Adaptive attacks on FESTA variants with masked-degree isogenies

Tomoki Moriya

Information Technology R&D Center, Mitsubishi Electric Corporation, Japan
Moriya.Tomoki@bp.MitsubishiElectric.co.jp

Abstract. FESTA is an isogeny-based trapdoor function proposed as a high-performance trapdoor function in isogeny-based cryptography. Its core design principles have inspired a number of related constructions, collectively referred to as *FESTA variants*.
The MOXZ attack is an adaptive attack that exploits malicious ciphertexts together with access to a checking oracle, aiming to compromise FESTA and its variants. This attack applies to FESTA variants whose secret keys are derived from isogenies of known degree; however, it does not extend to variants employing masked-degree isogenies.
In this work, we present a novel adaptive attack that generalizes the MOXZ attack. Our attack successfully targets several FESTA variants even when their secret keys are isogenies of masked degree. We also identify POKÉ-4D as an exception for which our attack does not appear to be applicable.

1 Introduction

Isogeny-based cryptography is considered a promising candidate for post-quantum cryptography, as it is believed to be resistant to attacks by quantum computers. Indeed, an isogeny-based digital signature scheme SQISign [6,1] is selected as one of the candidates in the NIST Post-Quantum Cryptography Standardization Process for additional digital signatures [22].

One of the main advantages of isogeny-based cryptography is its ability to achieve relatively compact key and ciphertext sizes. In particular, SQISign is the most compact signature scheme among the candidates submitted to the NIST Post-Quantum Cryptography Standardization Process.

On the other hand, two major challenges remain for isogeny-based cryptography: the lack of highly efficient implementations and the relatively limited body of cryptanalytic research.

SIDH [10] was a promising isogeny-based key exchange scheme, which realized a compact and relatively efficient implementation. However, several studies showed that SIDH can be broken in polynomial-time [5,15,23].

FESTA [3] is one of the proposals aimed at achieving SIDH-like efficiency in isogeny-based cryptography. The central idea behind FESTA is to mask public-key information using secret scalars. While this technique prevents the SIDH attacks, it preserves much of the underlying mathematical structure of

SIDH, thereby enabling FESTA to retain a comparable level of compactness and efficiency. The core idea of FESTA has subsequently inspired a number of isogeny-based cryptosystems, which we call FESTA variants, *e.g.*, QFESTA [21], POKÉ [2] and so on.

The GPST attack [8] is an adaptive attack against SIDH that recovers a static secret key by combining malicious messages with access to a key-exchange oracle in time polynomial of the security parameter. The MOXZ attack [18] follows a similar adaptive strategy, but targets FESTA[1] rather than SIDH. Moreover, it can be extended to several FESTA variants in which the degree of the secret isogeny is publicly known. However, the attack does not apply to variants whose secret isogeny has a hidden degree, such as POKÉ. An adaptive attack against POKÉ was proposed in [13]; however, its goal is to recover the sender's static key rather than the receiver's secret key.

To the best of our knowledge, no polynomial-time adaptive attack is currently known for secret keys of FESTA variants that involve masked-degree isogenies. This naturally leads to the following research question:

> *Is there a polynomial-time adaptive attack against FESTA variants with masked-degree isogenies?*

Contribution. In this work, we provide a partial answer to this question. Specifically, we present a novel polynomial-time adaptive attack against several FESTA variants based on masked-degree isogenies.

We prove that the attack is effective against four schemes: POKÉ-2D [2], PIKE [4], INKE [12], and MD-LIT-SiGamal [19], none of which are vulnerable to the MOXZ attack. On the other hand, we discuss our approach does not appear to extend to POKÉ-4D [2]. A summary of our results is provided in Figure 1.

