Computer Security Symposium 2025
27-31 October 2025

Attacks on PRISM-id via Torsion over Small Extension Fields

Kohei Nakagawa[1,a)] Hiroshi Onuki[2,b)]

Abstract: PRISM is an isogeny-based cryptographic framework that relies on the hardness of computing a large prime-degree isogeny from a supersingular elliptic curve with an unknown endomorphism ring. It includes both an identification scheme PRISM-id and a signature scheme PRISM-sig. In this work, we present two attacks on PRISM-id. First, we analyze the probability that a randomly sampled prime q in PRISM-id results in a q-torsion subgroup defined over a small extension field, and we show that this probability is higher than claimed in the original proposal. Exploiting this observation, we construct a forgery attack that computes a q-isogeny with success probability $O(2^{-3\lambda/4})$ and an expected time complexity $\tilde{O}(2^{3\lambda/4})$. Moreover, in a scenario where the adversary is allowed to reject challenge primes q, we present another forgery attack with expected time complexity $\tilde{O}(2^{6\lambda/7})$. Finally, we describe an attack on the underlying hardness assumption of PRISM-id, achieving an expected time complexity $\tilde{O}(2^{\lambda/2})$. Note that our results do not affect the security of PRISM-sig.

1. Introduction

Isogeny-based cryptography is a promising candidate for post-quantum cryptography. A representative scheme within isogeny-based cryptography is SQIsign [1], which is one of the candidates for the NIST post-quantum additional digital signature standard. An advantage of isogeny-based cryptography is its small key size compared to other post-quantum candidates.

Many of isogeny-based cryptographic protocols use supersingular elliptic curves and rely on the hardness of the *supersingular isogeny problem*, which requires computing an isogeny between two given supersingular elliptic curves. This problem is believed to be hard even in the quantum setting. Also, it is known that the supersingular isogeny problem is equivalent to the problem of computing the endomorphism ring of a supersingular elliptic curve [16], [19].

Recently, a new problem has been proposed and attracted attention: computing an isogeny of a given degree d from a given supersingular elliptic curve E. We call this the *given-degree isogeny problem.* If d is smooth or the endomorphism ring of E is known, then the given-degree isogeny problem can be solved efficiently. However, when d is a large prime and the endomorphism ring of E is unknown, no efficient algorithm is known to solve this problem.

The given-degree isogeny problem first appeared in the context of the security of SQIsignHD [9]. SQIsignHD and some variants of SQIsign including the NIST SQIsign rely on the hardness of the supersingular isogeny problem with oracle access to the given-degree isogeny problem. On the other hand, a new cryptographic framework PRISM [2] has been proposed, assuming the hardness of the given-degree isogeny problem with d being a large prime. PRISM consists of an identification scheme PRISM-id and a signature scheme PRISM-sig. PRISM-sig has a relatively simple structure, so it could serve as a building block for more advanced cryptographic protocols. Consequently, analyzing the hardness of the given-degree isogeny problem is important for isogeny-based cryptography. If it is found to be easy, then the security of some variants of SQIsign becomes clearer. If it is found to be hard, then the security of PRISM is guaranteed.

Contributions. We analyze the hardness of the given-degree isogeny problem. In particular, we focus on the setting used in PRISM. We show that there exists an algorithm with time complexity $\tilde{O}(2^{\lambda/2})$ that solves the underlying problem in PRISM with parameter in PRISM-id claimed λ-bit security.

In addition, we show that the proposed parameter of PRISM-id does not achieve the claimed security level. In particular, against the parameter of PRISM-id claimed λ-bit security, we propose a forgery attack with success probability $\tilde{O}(2^{-3\lambda/4})$ and an expected time complexity $\tilde{O}(2^{3\lambda/4})$, and another forgery attack in which the attacker can refuse challenges with expected time complexity $\tilde{O}(2^{6\lambda/7})$. Note that our attacks do not affect the security of PRISM-sig.

[1] NTT Social Informatics Laboratories, Japan
[2] The University of Tokyo, Japan
[a)] kohei.nakagawa@ntt.com
[b)] onuki@mist.i.u-tokyo.ac.jp

2. Preliminaries

In this section, we give some background knowledge used in this paper. Throughout this paper, p is a prime number greater than 3.

2.1 Supersingular Elliptic Curves and Quaternion Algebras

Let E' be a supersingular elliptic curve defined over $\overline{\mathbb{F}}_p$. Then there exists an elliptic curve E over $\mathbb{F}_{p^2}$ such that the p^2-th power Frobenius endomorphism on E is equal to the multiplication by $-p$. Then, for positive integers k and n, the group $E(\mathbb{F}_{p^{2k}})$ of rational points contains the n-torsion subgroup $E[n]$ if and only if $(-p)^k \equiv 1 \pmod{n}$. We assume that all the supersingular elliptic curves considered in this paper are defined over $\mathbb{F}_{p^2}$ and have the above property. In addition, we can assume that E is a Montgomery curve, which is defined by the equation $y^2 = x^3 + Ax^2 + x$ for some $A \in \mathbb{F}_{p^2}$. Thus, we assume that a supersingular elliptic curve E is represented by its coefficients A.

In this paper, we only consider quaternion algebras over $\mathbb{Q}$ ramified exactly at p and the infinite place. If p is fixed, such an algebra is unique up to isomorphism, thus we denote it by $B_{p,\infty}$. There exists $\mathrm{i}, \mathrm{j} \in B$ such that $(1, \mathrm{i}, \mathrm{j}, \mathrm{ij})$ is a basis of B as a $\mathbb{Q}$-vector space, i.e.,

$$B_{p,\infty} = \mathbb{Q} + \mathbb{Q}\mathrm{i} + \mathbb{Q}\mathrm{j} + \mathbb{Q}\mathrm{ij},$$

and that $\mathrm{i}^2 = -q$, $\mathrm{j}^2 = -1$, and $\mathrm{ij} = -\mathrm{ji}$, where $q = 1$ if $p \equiv 3 \pmod{4}$, $q = 2$ if $p \equiv 5 \pmod{8}$, and q is a prime such that $q \equiv 3 \pmod{4}$ and $\left(\frac{q}{p}\right) = -1$ otherwise.

If E is a supersingular elliptic curve over $\mathbb{F}_{p^2}$, then $\mathrm{End}(E)$ is isomorphic to a maximal order in the quaternion algebra $B_{p,\infty}$. Let $\mathcal{E}_p$ be the set of the isomorphism classes of supersingular elliptic curves over $\mathbb{F}_{p^2}$, and let $\mathscr{O}_p$ be the set of the isomorphism (as rings) classes of maximal orders in $B_{p,\infty}$. Then, we have the following map

$$\mathcal{E}_p \to \mathscr{O}_p, \quad E \mapsto \mathcal{O} \text{ such that } \mathcal{O} \cong \mathrm{End}(E).$$

This map is surjective and the images of two distinct elliptic curves are the same if and only if the two elliptic curves are p-th power Frobenius conjugate to each other. We call this map the *Deuring correspondence*.

