cdot 26602537156291$ divides $p+1$, we have two possibilities for $N$. We choose the biggest one in order to have the larger output space and fix $N = 26602537156291$, a 44-bit prime.

Our chosen prime $p$ satisfies $p = 3 \mod 4$ and $p = 1 \mod 3$. We can choose the quaternion algebra to be $B_{p,\infty} = H(-1, -p)$. In this case, we have a very nice example of extremal order $\langle 1, i, \frac{1+j}{2}, \frac{i+k}{2} \rangle$ which is the endomorphism ring of the curve $y^2 = x^3 + x$ of j-invariant 1728. Unfortunately we cannot use this curve as $E_0$ as there is a non-trivial automorphism corresponding to the element $i$. When $p = 1 \mod 3$, this is the only such problematic curve so this leaves several other suitable choices for $E_0$. The most natural one is probably the curve of $j$-invariant 287496. This curve is 2-isogenous to $y^2 = x^3 + x$ and is defined over $\mathbb{F}_p$, which means that $\mathrm{End}(E_0)$ contains the nice suborder $\langle 1, 2i, j, 2k \rangle$. To choose $D$ inert in $\mathbb{Z}[2i]$, it suffices to take $D = 3 \mod 4$. Then, we look for $D$ such that $(D-1)/2$ is prime as well. For instance we can take $D = 257974546961629614027466807744098283 07$, but there are a lot of other possibilities.

With these choices, our heuristic estimates predicts that we can find endomorphisms $\theta_1, \theta_2, \theta_3$ with degree $\ell^e$ where the exponents are around $16 \times 128 + 3 \times 44 = 2180$.

Finally, we must have $\log M_u \geq 24 \cdot 128 + 9/2 \cdot 44 = 3270$. Then, we can take $M_u = p^{13} + 1$ which satisfies $\log M = 3319$. It can easily be verified that the $M_u$ torsion is defined over $\mathbb{F}_{p^{26}}$ for the supersingular elliptic curves of characteristic $p$. This is an extension of constant degree 13 over $\mathbb{F}_{p^2}$ the field of definition of all our curves.

In comparison, when taking $M_c = \mathrm{lcm}(p-1, p+1) = 2^{33} \cdot T \cdot N \cdot 517434778561$, we can get all the relevant torsion points defined over $\mathbb{F}_{p^2}$ (but on two distinct models). Should $M_c = (p^2-1)/2$ prove to be too small of a bound to retain reasonable security, there is a wide range of choices between $p^2 - 1/2$ and $p^{13} + 1$.

**Efficiency analysis**

*Evaluation* The main cost in VRFEval is clearly the computation of the representations of $\theta_1, \ldots, \theta_n$. As conjectured in Section 6.3, we only need 3 endomorphisms $\theta_1, \theta_2, \theta_3$ to describe a basis of the order $\mathfrak{O}$. We achieve the computation of the representation of an endomorphism $\theta$ by considering it as an ideal and then apply the $\mathsf{IdealToIsogeny}_{\ell^\bullet}$ algorithm from [18]. It is easy to see that the complexity of this algorithm grows linearly in the exponent of the ideal in input. In the signature from [18], this computation is performed on ideals of norm $2^{1000}$, whereas for our VRF we need to do this for three endomorphisms of norm $2^{2200}$. Hence, the computation should be roughly 6 times slower.

However, we stress out that in our case, the computation can be easily parallelized. Indeed, the three computations corresponding to $\theta_1, \theta_2, \theta_3$ are totally

independent and can be performed at the same time. Furthermore, for each $\theta_i$ we can divide the task into 2 independant subtasks of half the size. Indeed $\theta_i : E \to E$ can be written as the composition of two isogenies $\hat{\sigma}_i \circ \rho_i$ where $\sigma_i : E \to E_i$ and $\rho : E \to E_i$ for some middle curve $E_i$. Representations of $\theta_i$ as $\sigma_i, \rho_i$ are completely sufficient to do what we want. Thus, our main computation for $\mathsf{VRFEval}$ can be divided into 6 independant executions of $\mathsf{IdealToIsogeny}$ on ideals of norm roughly $2^{1100}$. Hence, depending on the amount of parallelization accessible to the prover, the VRF evaluation's efficiency should be approximately ranging from 1 to 6 SQISign signatures. The C implementation of SQISign presented in [18] runs in roughly 2 seconds for signature.

*Verification* For the verification, the significant steps are:

1. Checking that the representations given by the prover are valid endomorphisms.
2. Performing $\mathsf{CheckTrace}_M$.
3. Verifying that the subgroups $H_i$ are stable under the endomorphisms.

The first item can be done by computing the isogeny from the representation provided in $\pi$. This is similar to the signature verification performed in [18], but for several isogenies (as explained above we can divide the task into the computation of 6 independent isogenies of size roughly equal to the signature isogeny from SQISign). In [18], the verification process takes around 40ms. The second and third item can be achieved by evaluating the given endomorphisms $\theta_1, \theta_2, \theta_3$ and the corresponding duals on a basis of $E[M]$. The cost of evaluating an isogeny of degree $2^e$ on a point is approximately equivalent to computing this isogeny when the points are defined over $\mathbb{F}_{p^2}$ which is the case when $M = M_c$, the bound for computation uniqueness.