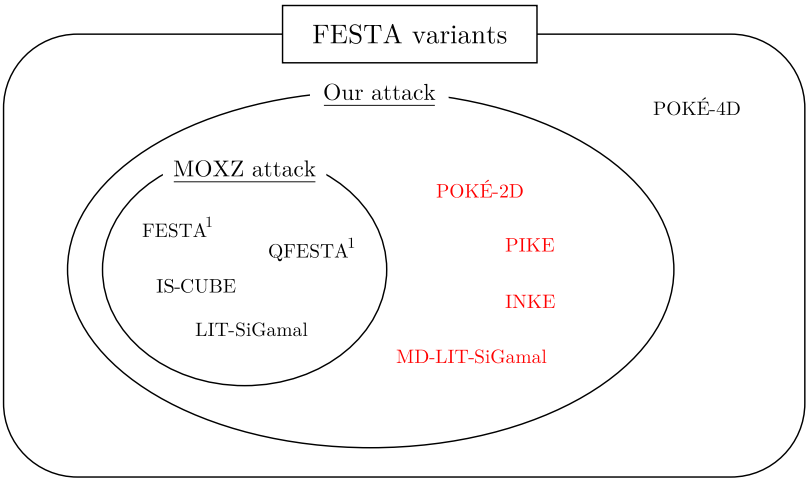
Remark 1. Our attack is complementary to the attack presented in [13] (the Lim–Ti attack). Specifically, our attack reveals the receiver's secret keys for several POKÉ variants, whereas the Lim–Ti attack reveals their senders' static keys. Therefore, some POKÉ variants, particularly POKÉ-2D, INKE, and PIKE, are vulnerable to adaptive attacks, and their secret keys can be fully exposed without the Fujisaki–Okamoto transform.

2 Preliminaries

In this section, we review the mathematical preliminaries underlying isogeny-based cryptography.

[1] More precisely, this attack targets FESTA instances that either lack a verification procedure or are not implemented in constant time. The original FESTA (and QFESTA) proposals are not vulnerable to the MOXZ attack.

Fig. 1. Adaptive attacks agaist FESTA variants



2.1 Elliptic curves and isogenies

We begin by reviewing elliptic curves and isogenies. For a comprehensive treatment of these topics, see [24].

Let p be a prime, and let k be a finite field of characteristic p. An *elliptic curve* over k is a smooth projective algebraic curve equipped with a group law compatible with its algebraic structure. More generally, a complete, smooth, and geometrically irreducible algebraic variety endowed with a compatible group structure is called an *abelian variety* [20].

For an elliptic curve E and a positive integer n, the n-torsion subgroup of E is defined by

$$E[n] := \{P \in E(\overline{k}) \mid nP = 0\}.$$

If $\gcd(n, p) = 1$, then

$$E[n] \cong (\mathbb{Z}/n\mathbb{Z})^2.$$

Moreover, the group $E[p]$ is isomorphic either to $\mathbb{Z}/p\mathbb{Z}$ or to the trivial group $\{0\}$. An elliptic curve E is called *ordinary* if $E[p] \cong \mathbb{Z}/p\mathbb{Z}$, and *supersingular* if $E[p] \cong \{0\}$.

An *isogeny* over k is a surjective morphism of abelian varieties defined over k with finite kernel.

Let E_0 and E_1 be elliptic curves. An isogeny $\phi\colon E_0 \to E_1$ induces a field homomorphism

$$\phi^*\colon k(E_1) \to k(E_0), \qquad f \mapsto f \circ \phi,$$

where $k(E_0)$ and $k(E_1)$ denote the function fields of E_0 and E_1, respectively.

The isogeny ϕ is said to be *separable* if the field extension

$$k(E_0)/\phi^*(k(E_1))$$

is separable. The degree of ϕ, denoted by $\deg \phi$, is defined as the extension degree

$$[k(E_0) : \phi^*(k(E_1))].$$

For an isogeny $\phi\colon E_0 \to E_1$, there exists an isogeny $\hat{\phi}\colon E_1 \to E_0$ such that

$$\hat{\phi} \circ \phi = [\deg \phi] \quad \text{and} \quad \phi \circ \hat{\phi} = [\deg \phi].$$

The isogeny $\hat{\phi}$ is called the *dual isogeny* of ϕ.

If an isogeny ϕ is separable, then $\deg \phi = \#\ker \phi$. Furthermore, for any finite subgroup $G \subset E_0$, there exists a separable isogeny $\phi\colon E_0 \to E_1$ with kernel G, unique up to isomorphism. Given E_0 and G, both ϕ and the codomain curve E_1 can be computed explicitly using Vélu's formulas [26].

2.2 Kani's lemma and the SIDH attacks

Kani's lemma derives an isogeny between high-dimensional abelian varieties from a commutative diagram of isogenies. It is a fundamental tool in attacks on SIDH and has also been employed in several isogeny-based cryptographic constructions.