Let E be a supersingular elliptic curve over $\mathbb{F}_{p^2}$, and let $\mathcal{O}$ be a maximal order in $B_{p,\infty}$ isomorphic to $\mathrm{End}(E)$. We fix an isomorphism $\iota : \mathcal{O} \to \mathrm{End}(E)$. For a left $\mathcal{O}$-ideal I, we define a set $E[I] := \bigcap_{\alpha \in I} \ker(\iota(\alpha))$. Then, $E[I]$ is a finite subgroup of $E(\mathbb{F}_{p^2})$. An isogeny with kernel $E[I]$ is called the *isogeny corresponding to the ideal I*.

In the case $p \equiv 3 \pmod{4}$, the elliptic curve E_0 defined by $y^2 = x^3 + x$ over $\mathbb{F}_{p^2}$ is supersingular. The endomorphism ring $\mathrm{End}(E_0)$ is isomorphic to the maximal order $\mathcal{O}_0$ defined by

$$\mathcal{O}_0 = \mathbb{Z} + \mathbb{Z}\mathrm{i} + \mathbb{Z}\frac{\mathrm{i} + \mathrm{j}}{2} + \mathbb{Z}\frac{1 + \mathrm{ij}}{2}.$$

An isomorphism is given by

$$((x, y) \mapsto (-x, \sqrt{-1}y)) \mapsto \mathrm{i} \text{ and } ((x, y) \mapsto (x^p, y^p)) \mapsto \mathrm{j},$$

where $\sqrt{-1}$ is a square root of -1 in $\mathbb{F}_{p^2}$.

2.2 Kani's Lemma

The following lemma, which is called *Kani's lemma* because it comes from Kani's work [13], is a key tool in the recent study of isogeny-based cryptography.

Lemma 1 (Theorem 1 in [15]). *Let d_1 and d_2 be coprime integers and let $D = d_1 + d_2$. Let E, E_1, E_2, and F be elliptic curves connected by the following commutative diagram of isogenies:*

$$\begin{tikzcd} E \arrow[r, "\varphi_1"] \arrow[d, "\varphi_2"'] & E_1 \arrow[d, "\varphi_2'"] \\ E_2 \arrow[r, "\varphi_1'"'] & F, \end{tikzcd}$$

where $\deg(\varphi_i) = \deg(\varphi_i') = d_i$ for $i \in \{1, 2\}$. Then, the map

$$\Phi = \begin{pmatrix} \varphi_1 & -\widehat{\varphi}_2' \\ \varphi_2 & \widehat{\varphi}_1' \end{pmatrix} : E \times F \to E_1 \times E_2$$

is a (D, D)-isogeny with respect to the natural product polarizations on $E_1 \times E_2$ and $E \times F$, and has kernel $\{([d_2]P, \varphi_2' \circ \varphi_1(P)) \mid P \in E[D]\}$. Conversely, a (D, D)-isogeny with this kernel is equal to Φ up to isomorphism.

This lemma tells us the method to compute all the 1-dimensional isogenies in the component of the 2-dimensional isogeny Φ from the restriction of $\varphi_2' \circ \varphi_1$ to $E[D]$ and the degrees d_1, d_2. Note that it is known that a (D, D)-isogeny can be computed from its kernel (e.g., [7], [14]). Especially when $D = 2^e$ for a positive integer e, we can compute the $(2^e, 2^e)$-isogeny efficiently by the formulas proposed by [10].

Remark 1. *A higher-dimensional version of Kani's lemma is known (Lemma 3.6 in [17]). It can be used to represent a 1-dimensional isogeny as we will explain in Section 2.3.*

2.3 Representation of Isogenies

An isogeny of degree d is a rational map consisting of polynomials of degree approximately d. If the degree d is exponential in $\log p$, which is the size of the coefficients of the elliptic curve, then we need to represent the isogeny by a more compact form. To describe such a representation, we introduce the notion of an *efficient representation* of an isogeny.

Definition 1. *Let d be a positive integer and $\varphi : E \to E'$ be an isogeny of degree d between elliptic curves over $\mathbb{F}_q$. An* efficient representation *of φ with respect to an algorithm $\mathcal{A}$ is some data $D_\varphi \in \{0, 1\}^*$ of polynomial size in* $\log q$ *and* $\log d$ *such that for all $P \in E(\mathbb{F}_{q^k})$, $\mathcal{A}$ with input (D_φ, P) returns E' and $\varphi(P)$ in polynomial time in* $\log(d)$ *and* $k \log(q)$*.*

If the degree d is smooth, then we can decompose the isogeny φ into a composition of isogenies of small degree. The sequence of the representations of these small degree isogenies as rational maps is an efficient representation of φ.

In the case where the degree d is not smooth, we can represent the isogeny φ by using Kani's lemma (Lemma 1). Assume that there exists a smooth integer n such that $n^2 > d$ and $E[n]$ is rational over $\mathbb{F}_{q^k}$, where k is a positive integer in polynomial size in $\log q$. Then, a tuple $(E, P, Q, \varphi(P), \varphi(Q), d)$, where (P, Q) is a basis of $E[n]$, is an efficient representation of φ (see Theorem 1 and Remark 1 in [17]). The computation of the isogeny by this representation requires computing a 8-dimensional isogeny in general, and is not efficient in practice.

In some special cases, we can improve the efficiency of the computation of the isogeny. Assume that there exist an odd integer d' and a positive integer a such that $d = d'(2^a - d')$ and $E[2^a]$ is rational over $\mathbb{F}_{q^k}$ for some positive integer k in polynomial size in $\log q$. Then, a tuple $(E, E', P, Q, \varphi(P), \varphi(Q), d')$, where (P, Q) is a basis of $E[2^a]$, is an efficient representation of φ. In this case, we can decompose φ into the composition of a d'-isogeny and a $(2^a - d')$-isogeny, and apply Kani's lemma (Lemma 1) to obtain efficient representations of these isogenies. Here, we can use an efficient method to compute $(2,2)$-isogenies by [10]. This representation is used in POKÉ [4] and PRISM [2]. We also use this representation in our attacks.

2.4 Ideal to Isogeny Translation

Translating an ideal into the corresponding isogeny under the Deuring correspondence is a fundamental operation in isogeny-based cryptography.