The overhead for verification when opting for unconditional rather than computational uniqueness is proportional to the slow-down caused by performing operations over $\mathbb{F}_{p^{26}}$ rather than $\mathbb{F}_{p^2}$. The cost of multiplications, in particular, is increased by a factor between 13 to $13^2$ depending on the method used to perform arithmetic operations over $\mathbb{F}_{p^{26}}$.

*Key Generation* Given that our construction is only one-time, the cost of key generation has to be taken into consideration for the overall efficiency of the scheme. The main operation is to compute the isogeny $\varphi_J$ of degree $\ell^f$. This isogeny is computed from the ideal $J \sim I$ where $I$ is the ideal of norm $D$ corresponding to the secret isogeny $\varphi$. The ideal $J$ is found from $I$ by applying the $\mathsf{KLPT}$ algorithm from [32]. The best version of this algorithm finds $J$ of norm $\ell^f$ where $f \sim 3\log(p)$. Thus, key generation is essentially the translation of an ideal of norm $2^{768}$ to the corresponding isogeny. We can estimate, it takes 75% of one SQISign signature. Pushing the points $P_0, Q_0$ through $\varphi_J$ can be done at the same time as the computation of $\varphi_J$ at very small additional cost.

**Size estimates** Our VRF public key is made of a curve $E$ and two points $P_E, Q_E$. The representation of $E$ can be compacted to one element of $\mathbb{F}_{p^2}$, and a point can be described by an element of $\mathbb{F}_{p^2}$ and a bit. Thus, the size of our public key is 770 bits or 97 bytes.

The proof $\pi$ is made of the representation of three endomorphisms and two elements of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})$. The projective elements can be represented by $\log N = 44$ bits and we estimate the norm of each endomorphism to be around $2^{2200}$. Hence, the proof size of our VRF is 6688 bits or 836 bytes.

The output space has size $n_2(\lambda)$ which is at most $2\log(N) = 88$ bits or 11 bytes.

| | Public Key (bytes) | Proof (bytes) | Post-quantum | Few-times |
|---|---|---|---|---|
| EC - VRF [2] | 32 | 80 | No | No |
| BLS - VRF [1] | 96 | 48 | No | No |
| LB - VRF [22] | 3.29K | 4.66K | Yes | Yes |
| **This work** | 96 | 836 | Yes | Yes |

**Table 2.** Size of our one-time VRF compared to several classical schemes and a post-quantum few-times VRF based on lattices.

We can now illustrate (see Table 3) the advantage of our construction with respect to the lattice scheme from [22]. Indeed, it is clear that in terms of efficiency, our solution is not competitive at all with solutions based on lattices. However, isogenies offer key and proof sizes much closer to those obtained from classical solutions. In this regard, our VRF may be better suited to be integrated in blockchain applications such as Algorand [12]. In these types of applications, the number of transaction per second (TPS) for a given number of nodes is limited by the blocksize, which is a fixed parameter of the blockchain. Thus, if the size of VRF keys and proofs are too big, the throughput is going to be seriously reduced. The most-compact post-quantum VRF based on lattices is LB-VRF which was introduced in [22] where they estimated that their current sizes could not allow Algorand to reach 1000 nodes. Our VRF cost being significantly smaller than the one from LB-VRF, we see that using our VRF would allow to reach 1000 nodes with several hundred TPS.

The article [22] introduces a model to use a one-time VRF in Algorand. The following formula gives the number of transaction per second in this model:

$$\text{TPS} = \frac{\text{payload} - (\text{VRF cost} + \text{digest} + \text{signature}) \times \#\text{nodes}}{(\text{transaction size} + \text{signature size}) \times \text{blocktime}}$$

Following the parameters used in [22], we take blocktime as 5 seconds, transaction size as 1KB, digest as 32B and payload to be 5.4MB. Each transaction also contains a signature. The signature size will depend on the scheme used in the instantiation. This signature scheme need not be related to the VRF scheme we use.

For comparison, we use three different signature schemes: the classical signature Ed2559 of size 64 bytes and two post-quantum protocols: Falcon [23] a 700B signature based on lattices and SQISign [18], a 200B post-quantum signature built from isogenies. Given the many similarities between our construction and SQISign, this seems like a natural match.

We estimate the VRF cost as the size of key, proof and output. In our example, we get $943 = 836 + 96 + 11$.

| VRF + sign. | Assumptions | Post-quantum | 10 nodes | 100 nodes | 500 nodes | 1000 nodes |
|---|---|---|---|---|---|---|
| ECVRF + Ed25519 | ECC | No | 1000 | 1000 | 1000 | 1000 |
| LBVRF + Ed25519 | Lattice + ECC | Hybrid | 1000 | 882 | 246 | — |
| LBVRF + SQISign | Lattice + Isog. | Yes | 910 | 781 | 208 | — |
| LBVRF + Falcon | Lattice | Yes | 646 | 549 | 118 | — |
| **This work** + Ed25519 | Isogenies + ECC | Hybrid | 1000 | 1000 | 945 | 849 |
| **This work** + SQISign | Isogenies | Yes | 923 | 906 | 829 | 733 |
| **This work** + Falcon | Isogenies + Latt. | Yes | 654 | 637 | 559 | 462 |

**Table 3.** Projected performance comparison in terms of Transactions per seconds (TPS) for the Algorand blockchain using the model from [22] with different-sized signatures and VRF schemes. The symbol — indicates that the block size is too small to make one transaction.