We consider the following commutative diagram of elliptic curves and isogenies:

$$\begin{array}{ccc} E_0 & \xrightarrow{\phi} & E_1 \\ {\scriptstyle\psi}\downarrow & & \downarrow{\scriptstyle\psi'} \\ E_0' & \xrightarrow[\phi']{} & E_1' \end{array}$$

If it holds that

$$\gcd(\deg \phi, \deg \psi) = 1, \quad \deg \phi = \deg \phi' \quad \text{and} \quad \deg \psi = \deg \psi',$$

then we refer to this diagram as an *SIDH diagram*.

Proposition 2 (Kani's lemma [11] and [23, Lemma 3.6]). *We assume that there is an SIDH diagram given by:*

$$\begin{array}{ccc} E_0 & \xrightarrow{\phi} & E_1 \\ {\scriptstyle\psi}\downarrow & & \downarrow{\scriptstyle\psi'} \\ E_0' & \xrightarrow[\phi']{} & E_1' \end{array}$$

Then, there is a d-isogeny[2] $\Phi\colon E_0' \times E_1 \to E_0 \times E_1'$ *given by*

$$\Phi = \begin{pmatrix} \hat{\psi} & \hat{\phi} \\ -\phi' & \psi' \end{pmatrix},$$

[2] See [23, Definition 3.1] for the definition of a d-isogeny among polarized abelian varieties of dimension g. Note that the kernel of a d-isogeny has order d^g.

where $d = \deg\phi + \deg\psi$. *The kernel of the isogeny* Φ *is given by*

$$\ker\Phi = \langle\, (\psi(P), \phi(P)) \mid P \in E_0'[d] \,\rangle.$$

Kani's lemma extends to commutative diagrams of principally polarized abelian varieties [23]. In particular, it applies to commutative diagrams whose vertices are products of elliptic curves equipped with the product polarization.

Assume that d is coprime to p. Then, using algorithms for computing isogenies of higher-dimensional abelian varieties (see, for example, [14]), we can recover Φ from its kernel and hence compute the isogeny ϕ. Therefore, Kani's lemma provides an alternative method for evaluating ϕ using d-torsion points.

Definition 3 (The SSI-T problem). *Let* p *be a prime, let* d *be a positive integer. Let* $\mathcal{B}$ *be a positive integer, and let* A *be a positive integer such that*

$$A \mid (p+1), \quad A^2 > d, \quad \gcd(A, d) = 1,$$

and every prime factors of A *is less than* $\mathcal{B}$ *(i.e.,* A *is* $\mathcal{B}$*-smooth). Let* E_0 *and* E_1 *be supersingular elliptic curves defined over* $\mathbb{F}_{p^2}$*, and let* $\{P_0, Q_0\}$ *and* $\{P_1, Q_1\}$ *be bases of* $E_0[A]$ *and* $E_1[A]$ *respectively.*

The supersingular isogeny with torsion problem (the SSI-T problem) is formulated as the following computational problem:

Given

$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1)),\ d,$$

find an isogeny $\phi\colon E_0 \to E_1$ *such that*

$$\deg\phi = d, \quad \textit{and} \quad \begin{pmatrix} P_1 \\ Q_1 \end{pmatrix} = \begin{pmatrix} \phi(P_0) \\ \phi(Q_0) \end{pmatrix}.$$

The SSI-T problem, is closely related to the security of SIDH. In particular, if the problem can be solved when $A \approx d$, then SIDH can be broken.

Kani's lemma yields the following proposition, which in turn gives rise to an algorithm for solving the SSI-T problem and hence a practical attack on SIDH. We refer the reader to [23] for a proof of the proposition.

Proposition 4 (SIDH attacks). *There exists a probabilistic polynomial-time algorithm that solves the SSI-T problem.*

3 FESTA variants

FESTA is an isogeny-based trapdoor function introduced in [3]. Its core design has inspired a variety of isogeny-based cryptographic constructions (see, *e.g.*, [21,16,2,19,12,4]).

In this section, we present an abstract framework that captures FESTA and its variants, and we review the MOXZ attack [18], which may concern the security of these schemes.