Let E be a supersingular elliptic curve over $\mathbb{F}_{p^2}$, and $\mathcal{O}$ be a maximal order in $B_{p,\infty}$ such that $\mathcal{O} \cong \mathrm{End}(E)$. Assume that we are given E, a basis of $\mathcal{O}$, and efficient representations of the endomorphisms of E corresponding to the basis elements. In this setting, an algorithm to compute an efficient representation of the isogeny corresponding to a given left $\mathcal{O}$-ideal I is called an *ideal-to-isogeny algorithm.*

In PRISM, the ideal-to-isogeny algorithm in [3] is used. This algorithm efficiently works if the characteristic p of the base field is of the form $p = f2^e - 1$ for integers f and e such that f is odd and $p \approx 2^e$. For the details of this algorithm, we refer the reader to [3], §3.2. Let E_0 and $\mathcal{O}_0$ be the elliptic curve and the maximal order defined in Section 2.1. We denote the ideal-to-isogeny algorithm for $\mathcal{O}_0$ by `IdealToIsogeny`. More precisely, given a left $\mathcal{O}_0$-ideal I, the algorithm computes an efficient representation of the isogeny from E_0 corresponding to I. Here, we omit the elliptic curve and the maximal order from the input of `IdealToIsogeny` since we regard these as system parameters.

2.5 Computational Problems

Isogeny-based protocols using supersingular elliptic curves are based on the hardness of the following computational problems.

Problem 1 (Supersingular isogeny problem). *Given two supersingular elliptic curves E and E' over $\mathbb{F}_{p^2}$, find an isogeny $\varphi : E \to E'$.*

One of the best algorithms for solving the supersingular isogeny problem is the Delfs-Galbraith algorithm [11], which has expected time complexity $\tilde{O}(p^{1/2})$ in classical computation and $\tilde{O}(p^{1/4})$ in quantum computation with Grover's algorithm [12].

The following related problem is also important in isogeny-based cryptography.

Problem 2 (Maximal order problem). *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$, find a maximal order $\mathcal{O}$ in $B_{p,\infty}$ such that $\mathcal{O} \cong \mathrm{End}(E)$.*

Here, finding a maximal order in $B_{p,\infty}$ means that finding a basis of $\mathcal{O}$ as a tuple of $\mathbb{Q}$-vectors with respect to the basis $(1, \mathrm{i}, \mathrm{j}, \mathrm{ij})$.

It is known that Problem 1 and Problem 2 are equivalent, i.e., there exist polynomial-time reductions between them. It was showed by assuming the generalized Riemann hypothesis in [19], and later unconditionally in [16].

Another problem relevant to isogenies between supersingular elliptic curves is a problem to compute an isogeny from a given elliptic curve E and degree d.

Problem 3 (given-degree isogeny problem). *Given a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ and a degree d, find an isogeny φ from E of degree d.*

If d is smooth, then this problem can be solved efficiently by applying Vélu's formula [18] to each prime factor of d. Even if d is not smooth, if we are given a maximal order $\mathcal{O}$ in $B_{p,\infty}$ such that $\mathcal{O} \cong \mathrm{End}(E)$, then we can still compute an isogeny of degree d from E by using an ideal-to-isogeny algorithm. This is a core component of PRISM and will be explained in Section 3.1. On the other hand, if d is not smooth, and we are not given a maximal order $\mathcal{O}$, then there is no known efficient algorithm to solve this problem.

3. PRISM-id

In this section, we recall PRISM-id, an identification scheme in the PRISM framework [2].

3.1 Protocol

We give an overview of the identification scheme PRISM-id. We refer to the original paper [2] for further details.

First, we define the notation used in the scheme. Let p be a prime of the form $p = f2^e - 1$ for some integers f and e. We require $p \approx 2^e$ so that we can use `IdealToIsogeny` efficiently. Let a be an integer less than e, and let Primes_a be the set of all primes in $[2^{a-1}, 2^a]$.

PRISM-id is a two-round identification scheme, which is based on the following diagram.

$$\begin{tikzcd}
E_0 \arrow[d, "\varphi_{\mathrm{sk}}"'] & \\
E_{\mathrm{vk}} \arrow[r, "\varphi_{\mathrm{rsp}}"] & E_{\mathrm{rsp}}
\end{tikzcd}$$

Here, E_0 is the supersingular elliptic curve defined in Section 2.1. Note that $p \equiv 3 \pmod 4$ in PRISM. This is a system parameter for the scheme, and we assume that E_0 and the order $\mathcal{O}_0$ are implicitly given to the algorithms in

Table 1 PRISM-id identification scheme.

Prover($\mathrm{sk} = I_{\mathrm{sk}}$)		Verifier($\mathrm{vk} = E_{\mathrm{vk}}$)
		Sample a large prime $q \xleftarrow{\$} \mathsf{Primes}_a$
	$\xleftarrow{q}$	
$\varphi_{\mathrm{rsp}} \leftarrow \mathtt{GenIsogeny}(E_{\mathrm{vk}}, I_{\mathrm{sk}}, q)$		
	$\xrightarrow{\varphi_{\mathrm{rsp}}}$	
		Return $\mathtt{VerIsogeny}(E_{\mathrm{vk}}, \varphi_{\mathrm{rsp}}, q)$

the scheme.

To describe PRISM-id, we outline the key algorithms involved:

- $\mathtt{GenIsogeny}(E, I, q) \to \varphi$: On input a supersingular elliptic curve E over $\mathbb{F}_{p^2}$, a left $\mathcal{O}_0$-ideal I such that the codomain of the isogeny corresponding to I is E, and a prime $q \in \mathsf{Primes}_a$, outputs a $q(2^a - q)$-isogeny φ from E.
- $\mathtt{VerIsogeny}(E, \varphi, q) \to \mathtt{accept}/\mathtt{reject}$: On input a supersingular elliptic curve E over $\mathbb{F}_{p^2}$, an isogeny φ, and a prime $q \in \mathsf{Primes}_a$, outputs $\mathtt{accept}$ if the domain of φ is E and $\deg \varphi = q(2^a - q)$, and $\mathtt{reject}$ otherwise.

The procedure of the identification scheme is depicted in Table 1. Note that the isogeny φ_{sk} in the above diagram corresponds to the secret key ideal I_{sk}.

The details of the algorithms used in the identification scheme are as follows.

Key generation. Let λ be the security parameter. First, we prepare system parameters $e, f, a, E_0, \mathcal{O}_0$. The integers e and f are taken so that $p := f2^e - 1 \approx 2^{2\lambda}$ and f is as small as possible. Then a is chosen such that $2^a \approx \lambda + \log_2 \lambda$. The reason for these choices will be explained in Section 3.2. In addition to these choices, we fix a basis (P_0, Q_0) of $E_0[2^a]$ and add this to the system parameters.

Next, we sample a random ideal I_{sk} of norm ℓ^n, where ℓ is a prime larger than 2^a and n is an integer such that we get a distribution statistically close to uniform. This process can be done by using Algorithm 3 in [3]. Then, we compute the isogeny φ_{sk} corresponding to I_{sk} by $\mathtt{IdealToIsogeny}(I_{\mathrm{sk}})$, and let E_{vk} be the codomain of φ_{sk}. The secret key is I_{sk} and the public key is E_{vk}.