# 8 Prospects and open questions

## 8.1 On the question of removing the one-time restriction

In this section, we discuss the prospects of removing the one-time restriction from our construction and present some ideas to do so. We are also going to illustrate the new technical challenges raised by these potential changes and why we chose to leave them for future work.

As explained in Section 4.2, our VRF function is the composition of a secret isogeny (through the bijection between the projective line over $\mathbb{Z}/N\mathbb{Z}$ and cyclic subgroups of order $N$) and a hash function. Even though the final output is derived from elements of the projective line, the bijection with the subgroups is explicit and can be computed by anyone. This fact is actually important since the subgroups are required to verify the computation's correctness. Indeed, the verification is done by checking that these subgroups are stable by endomorphisms inside a well-chosen suborder of the endomorphism ring. Unfortunately, these subgroups are also the cause for our scheme's one-timeness. Indeed, isogenies are morphisms over the $N$ torsion which is isomorphic to $\mathbb{Z}/N\mathbb{Z}^2$. Thus, as shown in Lemma 3, as soon as one knows the image of three subgroups of order $N$ through the secret isogeny, one can compute the image of any other subgroups

by solving a few DLPs. Hence, to remove the one-time restriction, we need to find a way to derive a random output (and verify it) without having to reveal the related subgroups. The first idea that comes to mind is to replace the subgroups by isogenies using the classic correspondence between cyclic subgroups of order $N$ and cyclic isogenies of degree $N$. Given a subgroup $G$, let us write $\varphi_G : E \to E/G$ for the associated isogeny. The natural idea is then to compute the final output from the $j$-invariant of the codomain $E/G$ as in the CGL hash function [11] or the KDM based on SIDH [29]. We can develop this idea in two directions with distinct verification mechanisms: either we reveal the isogeny $\varphi_G$ in the proof or we don't.

In the first case, we must take the order $N$ to be smooth (to make $\varphi_G$ efficiently computable and representable) and such that the elements of the $N$-torsion are not efficiently representable (otherwise it is easy to compute $G$ from $\varphi_G$ and our variant is equivalent to the one-time construction). Then, we can verify the computation quite similarly to the one-time construction. Indeed, when $G$ is left stable by an endomorphism $\theta$, the pushforward $[\theta]_*\varphi_G$ is equal to $\varphi_G$. Thus, it suffices to compute $[\theta]_*\varphi_G$. The usual method uses the kernels as in SIDH, but in can be done without it by forming an isogeny ladder as in [13]. This is much less efficient but can still be done in reasonable time.

In the second case (where we want to avoid revealing the isogeny $\varphi_G$ in the proof) we can keep $N$ as a prime. However, verification becomes more complicated. We can use the following fundamental property: when $G$ is stable under the action of $\varphi_G$, $\theta$ is also embedded in $\mathrm{End}(E/G)$. Thus, if the proof include a curve $E_1$ together with the embedding of our well-chosen suborder inside $\mathrm{End}(E_1)$ we could hope to verify that $E_1$ was correctly computed. Unfortunately this is not enough. Indeed, a curve $E_1$ with the embedding of the correct suborder $\mathbb{Z} + D\mathfrak{O}_0$ could have been computed from any curve $E'$ that is $D$-isogenous to $E_0$. Thus, we need an additional information to tie the curve $E_1$ with the public key $E$. The only plausible solution seems to be to reveal an isogeny between $E$ and $E_1$ and show that it ties together the embeddings of $\mathbb{Z} + D\mathfrak{O}_0$ inside both $\mathrm{End}(E_1)$ and $\mathrm{End}(E)$. However, it is not exactly clear how one would compute this isogeny and we leave this for future work.

Additionally, these two variants raise the issue of security and pseudo-randomness in particular. If we remove the few-times restriction, the adversary has access to a polynomial number of calls to the evaluation oracle. In the variant where $\varphi_G$ is included in the proof, the pseudo-randomness would rely on the difficulty to do the analog of Lemma 3 when subgroups have been replaced by the corresponding isogenies. As we explained, a necessary condition for this problem to be difficult is that the $N$-torsion must not be efficiently representable (otherwise it is easy to recover $G$ from $\varphi_G$ as in Lemma 3). Even in that case, arguing about the hardness of the pseudo-randomness problem would require yet-another assumption, and one that appears to be far more dubious than the ones we already introduced for the one-time construction. For this reason, we leave further analysis of this non-restricted variant to future work in hope to build more confidence about this potential new security problem.

For the second non-restricted variant, pseudo-randomness could be easier to argue. By essence, we do not reveal $\varphi_G$ in this variant and so applying something analog to Lemma 3 seems a lot more difficult. However, without a detailed description of the verification mechanism we cannot even formulate the precise pseudo-randomness problem so it is still too early to draw any meaningful conclusions about its hardness.