3.1 Structure of FESTA variants

From the SIDH attack, one observes that a hidden isogeny $\phi\colon E_0 \to E_1$ can be recovered, provided that a suitable basis of a torsion subgroup of E_0, together with its image under ϕ, is given. However, such torsion-point data also plays a crucial role in constructing isogeny-based cryptosystems, as in the SIDH key exchange.

A central idea underlying FESTA is to conceal torsion-point information by means of a suitable matrix, typically chosen to be diagonal. We now provide a more detailed description. Let E_0, E_1 be supersingular elliptic curves, let $\phi\colon E_0 \to E_1$ be an isogeny, let A be a positive integer coprime to $\deg\phi$ satisfying $\deg\phi < A^2$, and let $\{P_0, Q_0\}$ be a basis of $E_0[A]$. We define points $P_1, Q_1 \in E_1[A]$ by

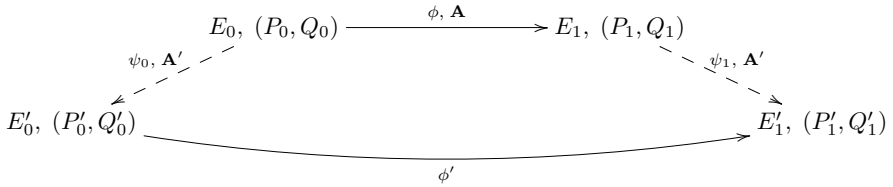
$$\begin{pmatrix} P_1 \\ Q_1 \end{pmatrix} = \mathbf{A} \begin{pmatrix} \phi(P_0) \\ \phi(Q_0) \end{pmatrix},$$

where $\mathbf{A} \in \mathrm{GL}_2(\mathbb{Z}/A\mathbb{Z})$ is a diagonal matrix. The security of the construction of FESTA relies on the assumption that recovering the isogeny ϕ from the tuple

$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1))$$

is computationally hard.

We now consider the following (non-commutative) diagram, which can represent the trapdoor of FESTA.



Here, we let E'_0 and E'_1 be supersingular elliptic curves, and let $\psi_0\colon E_0 \to E'_0$ and $\psi_1\colon E_1 \to E'_1$ be isogenies. Let $\mathbf{A}' \in \mathrm{GL}_2(\mathbb{Z}/A\mathbb{Z})$ be a diagonal matrix, and define points $P'_0, Q'_0 \in E'_0[A]$ and $P'_1, Q'_1 \in E'_1[A]$ by

$$\begin{pmatrix} P'_0 \\ Q'_0 \end{pmatrix} = \mathbf{A}' \begin{pmatrix} \psi_0(P_0) \\ \psi_0(Q_0) \end{pmatrix}, \qquad \begin{pmatrix} P'_1 \\ Q'_1 \end{pmatrix} = \mathbf{A}' \begin{pmatrix} \psi_1(P_1) \\ \psi_1(Q_1) \end{pmatrix}.$$

We define $\phi' := \psi_1 \circ \phi \circ \hat{\psi}_0$.

Since $\mathbf{A}$ and $\mathbf{A}'$ are diagonal matrices, they commute, i.e., $\mathbf{A}\mathbf{A}' = \mathbf{A}'\mathbf{A}$. Therefore, it follows from the direct computation that

$$\begin{aligned}
\mathbf{A}\begin{pmatrix}\phi'(P_0')\\ \phi'(Q_0')\end{pmatrix} &= \mathbf{A}\begin{pmatrix}(\psi_1\circ\phi\circ\hat{\psi}_0)(P_0')\\ (\psi_1\circ\phi\circ\hat{\psi}_0)(Q_0')\end{pmatrix}\\
&= \deg\psi_0\cdot\mathbf{A}\cdot\mathbf{A}'\begin{pmatrix}(\psi_1\circ\phi)(P_0)\\ (\psi_1\circ\phi)(Q_0)\end{pmatrix}\\
&= \deg\psi_0\cdot\mathbf{A}'\cdot\mathbf{A}\begin{pmatrix}(\psi_1\circ\phi)(P_0)\\ (\psi_1\circ\phi)(Q_0)\end{pmatrix}\\
&= \deg\psi_0\cdot\mathbf{A}'\begin{pmatrix}\psi_1(P_1)\\ \psi_1(Q_1)\end{pmatrix}\\
&= \deg\psi_0\cdot\begin{pmatrix}P_1'\\ Q_1'\end{pmatrix}.
\end{aligned}$$

Consequently, knowledge of the matrix $\mathbf{A}$ allows one to recover the isogeny ϕ' via the SIDH attacks. In this manner, the above construction naturally yields a trapdoor, thereby realizing FESTA and its variants.