Isogeny generation ($\mathtt{GenIsogeny}(E_{\mathrm{vk}}, I_{\mathrm{sk}}, q)$). As in the key generation, we compute the isogeny φ_{sk} corresponding to I_{sk}. Next, we compute a random left $\mathcal{O}_0$-ideal I_{chl} of norm $q(2^a - q)$ as in the key generation. Then, we compute the isogeny φ_I corresponding to the ideal $I := I_{\mathrm{chl}} \cap I_{\mathrm{sk}}$ by $\mathtt{IdealToIsogeny}(I)$. Since $E_0[I_{\mathrm{sk}}] \subset E_0[I]$, the isogeny φ_I decomposes into $\varphi_{\mathrm{rsp}} \circ \varphi_{\mathrm{sk}}$, where φ_{rsp} is a $q(2^a - q)$-isogeny from E_{vk}. Let E_{rsp} be the codomain of φ_{rsp}. From φ_{sk} and $\varphi_{\mathrm{rsp}} \circ \varphi_{\mathrm{sk}}$, we obtain the isogeny $\varphi_{\mathrm{rsp}} : E_{\mathrm{vk}} \to E_{\mathrm{rsp}}$. We chose a basis $(P_{\mathrm{vk}}, Q_{\mathrm{vk}})$ of $E_{\mathrm{vk}}[2^a]$ independently of the images of P_0, Q_0 under φ_{sk}. Finally, we return $(E_{\mathrm{rsp}}, P_{\mathrm{vk}}, Q_{\mathrm{vk}}, \varphi_{\mathrm{rsp}}(P_{\mathrm{vk}}), \varphi_{\mathrm{rsp}}(Q_{\mathrm{vk}}))$ as an efficient representation of φ_{rsp}.

Verification ($\mathtt{VerIsogeny}(E_{\mathrm{vk}}, \varphi_{\mathrm{rsp}}, q)$). If the response φ_{rsp} is valid, then we can decompose φ_{rsp} as $\varphi_{\mathrm{rsp}} = \varphi_{2^a-q} \circ \varphi_q = \psi_q \circ \psi_{2^a-q}$, where $\deg \varphi_q = \deg \psi_q = q$ and $\deg \varphi_{2^a-q} = \deg \psi_{2^a-q} = 2^a - q$. Thus, we have the following commutative diagram.

$$\begin{tikzcd}
E_{\mathrm{vk}} \arrow[r, "\varphi_q"] \arrow[d, "\psi_{2^a-q}"'] \arrow[dr, "\varphi_{\mathrm{rsp}}"] & E_1 \arrow[d, "\varphi_{2^a-q}"] \\
E_2 \arrow[r, "\psi_q"'] & E_{\mathrm{rsp}}
\end{tikzcd}$$

By applying Kani's lemma (Lemma 1), we can obtain efficient representations of the isogenies in the diagram. The degrees of these isogenies can be checked by computing the 2^a-Weil pairings of the images of 2^a-torsion points under the isogenies. For more details, see §3.3 in [2].

3.2 Security

In this subsection, we briefly recall the security analysis of PRISM-id in [2].

First, to define the underlying problem of PRISM-id, we recall the definition of the oracle SPEDIO defined in [2].

Definition 2 (Definition 5 in [2]). *A special degree isogeny oracle (SPEDIO) is an oracle which takes as input a supersingular elliptic curve E and a prime $q \in \mathsf{Primes}_a$, and returns a uniformly random isogeny of degree $q(2^a - q)$ from E.*

Using this oracle, we define the underlying problem of PRISM-id.

Problem 4 (Problem 2 in [2]). *Given a supersingular elliptic curve E and SPEDIO, compute an isogeny of degree $q(2^a - q)$ from E with a prime $q \in \mathsf{Primes}_a$ different from all degrees formerly queried to the oracle.*

The security of PRISM-id is established based on this problem in [2] as follows.

Proposition 1 (Proposition 1 in [2]). *Under the assumption that Problem 4 is hard, the security against active attacks of PRISM-id performing N interactions has a winning probability bounded by $N/\#\mathsf{Primes}_a$.*

Next, we recall how to choose the parameters p and a to ensure a λ-bit security for classical attacks and a $\lambda/2$-bit security for quantum attacks (see §4.4 in [2] for details).

One of the best methods for key recovery attack on PRISM-id is to find an isogeny from E_0 to E_{vk} of smooth degree. As we mentioned in Section 2.5, its cost is $\tilde{O}(\sqrt{p})$ in classical settings and $\tilde{O}(p^{1/4})$ in quantum settings. Therefore, we require that p is at least approximately $2^{2\lambda}$.

In [2], they estimate the cost to compute a $q(2^a - q)$-isogeny from E_{vk} for $q \in \mathsf{Primes}_a$. They claim that its expected time complexity is in at least $\tilde{O}(q^2)$ in classical settings. Therefore, we require that a is at least approximately $\lambda/2$.

For the security of PRISM-id, we require that the probability in Proposition 4 is less than $2^{-\lambda}$. This leads to the requirement that $\#\mathsf{Primes}_a > 2^\lambda$. From the prime number theorem, we have

$$\#\mathsf{Primes}_a \approx \frac{2^a}{a\log 2} - \frac{2^{a-1}}{(a-1)\log 2} \approx \frac{2^a}{2a\log 2}.$$

Therefore, we require that $a > \lambda + \log_2 \lambda$.

4. Foundations of Our Attacks

In this section, we give algorithms and heuristics used in our attacks. In the rest of the paper, we use the same notation as in the previous section.

4.1 Algorithms

We introduce the following algorithms used in our attacks. From now on, the notation $\tilde{O}$ hides logarithmic factors in p as well as in its input.

Since our algorithms require operations in extension field $\mathbb{F}_{p^{2k}}$ for some exponentially large k, we need an algorithm to construct a computable representation of $\mathbb{F}_{p^{2k}}$. We denote this algorithm by $\mathtt{ConstructField}(k)$. This algorithm takes as input a positive integer k, and outputs an irreducible polynomial $F(x)$ of degree k over $\mathbb{F}_{p^2}$, which allow us to represent $\mathbb{F}_{p^{2k}}$ as $\mathbb{F}_{p^2}[x]/F(x)$. The cost of this algorithm is in $\tilde{O}(k^3)$. Roughly speaking, this comes from the cost to determine the irreducibility of a polynomial of degree k, which is in $\tilde{O}(k^2)$ (see Proposition 3.4.4 in [6]), times the expected number of polynomials that need to be checked for irreducibility, which is approximately k.