### 8.2 Potential for other cryptographic applications

We introduced our VRF construction as a mean to illustrate the possibilities offered by the new isogeny proof of knowledge mechanism outlined in Section 3. In this section, we discuss two other potential applications. As in Section 8.1, we propose directions to explore for future work rather than concrete protocols.

*Verifiable Delay Functions (VDF)* are similar to VRF in the sense that a VDF is made of a function $f$ whose evaluation at a given $x$ must be verified efficiently. The security properties are quite different though, as the main feature of a VDF is *sequentiality*: given a prescribed time $T$, evaluating the VDF should not be possible in less time than $T$. In comparison, verification should run in $O(\log(T))$ or even constant time (depending only of the security parameter). The first ideas of VDF where based on the computation of $T$ repeated squarings ($x \mapsto x^{2^T}$ in groups of unknown order [41]). We mentioned already in Section 2 that there is a VDF construction based on isogenies [19]. This scheme share some similarities with our VRF construction as the VDF function is an isogeny. Sequentiality comes from the degree of this isogeny which is $2^T$ (adapting the idea of repeated squaring to the case of isogenies). However, their scheme is not post-quantum as pairings are used to verify the correctness of the computation.

The natural idea is to try and see if we could use our new proof ideas to verify the computation of this VDF. If we keep the degree of the isogeny to be evaluated as $2^T$, then our proof technique will not fit the VDF framework as the norm of the endomorphisms given in the proof would be in $O(T)$ and so the proof would not be efficient to verify However, if the size of the degree of the VDF isogeny is fixed, then we can obtain an efficient verification. In this setting, we would need to use the smoothness of the degree to play on the difficulty to evaluate the isogeny: the bigger the prime factors in the degree are, the longer the computation will take. Sequentiality could be proven by assuming a lower bound on the asymptotic complexity to compute an isogeny of a given prime degree $\ell$. Given the recent advances [4] in the problem of evaluating isogenies of prime degree it is hard to estimate how much confidence we can put on the fact that the best currently known algorithms are essentially optimal.

*Trapdoor mechanism from endomorphisms revelation* Our second potential application is rather a generic mechanism than a precise protocol. The goal is to try and exploit the idea that revealing some endomorphisms of a supersingular curve is not necessarily problematic. The encryption scheme Séta from [37] is a good example of a protocol where the trapdoor is some endomorphism of the

public key curve. In this protocol, the participant can usually compute the endomorphism ring of the public key curve during key generation. But we could imagine a situation where one participant $P_1$ generates a curve $E$ (and compute its endomorphism ring along the way) before revealing a well-chosen endomorphism of $E$ to another participant $P_2$. Then, $P_2$ could use this endomorphism to perform some protocol (for instance the Séta encryption scheme) without knowing anything else on the curve $E$.

It seems tempting to try to build IBE from this setting. For instance, the master public key could be a curve $E$ with the master secret key as $\mathrm{End}(E)$, identities would be isogenies from $E$ to $E_{id}$ and the corresponding secret key, an endomorphism of $E_{id}$ that can be used as a Séta secret key. Unfortunately, it seems hard to choose these secret keys in a way that would prevent an adversary who has access to several such secret endomorphisms to recover enough information to generate secret keys for himself and break the IBE security. Even though IBE appears to be out of reach from this idea, lesser primitive could still be achievable.

### 8.3 Conclusion

We have introduced the first post-quantum VRF construction based on isogenies. Our protocol is only one-time, and relies on a new security assumption, but it offers the best sizes among post-quantum solutions. In terms of efficiency, our solution is not competitive with lattices-based schemes but it is reasonably efficient in the isogeny-based landscape. To obtain unconditional uniqueness, the performances of the verification protocol are not satisfying, and we offer a security/efficiency tradeoff at the cost of an additional security assumption.

This new protocol is built upon two blocks that may be of independent interest: a new compact proof of isogeny knowledge method, and a new algorithm to solve norm equations in a large class of quaternion orders. In particular, our proof of isogeny knowledge is the first protocol based on the explicit revelation of endomorphisms. A new security assumption stems from this principle. Works needs to be done to understand this new assumption.

Among the prospects for efficiency improvements, we can list: reducing the norm of the elements given in output of our new norm equation algorithm, improving the efficiency of the ideal to isogeny algorithm from [18] and improving the efficiency of unconditional uniqueness either by finding a new algorithm to compute traces exactly or by reducing the theoretical bound on the trace modulus required to perform the verification.

Finally, we have exposed several directions to explore in order to either remove the one-time restriction from our construction or construct new applications from our isogeny proof of knowledge framework. More analysis is required to assess if concrete schemes could be derived from these ideas.