The following provides an explicit description of an abstract variant of FESTA, which may be instantiated either as a public-key encryption scheme or as a key encapsulation mechanism.

Key Generation: Let A be a $\mathcal{B}$-smooth positive integer, and let p be a prime such that $A \mid (p+1)$. Let E_0 be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$, and let $\{P_0, Q_0\}$ be a basis of $E_0[A]$. Let $\mathcal{M}_A \subset \mathrm{GL}_2(\mathbb{Z}/A\mathbb{Z})$ denote an abelian subgroup consisting of diagonal matrices, and let $\mathbf{A} \in \mathcal{M}_A$.
Let $\phi\colon E_0 \to E_1$ be a separable isogeny satisfying $\gcd(\deg\phi, A) = 1$ and $\deg\phi < A^2$, and define

$$\begin{pmatrix}P_1\\ Q_1\end{pmatrix} := \mathbf{A}\begin{pmatrix}\phi(P_0)\\ \phi(Q_0)\end{pmatrix}.$$

We define the public key $\mathbf{pk}$ and the secret key $\mathbf{sk}$ as follows:

$$\begin{aligned}
\mathbf{pk} &:= (E_0, (P,Q), E_1, (P_1,Q_1), \mathcal{M}_A, \mathbf{pk}^*),\\
\mathbf{sk} &:= (\mathbf{A}, \mathbf{sk}^*),
\end{aligned}$$

where $\mathbf{pk}^*$ and $\mathbf{sk}^*$ denote sets of additional data that may vary depending on the underlying scheme.

Encapsulation / Encryption: Let $\mathbf{A}' \in \mathcal{M}_A$ be an ephemeral key. Let E_0' and E_1' be supersingular elliptic curves over $\mathbb{F}_{p^2}$, and let

$$\begin{aligned}
\psi_0&\colon E_0 \longrightarrow E_0',\\
\psi_1&\colon E_1 \longrightarrow E_1'
\end{aligned}$$

be isogenies, which are assumed to be computed in polynomial time from the public key $\mathbf{pk}$. Define points $P_0', Q_0' \in E_0'[A]$ and $P_1', Q_1' \in E_1'[A]$ by

$$\begin{pmatrix}P_0'\\ Q_0'\end{pmatrix} = \mathbf{A}'\begin{pmatrix}\psi_0(P_0)\\ \psi_0(Q_0)\end{pmatrix}, \qquad \begin{pmatrix}P_1'\\ Q_1'\end{pmatrix} = \mathbf{A}'\begin{pmatrix}\psi_1(P_1)\\ \psi_1(Q_1)\end{pmatrix}.$$

We then define the ciphertext $\mathbf{ct}$ by

$$\mathbf{ct} := \big(E'_0, (P'_0, Q'_0),\, E'_1, (P'_1, Q'_1),\, \mathbf{ct}^*\big),$$

where $\mathbf{ct}^*$ denotes auxiliary data determined by the underlying scheme and the plaintext.

Decapsulation / Decryption: Using $\mathbf{A} \in \mathbf{sk}$, we compute

$$\begin{pmatrix} P''_0 \\ Q''_0 \end{pmatrix} = \mathbf{A} \begin{pmatrix} P'_0 \\ Q'_0 \end{pmatrix}.$$

From the data

$$(E'_0, (P''_0, Q''_0)),\ (E'_1, (P'_1, Q'_1)),\ \deg \phi,$$

one can recover the isogeny ϕ' via the SIDH attack, where $\deg \phi$ is included in either $\mathbf{pk}^*$ or $\mathbf{sk}^*$.

Finally, the isogeny ϕ' together with the auxiliary data $\mathbf{ct}^*$ enables the derivation of the shared key or the recovery of the plaintext.

3.2 Hard problems associated with FESTA variants

In this subsection, we introduce several mathematical problems that are closely related to the security of FESTA and its variants. In particular, we present the main computational problem that constitutes the primary focus of our study.