Next, we define an algorithm to compute an isogeny over $\mathbb{F}_{p^{2k}}$. Let E be a supersingular elliptic curve over $\mathbb{F}_{p^2}$, let q be a prime such that $E[q]$ is rational over $\mathbb{F}_{p^{2k}}$. Then we denote an algorithm to compute an isogeny from E of degree q by $\mathtt{Isogeny}(E, q, \mathbb{F}_{p^{2k}})$. The cost of this algorithm is in $\tilde{O}(\max\{k^2, k\sqrt{q}\})$. The first term k^2 comes from the cost of computing a point of order q in $E(\mathbb{F}_{p^{2k}})$. The second term $k\sqrt{q}$ comes from the cost of computing the isogeny from a generator of its kernel by using $\sqrt{\text{é}}$lu's formulas [5].

For a positive integer B less than 2^a, we define a subset $\mathsf{GoodPrimes}_{a,B}$ of Primes_a as

$$\{q \in \mathsf{Primes}_a \mid \exists k \in [1, B] \text{ s.t. } (-p)^k \equiv 1 \mod q\}.$$

In other words, $\mathsf{GoodPrimes}_{a,B}$ is the set of primes q in Primes_a such that $E[q]$ is rational over $\mathbb{F}_{p^{2k}}$ for some $k \in [1, B]$ and any supersingular elliptic curve E over $\mathbb{F}_{p^2}$.

For a given prime $q \in \mathsf{Primes}_a$ and a positive integer $B < 2^a$, we can determine whether q is in $\mathsf{GoodPrimes}_{a,B}$ with cost $\tilde{O}(\sqrt{B})$. This can be done by the method described in §3.2 in [5].

4.2 Heuristic Assumptions

We assume the following two heuristics related to the set $\mathsf{GoodPrimes}_{a,B}$.

Heuristic 1. *Let B be a positive integer less than 2^a. Then we assume that*

$$\#\mathsf{GoodPrimes}_{a,B} \approx \frac{B}{2a\log 2}.$$

Heuristic 2. *Let u be a positive real number in $[1, a]$ and B be a positive integer less than 2^a. Then we assume that for a prime q sampled uniformly at random from $\mathsf{GoodPrimes}_{a,B}$, the probability that $2^a - q$ is $2^{a/u}$-smooth depends only on u.*

Heuristic 1 follows from the prime number theorem and the assumption that $\Phi_k(-p)$ for $k \in [1, B]$ behave independently in $\mathbb{Z}/q\mathbb{Z}$ for $q \in \mathsf{Primes}_a$, where $\Phi_k(x)$ is the k-th cyclotomic polynomial.

Heuristic 2 follows from the assumption that the distribution of $2^a - q$ for $q \in \mathsf{GoodPrimes}_{a,B}$ is similar to the distribution of random integer less than 2^a and a is sufficiently large. Then, from the Dickman theorem (Theorem 1.4.9 in [8]), the probability that $2^a - q$ is $2^{a/u}$-smooth is $\rho(u)$, where $\rho(u)$ is the Dickman function.

5. Forgery Attacks on PRISM-id

In this section, we give two classical forgery attacks on PRISM-id [2] whose expected computational complexities are exponential in the security parameter, but slightly smaller than the claimed security level. Let λ be the security parameter of PRISM-id. This means that the claimed time complexity of PRISM-id is at least proportional to 2^λ in classical settings, ignoring a polynomial factor in λ. Since PRISM-id uses $a \approx \lambda + \log\lambda$, our purpose is to show that the expected time complexity of our attacks is in $\tilde{O}(2^{a\epsilon})$ for some $\epsilon < 1$.

5.1 Attack without Refusing Challenge Primes

We first consider a forgery attack in which the attacker does not refuse a challenge prime q. This attack only succeeds when the challenge prime q is a good prime. Such a good prime appears with exponentially small probability, but the probability is greater than $2^{-a} \approx 2^{-\lambda}$. This attack is described in Algorithm 1 and its success probability and expected time complexity are given in the following theorem.

Theorem 1. *Assume Heuristic 1 and Heuristic 2. For a positive number $\varepsilon < 1/3$, there exist a positive number C_ε depending only on ε and a forgery attack on PRISM-id with success probability $C_\varepsilon \cdot 2^{-a(1-\varepsilon)}$ and expected time complexity $\tilde{O}(\max\{2^{3a\varepsilon}, 2^{a(\varepsilon+1/2)}\})$.*

Sketch of Proof. The attack is depicted in Algorithm 1. From Heuristic 1 and Heuristic 2, we can expect that the number of primes $q \in \mathsf{Primes}_a$ which pass the check in line 3 is approximately $C_\varepsilon \cdot 2^{a\varepsilon}/(2a\log 2)$, where C_ε is a constant depending only on ε. Therefore, the success probability of this algorithm is approximately

$$\frac{C_\varepsilon \cdot 2^{a\varepsilon}}{2a\log 2} \cdot \frac{1}{\#\mathsf{Primes}_a} = C_\varepsilon \cdot 2^{-a(1-\varepsilon)}.$$

The dominant parts of the algorithm are $\mathtt{ConstructField}(k)$ in line 8, $\mathtt{Isogeny}(E, q, \mathbb{F}_{p^{2k}})$ in

Algorithm 1: Attack on PRISM-id

```
Input: A supersingular elliptic curve $E$, a prime $q$ in $\mathsf{Primes}_a$, and a positive integer $\varepsilon < 1/3$.
Output: An isogeny from $E$ of degree $q(2^a - q)$.
1  $B_1 \leftarrow \lceil 2^{a\varepsilon} \rceil$;
2  $B_2 \leftarrow \sqrt[3]{\max\{2^{3a\varepsilon}, 2^{a(\varepsilon+\frac{1}{2})}\}}$;
3  if IsGoodPrime($q, B_1$) = False or $2^a - q$ is not $B_2$-smooth then
4      return Failure;
5  $k \leftarrow 1$;
6  while $(-p)^k \not\equiv 1 \pmod{q}$ do
7      $k \leftarrow k + 1$;
8  $\mathbb{F}_{p^{2k}} \leftarrow$ ConstructField($k$);
9  $\varphi \leftarrow$ Isogeny($E, q, \mathbb{F}_{p^{2k}}$);
10 $F \leftarrow$ the codomain of $\varphi$;
11 Let $\ell_1^{e_1} \cdots \ell_n^{e_n}$ be the prime factorization of $2^a - q$;
12 for $i \leftarrow 1$ to $n$ do
13     $\mathbb{F}_{p^{2(\ell_i - 1)}} \leftarrow$ ConstructField($\ell_i - 1$);
14     for $j \leftarrow 1$ to $e_i$ do
15         $\psi \leftarrow$ Isogeny($F, \ell_i, \mathbb{F}_{p^{2(\ell_i - 1)}}$);
16         $F \leftarrow$ the codomain of $\psi$;
17         $\varphi \leftarrow \psi \circ \varphi$;
18 return $\varphi$;
```

line 14, and ConstructField($\ell_i - 1$) in line 13. These costs are in $\tilde{O}(B_1^3)$, $\tilde{O}(\max\{B_1^2, B_1\sqrt{q}\})$, and $\tilde{O}(B_2^3)$ respectively. These are all bounded by $\tilde{O}(\max\{2^{3a\varepsilon}, 2^{a(\varepsilon+1/2)}\})$. □

If we allow the attacker to interact with a proving oracle, we can mitigate the second condition that $2^a - q$ is B_2-smooth. Let $\mathtt{ProvingOracle}_E$ be a proving oracle for a public key E, i.e., $\mathtt{ProvingOracle}_E$ takes $q \in \mathsf{Primes}_a$ as input and returns an isogeny of degree $q(2^a - q)$ from E.