## References

1. Algorand: Source code of bls-vrf https://github.com/algorand/bls_sigs_ref

2. Algorand: Source code of ec-vrf https://github.com/algorand/libsodium
3. Bank, E., Camacho-Navarro, C., Eisentraeger, K., Morrison, T., Park, J.: Cycles in the supersingular $\ell$-isogeny graph and corresponding endomorphisms (04 2018)
4. Bernstein, D.J., Feo, L.D., Leroux, A., Smith, B.: Faster computation of isogenies of large prime degree. ANTS (2020)
5. Bernstein, D.J., Hamburg, M., Krasnova, A., Lange, T.: Elligator: Elliptic-curve points indistinguishable from uniform random strings. In: Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security. pp. 967–980 (2013)
6. Beullens, W., Kleinjung, T., Vercauteren, F.: Csi-fish: Efficient isogeny based signatures through class group computations. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 227–247. Springer (2019)
7. Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: International Conference on the Theory and Applications of Cryptographic Techniques. pp. 416–432. Springer (2003)
8. Boneh, D., Kogan, D., Woo, K.: Oblivious pseudorandom functions from isogenies. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 520–550. Springer (2020)
9. Burdges, J., Feo, L.D.: Delay encryption. Cryptology ePrint Archive, Report 2020/638 (2020), https://eprint.iacr.org/2020/638
10. Castryck, W., Lange, T., Martindale, C., Panny, L., Renes, J.: Csidh: an efficient post-quantum commutative group action. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 395–427. Springer (2018)
11. Charles, D.X., Lauter, K.E., Goren, E.Z.: Cryptographic hash functions from expander graphs. Journal of Cryptology **22**(1), 93–113 (Jan 2009)
12. Chen, J., Gorbunov, S., Micali, S., Vlachos, G.: Algorand agreement: Super fast and partition resilient byzantine agreement. Cryptology ePrint Archive, Report 2018/377 (2018), https://eprint.iacr.org/2018/377
13. Colò, L., Kohel, D.: Orienting supersingular isogeny graphs. Number-Theoretic Methods in Cryptology 2019 (2019)
14. Costello, C.: B-sidh: supersingular isogeny diffie-hellman using twisted torsion. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 440–463. Springer (2020)
15. Costello, C., Meyer, M., Naehrig, M.: Sieving for twin smooth integers with solutions to the prouhet-tarry-escott problem
16. David, B., Gaži, P., Kiayias, A., Russell, A.: Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 66–98. Springer (2018)
17. De Feo, L., Galbraith, S.D.: Seasign: Compact isogeny signatures from class group actions. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 759–789. Springer (2019)
18. De Feo, L., Kohel, D., Leroux, A., Petit, C., Wesolowski, B.: Sqisign: compact post-quantum signatures from quaternions and isogenies. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 64–93. Springer (2020)
19. De Feo, L., Masson, S., Petit, C., Sanso, A.: Verifiable delay functions from supersingular isogenies and pairings. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 248–277. Springer (2019)

20. Eisenträger, K., Hallgren, S., Lauter, K., Morrison, T., Petit, C.: Supersingular isogeny graphs and endomorphism rings: Reductions and solutions. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology – EUROCRYPT 2018. pp. 329–368. Springer International Publishing (2018)
21. Eisenträger, K., Hallgren, S., Leonardi, C., Morrison, T., Park, J.: Computing endomorphism rings of supersingular elliptic curves and connections to path-finding in isogeny graphs. Open Book Series **4**(1), 215–232 (2020)
22. Esgin, M.F., Kuchta, V., Sakzad, A., Steinfeld, R., Zhang, Z., Sun, S., Chu, S.: Practical post-quantum few-time verifiable random function with applications to algorand. Cryptology ePrint Archive, Report 2020/1222 (2020), https://eprint.iacr.org/2020/1222
23. Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: Falcon: Fast-fourier lattice-based compact signatures over ntru. Submission to the NIST's post-quantum cryptography standardization process **36** (2018)
24. Galbraith, S.D., Petit, C., Silva, J.: Identification protocols and signature schemes based on supersingular isogeny problems. In: ASIACRYPT (2017)
25. Gilad, Y., Hemo, R., Micali, S., Vlachos, G., Zeldovich, N.: Algorand: Scaling byzantine agreements for cryptocurrencies. In: Proceedings of the 26th Symposium on Operating Systems Principles. pp. 51–68 (2017)
26. Goyal, R., Hohenberger, S., Koppula, V., Waters, B.: A generic approach to constructing and proving verifiable random functions. In: Theory of Cryptography Conference. pp. 537–566. Springer (2017)
27. H. Silverman, J.: The Arithmetic of Elliptic Curves, vol. 106 (01 2009)
28. Jao, D., De Feo, L.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In: Yang, B.Y. (ed.) Post-Quantum Cryptography. pp. 19–34. Springer Berlin Heidelberg, Berlin, Heidelberg (2011)
29. Jao, D., Soukharev, V.: Isogeny-based quantum-resistant undeniable signatures. In: International Workshop on Post-Quantum Cryptography. pp. 160–179. Springer (2014)
30. Kaneko, M.: Supersingular $j$-invariants as singular moduli ${mod}, p$ (1989)
31. Kohel, D.: Endomorphism rings of elliptic curves over finite fields. Ph.D. thesis, University of California at Berkeley (1996)
32. Kohel, D., Lauter, K.E., Petit, C., Tignol, J.P.: On the quaternion $\ell$-isogeny path problem. IACR Cryptology ePrint Archive **2014**, 505 (2014)
33. Kutas, P., Martindale, C., Panny, L., Petit, C., Stange, K.E.: Weak instances of sidh variants under improved torsion-point attacks. Cryptology ePrint Archive, Report 2020/633 (2020), https://eprint.iacr.org/2020/633
34. Micali, S., Rabin, M., Vadhan, S.: Verifiable random functions. In: 40th annual symposium on foundations of computer science (cat. No. 99CB37039). pp. 120–130. IEEE (1999)
35. Papadopoulos, D., Wessels, D., Huque, S., Naor, M., Včelák, J., Reyzin, L., Goldberg, S.: Making nsec5 practical for dnssec. Cryptology ePrintArchive, Report 2017/099 (2017)
36. Petit, C.: Faster algorithms for isogeny problems using torsion point images. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 330–353. Springer (2017)
37. de Saint Guilhem, C.D., Kutas, P., Petit, C., Silva, J.: Séta: Supersingular encryption from torsion attacks. Tech. rep., Cryptology ePrint Archive, Report 2019/1291, 2019. https://eprint. iacr. org ... (2019)