Throughout this paper, we employ the following terminologies:

- λ: the security parameter,
- $\mathcal{B}$: a positive integer with $\mathcal{B} = \mathrm{poly}(\lambda)$,
- t: a positive integer with $t = O(\log \lambda)$,
- A: a $\mathcal{B}$-smooth positive integer where the number of prime factors of A is bounded by t,
- p: a prime number with $A \mid (p+1)$ and $\log p = O(\lambda)$,
- E_0, E_1: supersingular elliptic curves over $\mathbb{F}_{p^2}$,
- (P_0, Q_0): an ordered basis of $E_0[A]$,
- (P_1, Q_1): an ordered basis of $E_1[A]$,
- $\mathcal{M}_A$: a subgroup of $\mathrm{GL}_2(\mathbb{Z}/A\mathbb{Z})$ consisting of diagonal matrices,
- d: an integer with $\gcd(d, A) = 1$ and $d < A^2$,
- Δ: a set of integers.

We first introduce the CIST problem, as defined in the original FESTA paper [3]; in particular, this problem is closely related to the security of the FESTA family of schemes.

Definition 5 (The CIST problem [3]). *The Computational isogeny with scaled-torsion (CIST) problem is formulated as the following computational problem:*

Given

$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1)),\ \mathcal{M}_A,\ d,$$

find an isogeny $\phi\colon E_0 \to E_1$ *such that*

$$\deg \phi = d, \quad \textit{and} \quad \begin{pmatrix} P_1 \\ Q_1 \end{pmatrix} = \mathbf{A} \begin{pmatrix} \phi(P_0) \\ \phi(Q_0) \end{pmatrix},$$

for some $\mathbf{A} \in \mathcal{M}_A$.

In the original FESTA setting, the degree of ϕ is fixed and hence publicly known. However, in certain variants of FESTA, most notably POKÉ, the value $\deg \phi$ is kept secret. In such settings, the problem of recovering the isogeny becomes more challenging.

Motivated by this observation, we introduce a variant of the CIST problem in which the value $\deg \phi$ is not provided as part of the input.

Definition 6 (The MD-CIST problem). *The masked-degree CIST problem (the MD-CIST problem) is formulated as the following computational problem:*

Given

$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1)),\ \mathcal{M}_A,\ \Delta,$$

find an isogeny $\phi\colon E_0 \to E_1$ *such that*

$$\deg \phi \in \Delta, \quad \textit{and} \quad \begin{pmatrix} P_1 \\ Q_1 \end{pmatrix} = \mathbf{A} \begin{pmatrix} \phi(P_0) \\ \phi(Q_0) \end{pmatrix},$$

for some $\mathbf{A} \in \mathcal{M}_A$.

Finally, we formalize our target problems, namely the (MD-)CIST problem equipped with a checking oracle O_{check}. Informally, a checking oracle is an oracle that, given a ciphertext, determines whether it can be successfully decapsulated (decrypted) or not. We also note that the checking oracle is related to the oracle in Model 2 of [8, Section 3].

There are several potential directions for attacking FESTA variants that exploit a checking oracle. In this work, we focus on an approach in which the adversary generates incorrect torsion-point data, namely P_0', Q_0', P_1', and Q_1'. More precisely, the adversary computes these points as

$$\begin{pmatrix} P_0' \\ Q_0' \end{pmatrix} = \mathbf{B}_0 \begin{pmatrix} \psi_0(P_0) \\ \psi_0(Q_0) \end{pmatrix}, \qquad \begin{pmatrix} P_1' \\ Q_1' \end{pmatrix} = \mathbf{B}_1 \begin{pmatrix} \psi_1(P_1) \\ \psi_1(Q_1) \end{pmatrix},$$

where $\mathbf{B}_0$ and $\mathbf{B}_1$ are matrices in $\mathrm{GL}_2(\mathbb{Z}/A\mathbb{Z})$, not necessarily elements of $\mathcal{M}_A$, and in general $\mathbf{B}_0 \neq \mathbf{B}_1$. Then, the adversary generates an incorrectly formed ciphertext

$$\mathbf{ct}' := (E_0', (P_0', Q_0'), E_1', (P_1', Q_1'), \mathbf{ct}^*),$$

and observes the output of $O_{check}(\mathbf{ct}')$.