By using this oracle, we construct an algorithm to compute an isogeny of degree t from E, where t is a non-smooth factor of $2^a - q$. Let s be the largest B_2-smooth number dividing $2^a - q$, and let $t = (2^a - q)/s$. Suppose that there exists a small integer s' such that $2^a - s't$ is a prime in Primes_a different from q. We can expect that such an s' exists when $2^{a-1}/t > 2a \log 2$ since the probability that $2^a - s't$ is a prime is approximately $1/(2a \log 2)$ for a random $s' \in [1, 2^a/t]$. Then the attacker can obtain an isogeny of degree t by calling a proving oracle with the input $2^a - s't$. This is done as follows: First, the attacker obtains an isogeny of degree $(2^a - s't)s't$ from E by calling the oracle with the input $2^a - s't$. Then, the attacker computes an isogeny φ of degree $s't$ from E by using Kani's lemma. By evaluating a basis of $E[s']$ under φ, the attacker obtains generators of the kernel of degree s' part of $\hat{\varphi}$. This allows the attacker to compute an isogeny of degree t from E. We describe this algorithm in Algorithm 2. In the algorithm, we can expect that s is in $O(a)$ as explained above. Therefore, the time complexity of the algorithm is a polynomial in $\log p$.

We describe the attack on PRISM-id using a proving oracle in Algorithm 3.

Algorithm 2: IsogenyFromOracle($q, t, \mathsf{ProvingOracle}_E$)

```
Input: A prime $q$ in $\mathsf{Primes}_a$, a positive integer $t$, and a proving oracle $\mathtt{ProvingOracle}_E$.
Output: An isogeny from $E$ of degree $t$.
1  $s \leftarrow 1$;
2  while $st < 2^{a-1}$ and $2^a - st$ is not a prime in $\mathsf{Primes}_a \setminus \{q\}$ do
3      $s \leftarrow s + 2$;
4  if $st \geq 2^{a-1}$ then
5      return Failure;
6  $\varphi \leftarrow \mathsf{ProvingOracle}_E(2^a - st)$;
7  $\psi \leftarrow$ the $st$-isogeny from $E$ obtained from $\varphi$;    // Using Kani's lemma.
8  $k \leftarrow$ the smallest positive integer such that $(-p)^k \equiv 1 \pmod{s}$;    // $k < s$.
9  $\mathbb{F}_{p^{2k}} \leftarrow$ ConstructField($k$);
10 $(P, Q) \leftarrow$ a basis of $E[s]$;    // $P$ and $Q$ are in $E(\mathbb{F}_{p^{2k}})$.
11 $\psi_s \leftarrow$ an isogeny of degree $s$ with kernel $\langle \psi(P), \psi(Q) \rangle$;
       // $\psi = \hat{\psi_s} \circ \psi_t$ for a $t$-isogeny $\psi_t$.
12 $(P_a, Q_a) \leftarrow$ a basis of $E[2^a]$;
13 $\psi_t(P_a) \leftarrow [1/s]\psi_s \circ \psi(P_a)$, $\psi_t(Q_a) \leftarrow [1/s]\psi_s \circ \psi(Q_a)$;
14 return $\psi_t$;    // expressed by $(P_a, Q_a, \psi_t(P_a), \psi_t(Q_a))$.
```

Algorithm 3: Attack on PRISM-id with a proving oracle

```
Input: A supersingular elliptic curve $E$, a prime $q$ in $\mathsf{Primes}_a$, a positive integer $\varepsilon < 1/3$, and a proving oracle $\mathtt{ProvingOracle}_E$.
Output: An isogeny from $E$ of degree $q(2^a - q)$.
1  $B_1 \leftarrow \lceil 2^{a/3} \rceil$;
2  $B_2 \leftarrow \sqrt[3]{\max\{2^{3a\varepsilon}, 2^{a(\varepsilon+\frac{1}{2})}\}}$;
3  if IsGoodPrime($q, B_1$) = False then
4      return Failure;
5  Let $s$ be the largest $B_2$-smooth number dividing $2^a - q$;
6  $t \leftarrow (2^a - q)/s$;
7  if $t = 1$ then
8      Do the same as after line 5 in Algorithm 1;
9  else
10     $\varphi \leftarrow$ IsogenyFromOracle($q, t$, $\mathtt{ProvingOracle}_E$);
11     if $\varphi$ = Failure then
12         return Failure;
13     $F \leftarrow$ the codomain of $\varphi$;
14     $k \leftarrow 1$;
15     while $(-p)^k \not\equiv 1 \pmod{q}$ do
16         $k \leftarrow k + 1$;
17     $\mathbb{F}_{p^{2k}} \leftarrow$ ConstructField($k$);
18     $\psi \leftarrow$ Isogeny($F, q, \mathbb{F}_{p^{2k}}$);
19     $\varphi \leftarrow \psi \circ \varphi$;
20     $F \leftarrow$ the codomain of $\varphi$;
21     Let $\ell_1^{e_1} \cdots \ell_n^{e_n}$ be the prime factorization of $s$;
22     for $i \leftarrow 1$ to $n$ do
23         $\mathbb{F}_{p^{2(\ell_i - 1)}} \leftarrow$ ConstructField($\ell_i - 1$);
24         for $j \leftarrow 1$ to $e_i$ do
25             $\psi \leftarrow$ Isogeny($F, \ell_i, \mathbb{F}_{p^{2(\ell_i - 1)}}$);
26             $F \leftarrow$ the codomain of $\psi$;
27             $\varphi \leftarrow \psi \circ \varphi$;
28     return $\varphi$;
```

5.2 Attack with Refusing Challenge Primes

In this subsection, we consider an attack on PRISM-id in which the attacker can refuse a challenge prime q. Our attack has expected time complexity in $\tilde{O}(2^{6a/7})$. This attack is obtained by refusing challenge primes until the condition in line 3 in Algorithm 3 is satisfied. At the first glance, the expected time complexity of this attack seems to be greater than $2^{a(1-\varepsilon)} \cdot 2^{3a\varepsilon} = 2^{a(1+2\varepsilon)}$. However, the latter part of Algorithm 3 is executed only when the condition in line 3 is satisfied. Therefore, the expected time complexity of this attack is the sum of the expected computational complexities of obtaining a prime q passing the check in line 3 and the remaining part of the algorithm. We explicitly state this in the following theorem.