38. Schoof, R.: Counting points on elliptic curves over finite fields. Journal de théorie des nombres de Bordeaux **7**(1), 219–254 (1995)
39. Voight, J.: Quaternion Algebras. Springer Graduate Texts in Mathematics series (2018)
40. Waterhouse, W.C.: Abelian varieties over finite fields. Annales Scientifiques de l'E.N.S (1969)
41. Wesolowski, B.: Efficient verifiable delay functions. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 379–407. Springer (2019)
42. Yang, R., Au, M.H., Zhang, Z., Xu, Q., Yu, Z., Whyte, W.: Efficient lattice-based zero-knowledge arguments with standard soundness: construction and applications. In: Annual International Cryptology Conference. pp. 147–175. Springer (2019)
43. Yoo, Y., Azarderakhsh, R., Jalali, A., Jao, D., Soukharev, V.: A post-quantum digital signature scheme based on supersingular isogenies. In: International Conference on Financial Cryptography and Data Security. pp. 163–181. Springer (2017)

## A Proof of Proposition 2

The result stems from two preliminary results presented in Lemmas 8 and 9

**Lemma 8.** *Let $\mathcal{O}$ be a maximal order of $B_{p,\infty}$ and $I_1, I_2$ two integral left-$\mathcal{O}$ ideals of prime norm $N$. There exists at most three $j$-invariants (up to Galois conjugacy) such that the corresponding elliptic curves admit an embedding of $(\mathbb{Z}+I_1)\cap(\mathbb{Z}+I_2)$ in their endomorphisms rings. Among those three curves, there is only one such that the endomorphisms of $\mathfrak{O}$ stabilize two cyclic subgroups of order $N$.*

*Proof.* The Deuring correspondence puts in bijection isomorphism classes of maximal orders and supersingular j-invariants over $\mathbb{F}_{p^2}$ up to Galois conjugacy (i.e action of the Frobenius). Thus, to prove our result, we will show that $(\mathbb{Z}+I_1)\cap(\mathbb{Z}+I_2)$ is embedded in at most three maximal orders up to isomorphism.

Given $\mathfrak{O}$ an Eichler order of level $\ell^e$, it is a classical result (see [21]) that it is contained in $e+1$ maximal orders. If we write $\mathfrak{O} = \mathbb{Z} + J$ where $J$ is an integral ideal of norm $\ell^e$, then $\mathfrak{O}$ is an eichler order of level $\ell^e$ [18]. The factorization $J = J_1 \cdots J_e$ allows us to compute these $e+1$ maximal orders and the corresponding embeddings. Writing $\mathcal{O}_i = \mathcal{O}_L(J_{i+1})$ for $i \in [0, e-1]$ and $\mathcal{O}_e = \mathcal{O}_R(J_e)$, we get that $\mathfrak{O} = (\mathbb{Z} + \overline{J_0 \dots J_i}) \cap (\mathbb{Z} + J_{i+1}) \cdots J_e \subset \mathcal{O}_i$ where $J_0 = \mathcal{O}_0$ and $J_{e+1} = J_e^{-1}$. Now we can assign curves $E_i$ to each maximal order $\mathcal{O}_i$ through the Deuring correspondence. Each ideal $J_i$ represents an $\ell$-isogeny $\varphi_i$ between $E_{i-1}$ and $E_i$. If we write $\psi_{i,1} = \hat{\varphi}_0 \circ \cdots \circ \hat{\varphi}_i$ and $\psi_{i,2} = \varphi_e \circ \cdots \circ \varphi_{i+1}$ then the endomorphisms of the embedding of $\mathfrak{O}$ in each $\mathrm{End}(E_i)$ stabilize the two subgroups $\ker \psi_{i,1}$ and $\ker \psi_{i,2}$ and no other subgroups.

Now it is easily verified that $(\mathbb{Z}+I_1)\cap(\mathbb{Z}+I_2)$ can be written as $\mathbb{Z}+J$ where $J = \overline{I_1} I_2$ has norm $N^2$. Then, if we apply the above reasoning we obtain the desired result with $\mathcal{O} = \mathcal{O}_1$. The two cyclic subgroups of order $N$ are $\ker \psi_{1,1}$ and $\ker \psi_{1,2}$. The other stable subgroups have either order 0 ($\ker_{\psi_{0,1}}$ and $\ker \psi_{2,2}$) or order $N^2$ ($\ker \psi_{0,2}$ and $\ker \psi_{2,1}$).