Theorem 2. *Assume Heuristic 1 and Heuristic 2. There exists a forgery attack on PRISM-id that allows the attacker to refuse a challenge prime q with expected time complexity in $\tilde{O}(2^{6a/7})$.*

Sketch of Proof. Consider Algorithm 1 with $\varepsilon = 2/7$ and modify it as follows:

- q is not given as input, but sampled uniformly at random from Primes_a.
- If the condition in line 3 is not satisfied, then the algorithm resamples q uniformly at random from Primes_a and checks the condition again.

Then the attack has expected time complexity in $\tilde{O}(2^{6a/7})$. □

5.3 Countermeasure

For PRISM-id to be secure against the forgery attacks in the previous subsections, the protocol should use a large enough a. Since the proposed p for PRISM-id in [2] is of the form $p = f2^e - 1$, where f is a small odd integer and $e \approx 2\lambda$, we can increase a as far as $a < 2\lambda$ while keeping $E[2^a]$ rational.

We propose two sizes of a for PRISM-id.

- **Conservative size:** $a = \frac{4}{3}\lambda$. This size ensures the following: In Algorithm 1 with any ε, the success probability is less than $2^{-\lambda}$ or the expected time complexity greater than approximately 2^λ.
- **Aggressive size:** $a = \frac{7}{6}\lambda$. This size ensures that the expected time complexity of the forgery attack in §5.2 is greater than approximately 2^λ.

Note that for the aggressive size, there exists a forgery attack whose success probability is greater than $2^{-\lambda}$ and whose expected time complexity is less than 2^λ. For example, by taking $\varepsilon = 3/14$ in Algorithm 1, the success probability is proportional to $2^{-11\lambda/12}$ and the expected time complexity is in $\tilde{O}(2^{5\lambda/6})$.

6. Attack on the Underlying Problem

In this section, we consider an attack on the underlying problem of PRISM-id, which is to find an isogeny of degree $q(2^a - q)$ for a prime $q \in \mathsf{Primes}_a$. The difference from the forgery attacks in Section 5 is that the attacker can choose the prime q as he likes. This allows the attacker to take $\varepsilon = 0$ in Algorithm 1, thus there exists an attack with expected time complexity in $\tilde{O}(2^{a/2})$. To describe the attack, we define the following problem, which is obtained by removing the oracle in Problem 4.

Algorithm 4: Attack on Problem 5

```
    Input: A supersingular elliptic curve $E$.
    Output: An isogeny from $E$ of degree $q(2^a - q)$ with a
            prime $q \in \mathsf{Primes}_a$.
 1  $k \leftarrow 1$;
 2  $q \leftarrow 0$;
 3  while $q = 0$ do
 4      $\Phi \leftarrow$ the $k$-th cyclotomic polynomial;
 5      if The prime factors of $\Phi(-p)$ contain a prime in
         $\mathsf{Primes}_a$ then
 6          $q \leftarrow$ the prime factor of $\Phi(-p)$ in $\mathsf{Primes}_a$;
 7          if $2^a - q$ is not $2^{a/6}$-smooth then
 8              $q \leftarrow 0$;
 9              $k \leftarrow k + 1$;
10      else
11          $k \leftarrow k + 1$;
12  Let $\ell_1^{e_1} \cdots \ell_n^{e_n}$ be the prime factorization of $2^a - q$;
13  $\mathbb{F}_{p^{2k}} \leftarrow$ ConstructField($k$);
14  $\varphi \leftarrow$ Isogeny($E, q, \mathbb{F}_{p^{2k}}$);
15  $F \leftarrow$ the codomain of $\varphi$;
16  for $i \leftarrow 1$ to $n$ do
17      $\mathbb{F}_{p^{2(\ell_i - 1)}} \leftarrow$ ConstructField($\ell_i - 1$);
18      for $j \leftarrow 1$ to $e_i$ do
19          $\psi \leftarrow$ Isogeny($F, \ell_i, \mathbb{F}_{p^{2(\ell_i - 1)}}$);
20          $F \leftarrow$ the codomain of $\psi$;
21          $\varphi \leftarrow \psi \circ \varphi$;
22  return $\varphi$;
```

Problem 5. *Given a supersingular elliptic curve E, compute an isogeny of degree $q(2^a - q)$ from E with a prime $q \in \mathsf{Primes}_a$.*

Obviously, the underlying problem of PRISM-id, that is Problem 4, reduce to Problem 5. The following theorem shows our attack on Problem 5.

Theorem 3. *Assuming Heuristic 1 and Heuristic 2, there exists an algorithm to solve Problem 5 with time complexity $\tilde{O}(2^{a/2})$.*

Sketch of Proof. Our attack is described in Algorithm 4. From Heuristic 1 and Heuristic 2, the variable k after the **while** loop is in $O(a)$. Therefore, the dominant parts of the algorithm are $\mathtt{Isogeny}(\mathsf{E}, \mathsf{q}, \mathbb{F}_{\mathsf{p}^{2k}})$ in line 14 and $\mathtt{ConstructField}(\ell_i - 1)$ in line 17. These costs are in $\tilde{O}(2^{a/2})$. □

As in the attack on PRISM-id, we can mitigate the condition that $2^a - q$ is $2^{a/6}$-smooth by using SPEDIO. Unlike $\mathtt{ProvingOracle}_E$, we can call SPEDIO with input a supersingular elliptic curve different from E. In other words, we have $\mathtt{ProvingOracle}_{E'}$ for all supersingular elliptic curves E'. By using SPEDIO, the attacker could obtain an isogeny of degree t from E even in the case that Algorithm 2 fails.

This is done as follows: Let t be a non-smooth factor of $2^a - q$. Suppose that t decomposes into a product $t_1 \cdots t_n$ such that there exist small integers $s_1, \ldots, s_n$ satisfying $2^a - s_i t_i \in \mathsf{Primes}_a \setminus \{q\}$ for all $i = 1, \ldots, n$. Then the attacker can obtain a t-isogeny from E as follows:

```
(1) Let $E_1$ be $E$.
(2) For $i = 1, \ldots, n$,
  (a) $\varphi_i \leftarrow \texttt{IsogenyFromOracle}(E_i, t_i, \texttt{ProvingOracle}_{E_i})$
  (b) Let $E_{i+1}$ be the codomain of $\varphi_i$.
(3) Output $\varphi_n \circ \cdots \circ \varphi_2 \circ \varphi_1$.
```

7. Conclusion

We proposed classical forgery attacks on PRISM-id in scenarios where refusing a challenge is allowed and where it is not. In both cases, we showed that the original parameter in PRISM-id does not achieve the claimed security level. We also proposed a countermeasure against our attacks. Additionally, we analyzed the underlying problem of PRISM-id and gave an explicit time complexity for solving it. We believe that our work provides valuable insights into the security of PRISM-id and is useful for constructing new protocols based on the hardness of the given-degree isogeny problem.

Acknowledgments The authors thank Shingo Sato for valuable discussion on attacks against identification schemes. This work was supported by JST K Program Grant Number JPMJKP24U2, Japan.

References

[1] Aardal, M. A., Adj, G., Aranha, D. F., Basso, A., Canales Martínez, I. A., Chávez-Saab, J., Santos, M. C., Dartois, P., De Feo, L., Duparc, M., Eriksen, J. K., Fouotsa, T. B., Filho, D. L. G., Hess, B., Kohel, D., Leroux, A., Longa, P., Maino, L., Meyer, M., Nakagawa, K., Onuki, H., Panny, L., Patranabis, S., Petit, C., Pope, G., Reijnders, K., Robert, D., Rodríguez Henríquez, F., Schaeffler, S. and Wesolowski, B.: SQIsign, Technical report, National Institute of Standards and Technology (2024). available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures.

[2] Basso, A., Borin, G., Castryck, W., Corte-Real Santos, M., Invernizzi, R., Leroux, A., Maino, L., Vercauteren, F. and Wesolowski, B.: PRISM: Simple and Compact Identification and Signatures from Large Prime Degree Isogenies, *Public-Key Cryptography – PKC 2025* (Jager, T. and Pan, J., eds.), Cham, Springer Nature Switzerland, pp. 300–332 (2025).

[3] Basso, A., Dartois, P., De Feo, L., Leroux, A., Maino, L., Pope, G., Robert, D. and Wesolowski, B.: SQIsign2D-West - The Fast, the Small, and the Safer, *ASIACRYPT 2024, Part III* (Chung, K.-M. and Sasaki, Y., eds.), LNCS, Vol. 15486, Kolkata, India, Springer, Singapore, Singapore, pp. 339–370 (online), https://doi.org/10.1007/978-981-96-0891-1_11 (2024).

[4] Basso, A. and Maino, L.: POKÉ: A Compact and Efficient PKE from Higher-Dimensional Isogenies, *EUROCRYPT 2025, Part II* (Fehr, S. and Fouque, P.-A., eds.), LNCS, Vol. 15602, Madrid, Spain, Springer, Cham, Switzerland, pp. 94–123 (online), https://doi.org/10.1007/978-3-031-91124-8_4 (2025).

[5] Bernstein, D. J., De Feo, L., Leroux, A. and Smith, B.: Faster computation of isogenies of large prime degree, *ANTS-XIV - 14th Algorithmic Number Theory Symposium* (Galbraith, S., ed.), Proceedings of the Fourteenth Algorithmic Number Theory Symposium (ANTS-XIV), Vol. 4, Auckland, New Zealand, Mathematical Sciences Publishers, pp. 39–55 (online), https://doi.org/10.2140/obs.2020.4.39 (2020).

[6] Cohen, H.: *A Course in Computational Algebraic Number Theory*, Graduate Texts in Mathematics, Vol. 138, Springer Berlin, Heidelberg (2010).

[7] Cosset, R. and Robert, D.: Computing (l, l)-isogenies in polynomial time on Jacobians of genus 2 curves, *Mathematics of Computation*, Vol. 84, No. 294, pp. 1953–1975 (2015).

[8] Crandall, R. and Pomerance, C. B.: *Prime Numbers: A Computational Perspective*, Springer New York, second edition (2005).

[9] Dartois, P., Leroux, A., Robert, D. and Wesolowski, B.: SQIsignHD: New Dimensions in Cryptography, *EUROCRYPT 2024, Part I* (Joye, M. and Leander, G., eds.), LNCS, Vol. 14651, Zurich, Switzerland, Springer, Cham, Switzerland, pp. 3–32 (online), https://doi.org/10.1007/978-3-031-58716-0_1 (2024).

[10] Dartois, P., Maino, L., Pope, G. and Robert, D.: An Algorithmic Approach to (2, 2)-Isogenies in the Theta Model and Applications to Isogeny-Based Cryptography, *ASIACRYPT 2024, Part III* (Chung, K.-M. and Sasaki, Y., eds.), LNCS, Vol. 15486, Kolkata, India, Springer, Singapore, Singapore, pp. 304–338 (online), https://doi.org/10.1007/978-981-96-0891-1_10 (2024).

[11] Delfs, C. and Galbraith, S. D.: Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$, *DCC*, Vol. 78, No. 2, pp. 425–440 (online), https://doi.org/10.1007/s10623-014-0010-1 (2016).

[12] Grover, L. K.: A Fast Quantum Mechanical Algorithm for Database Search, *28th ACM STOC*, Philadephia, PA, USA, ACM Press, pp. 212–219 (online), https://doi.org/10.1145/237814.237866 (1996).

[13] Kani, E.: The number of curves of genus two with elliptic differentials, *Journal für die reine und angewandte Mathematik*, Vol. 485, pp. 93–122 (online), https://doi.org/doi:10.1515/crll.1997.485.93 (1997).

[14] Lubicz, D. and Robert, D.: Fast change of level and applications to isogenies, *Research in Number Theory*, Vol. 9, No. 1, p. 7 (online), https://doi.org/10.1007/s40993-022-00407-9 (2023).

[15] Maino, L., Martindale, C., Panny, L., Pope, G. and Wesolowski, B.: A Direct Key Recovery Attack on SIDH, *EUROCRYPT 2023, Part V* (Hazay, C. and Stam, M., eds.), LNCS, Vol. 14008, Lyon, France, Springer, Cham, Switzerland, pp. 448–471 (online), https://doi.org/10.1007/978-3-031-30589-4_16 (2023).

[16] Merdy, A. H. L. and Wesolowski, B.: Unconditional foundations for supersingular isogeny-based cryptography (2025).

[17] Robert, D.: Breaking SIDH in Polynomial Time, *EUROCRYPT 2023, Part V* (Hazay, C. and Stam, M., eds.), LNCS, Vol. 14008, Lyon, France, Springer, Cham, Switzerland, pp. 472–503 (online), https://doi.org/10.1007/978-3-031-30589-4_17 (2023).

[18] Vélu, J.: Isogénies entre courbes elliptiques, *Comptes-Rendues de l'Académie des Sciences*, Vol. 273, pp. 238–241 (1971).

[19] Wesolowski, B.: The supersingular isogeny path and endomorphism ring problems are equivalent, *62nd FOCS*, Denver, CO, USA, IEEE Computer Society Press, pp. 1100–1111 (online), https://doi.org/10.1109/FOCS52979.2021.00109 (2022).