**Lemma 9.** *Let $D$ be a prime number different from $p$. When $\mathfrak{O} = \mathbb{Z} + D\mathfrak{O}_0$ is embedded in a maximal order $\mathcal{O}$, either $\mathcal{O}$ contains $\mathfrak{O}_0$ or there exists a left-$\mathcal{O}$ integral ideal of norm $D$ whose right order $\mathcal{O}_0$ contains $\mathfrak{O}_0$.*

*Proof.* Let us assume that $\mathfrak{O}_0$ is not contained in $\mathcal{O}$. Then we set $I = \{x \in \mathcal{O}, x\mathfrak{O}_0 \subset \mathcal{O}\}$. First, it is easy to verify that $I$ is an integral left $\mathcal{O}$-ideal. Then, we are going to see that it has norm $D$. It suffices to show that $D\mathcal{O} \subsetneq I \subsetneq \mathcal{O}$. To see that $I \neq \mathcal{O}$, it suffices to note that $1 \notin I$ since $\mathfrak{O}_0 \not\subset \mathcal{O}$. Then, with $D\mathfrak{O}_0 \subset \mathcal{O}$ we have $Dx\mathfrak{O}_0 = xD\mathfrak{O}_0 \subset \mathcal{O}$ for every $x \in \mathcal{O}$, which proves that $D\mathcal{O} \subset I$. Finally, to prove that $D\mathcal{O} \neq I$, we take $x_0 \in \mathfrak{O}_0$ and not contained in $\mathcal{O}$. It is clear that $Dx_0 \in I$, but $Dx_0 \notin D\mathcal{O}$. Finally, from the definition of $I$ it is quite clear that $\mathfrak{O}_0$ is contained in $O_R(I)$. This concludes the proof.

We are now ready to prove Proposition 2, that we recall here for the reader's convenience.

**Proposition 2** *Let $D$ and $N$ be distinct prime number different from $p$. Let $E_0$ be a supersingular curve defined over $\mathbb{F}_p$. Take $I_{G_1}$ and $I_{G_2}$ two $\mathcal{O}_0$-ideals of prime norm $N$, corresponding to the kernel ideals of two subgroups $G_1, G_2$ of order $N$ in $E_0[N]$. Given a supersingular curve $E$ not isomorphic to $E_0$, if there exists an embedding of $\mathfrak{O} = \mathbb{Z} + D((\mathbb{Z} + I_{G_1}) \cap (\mathbb{Z} + I_{G_2}))$ in $\mathrm{End}(E)$ and there exists two different subgroups $H_1, H_2 \subset E[N]$ stable under any endomorphism of $\mathfrak{O}$, there exists an isogeny $\varphi$ of degree $D$ between $E_0$ and $E$ and $H_i = \varphi(G_i)$ for $i = 1, 2$.*

*Proof.* First, we show that we can apply Lemma 9, by proving that $\mathfrak{O}_0 = (\mathbb{Z} + I_{G_1}) \cap (\mathbb{Z} + I_{G_2})$ cannot be embedded in $\mathrm{End}(E)$. If it were, since $E$ and $E_0$ are not isomorphic, then by Lemma 8 we would have that the only non-trivial subgroup stable under the endomorphisms of $\mathfrak{O}_0$ would be a cyclic subgroup of order $N^2$. The induced embedding of $\mathfrak{O} = \mathbb{Z} + D\mathfrak{O}_0$ would trivially respect the same property. This contradicts the assumption on $H_1$ and $H_2$, and thus proves that $\mathfrak{O}_0$ is not contained in $\mathcal{O}$. Then, by Lemma 9, there exists a maximal order $\mathcal{O}_0'$ connected to $\mathcal{O}$ through an ideal of norm $D$. We write $\varphi$ for the corresponding isogeny of degree $D$. We take $E_0'$ to be a supersingular elliptic curve with $\mathrm{End}(E_0') = \mathcal{O}_0'$. As explained in Section 3.1, every endomorphism $\alpha \in \mathfrak{O}$ can be decomposed as $[d] + \varphi\alpha_0\hat{\varphi}$, where $\alpha_0 \subset \mathfrak{O}_0$. It is easily verified that if $H$ is stable under $\mathfrak{O}$, then $\hat{\varphi}(H)$ is stable under $\mathfrak{O}_0$. Thus, $E_0'$ possesses two subgroups $\hat{\varphi}(H_i)$ for $i = 1, 2$ of order $N$ that are stable under the elements of $\mathfrak{O}$. By Lemma 8, $E_0'$ is isomorphic (up to Galois conjugacy) to $E_0$. Since $E_0$ is defined over $F_p$, the Galois group of $\mathbb{F}_{p^2}/\mathbb{F}_p$ acts trivially on $E_0$ and $E_0'$ is in fact isomorphic to $E_0$. This isomorphism must send $\hat{\varphi}(H_i)$ to $G_i$ (up to reordering). This is exactly the result we want to prove.

## B Provability and unbiasability

### B.1 Provability

**Definition 5.** *The VRF is provable if* $\mathsf{Verif}(pk, \pi, x, v) = 1$ *when* $(pk, sk) = \mathsf{KeyGen}(pp)$ *and* $(v, \pi) = \mathsf{VRFEval}(sk, x)$.

We show provability of our VRF computation in Proposition 8. Before that, we start with a simple preliminary result.

**Lemma 10.** *Given* $G_1, G_2$ *two cyclic subgroups of order* $N$ *in a supersingular curve* $E_0$*, the endomorphisms of norm coprime with* $N$ *contained in* $(\mathbb{Z} + I_{G_1}) \cap (\mathbb{Z} + I_{G_2})$ *are exactly the endomorphisms stabilizing both* $G_1$ *and* $G_2$.

*Proof.* It was proven in [18] that for $I$ an ideal of norm $N$, the elements of the Eichler order $\mathbb{Z} + I$ of norm coprime with $N$ are exactly the elements $\alpha$ such that $\alpha(G) = G$ for the kernel subgroup $G = E[I]$ associated to $I$. Applying this result to $I_{G_1}$ and $I_{G_2}$ at the same time proves the result.

**Proposition 8.** *The VRF introduced in Section 4.2 is provable.*

*Proof.* The order $\mathbb{Z} + D\mathcal{O}_0$ is contained in $\mathcal{O}_0 \cap \mathcal{O}_R(I) = \mathbb{Z} + I$ for any $I$ of norm $D$ (this is easy to see, since we can always decompose $I = \mathcal{O}_0\alpha_I + D\mathcal{O}_0$ for some $\alpha_I \in \mathcal{O}_0$). This proves that for any $\omega \in \mathbb{Z} + D\mathcal{O}_0$, there exists $\theta \in \mathrm{End}(E)$ with $\deg\theta = n(\omega)$ and $\mathrm{tr}(\theta) = \mathrm{tr}(\omega)$. As in $\mathsf{VRFEval}$ and $\mathsf{Verif}$, we set $\omega_1, \ldots, \omega_n = \mathsf{SmoothGen}(D, I_{G_1}, I_{G_2})$. Thus, when the $\theta_i$ are honestly computed from the $\omega_j$, it is clear that all the tests on the norm and traces of $\theta_j$ are performed successfully. It remains to show that $H_1, H_2$ are stable under each $\theta_j$. By Lemmas 2 and 10, we know that the only two subgroups stable under each of the $\omega_j$ are $G_1$ and $G_2$. By the construction of $y_1$ and $y_2$, we know there exist $w_1, z_1, w_2, z_2$ such that $x_{i,1} = B_{1,1}w_i + B_{2,1}z_i$ and $x_{i,2} = B_{1,2}w_i + B_{2,2}z_i$. In this case, from the definition of $B$ and $P_E, Q_E$ it can be verified easily that $H_i = \Psi_{P_E,Q_E}(y_i) = \varphi(G_i)$ for all $i \in \{1, 2\}$.

### B.2 Unbiasability

The final property to prove is unbiasability. This is a less common property for VRFs, but is necessary for some applications such as [16]. In any case, we achieve unbiasability quite easily from our existing assumptions.

**Definition 6.** *Let* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *be a polynomial-time adversary playing the following experiment:*

1. $pp \leftarrow\!\!: \mathsf{ParamGen}(1^\lambda)$
2. $(st, pk, v^*) = \mathcal{A}_1(pp)$.
3. $x \xleftarrow{\$} \{0,1\}^{n_1(\lambda)}$.
4. $(\pi, v) \leftarrow \mathcal{A}_2(x, st)$.
5. $b \leftarrow \mathsf{Verif}(pk, \pi, x, v)$.

*The VRF is unbiasable if*

$$\Pr(b = 1 \wedge v^* = v) \leq 2^{-n_2(\lambda)} + \mathrm{negl}(\lambda).$$

After we proved uniqueness, the unbiasability of our proposed VRF follows directly from the decomposition of our VRF function (see Section 4.2) as a permutation composed with a hash function.

**Theorem 3.** *Under the hardness of Problem 2 for the parameter $M$, the VRF scheme introduced in Section 4.2 is unbiasable in the random oracle model, if the parameters $E_0, p, D$ are such that there does not exist a curve $E$ with two distinct $D$-isogenies between $E_0$ and $E$.*

*Proof.* By the computational uniqueness of our VRF scheme proved under the same hypotheses in Theorem 2, we know that a pair $(\pi, v)$ of proof and output passing the verification for a given input $(x_1, x_2)$ must be such that $v = H(y_1, y_2)$ where $\Psi_{P_E,Q_E}(y_i) = \varphi(\Psi_{P_0,Q_0}(x_i))$ for $i \in \{1, 2\}$ with overwhelming probability. The map $(x_1, x_2) \mapsto (\Psi^{-1}_{P_E,Q_E}(\varphi(\Psi_{P_0,Q_0}(x_1))), \Psi^{-1}_{P_E,Q_E}(\varphi(\Psi_{P_0,Q_0}(x_2))))$ is a permutation of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta$, when $x_1, x_2$ is a uniformly random element of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta$, the output is $v = H(y_1, y_2)$ where $(y_1, y_2)$ is a uniformly random element of $\mathbb{P}^1(\mathbb{Z}/N\mathbb{Z})^2 \setminus \Delta$. This proves that $v$ is uniformly random in $\{0, 1\}^{n_2(\lambda)}$ in the random oracle model.