In this setting, the incorrectly formed ciphertext can be successfully decapsulated (or decrypted) if the relation

$$\mathbf{A}\mathbf{B}_0 = \mathbf{B}_1\mathbf{A}$$

holds. Therefore, if this relation holds, then $O_{check}(\mathbf{ct}')$ outputs $\mathsf{success}$. If the relation does not hold, without a negligible probability, the receiver cannot obtain ϕ' from the same discussion in [18, Section 4.1]. This implies that the incorrectly formed ciphertext is failed to be decapsulated (or decrypted) in any FESTA variants, and thus, $O_{check}(\mathbf{ct}')$ outputs $\mathsf{failure}$ in this case. Consequently, it suffices to consider the following oracle in place of the checking oracle:

$$O(\mathbf{B}_0, \mathbf{B}_1) = \begin{cases} 0 & (\text{if } \mathbf{A}\mathbf{B}_0 = \mathbf{B}_1\mathbf{A}) \\ 1 & (\text{if } \mathbf{A}\mathbf{B}_0 \neq \mathbf{B}_1\mathbf{A}) \end{cases}. \tag{1}$$

We now formalize our target computational problems, namely the OA-CIST problem and the OA-MD-CIST problem.

Definition 7 (The OA-CIST problem). *The oracle-assisted CIST problem (the OA-CIST problem) is formulated as the following computational problem:*

Given

$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1)),\ \mathcal{M}_A,\ d,\ O,$$

find an isogeny $\phi\colon E_0 \to E_1$ *such that*

$$\deg\phi = d, \quad \textit{and} \quad \begin{pmatrix} P_1 \\ Q_1 \end{pmatrix} = \mathbf{A}\begin{pmatrix} \phi(P_0) \\ \phi(Q_0) \end{pmatrix},$$

for some $\mathbf{A} \in \mathcal{M}_A$.

Definition 8 (The OA-MD-CIST problem). *The oracle-assisted masked-degree CIST problem (the OA-MD-CIST problem) is formulated as the following computational problem:*

Given

$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1)),\ \mathcal{M}_A,\ \Delta,\ O,$$

find an isogeny $\phi\colon E_0 \to E_1$ *such that*

$$\deg\phi \in \Delta, \quad \textit{and} \quad \begin{pmatrix} P_1 \\ Q_1 \end{pmatrix} = \mathbf{A}\begin{pmatrix} \phi(P_0) \\ \phi(Q_0) \end{pmatrix},$$

for some $\mathbf{A} \in \mathcal{M}_A$.

As discussed in the next subsection, the OA-CIST problem can be solved in polynomial time by the MOXZ attack [18]. However, solving the OA-MD-CIST problem appears to be significantly more challenging, and constitutes the main objective of this study.

3.3 MOXZ attack

The MOXZ attack is an adaptive attack on FESTA proposed in [18]. Although it was originally introduced for FESTA under specific settings, it can be extended to several variants of FESTA. In particular, the MOXZ attack yields a polynomial-time algorithm for solving the OA-CIST problem if A is a power of 2 (Proposition 9).

Proposition 9 (MOXZ attack [18]). *The OA-CIST problem can be solved in polynomial-time if A is a power of 2.*

We now present a proof of Proposition 9, following the approach of [18].

Proof. Let the secret matrix $\mathbf{A}$ be

$$\mathbf{A} := \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix}.$$

Recall that the SIDH attacks reduce the OA-CIST problem to the recovery of the secret matrix $\mathbf{A}$. Hence, our goal is to recover the parameters α and β.

To this end, we first introduce a new oracle $O_{\mathrm{coeff}} : (\mathbb{Z}/A\mathbb{Z})^2 \to \{0, 1\}$ defined by

$$O_{\mathrm{coeff}}(\varepsilon_0, \varepsilon_1) := O\left(\mathbf{A}' + \begin{pmatrix} 0 & 0 \\ \varepsilon_1 & 0 \end{pmatrix}, \mathbf{A}' + \begin{pmatrix} 0 & 0 \\ \varepsilon_0 & 0 \end{pmatrix}\right),$$

where $\mathbf{A}' \in \mathcal{M}_A$.

A direct computation yields

$$\mathbf{A}\left(\mathbf{A}' + \begin{pmatrix} 0 & 0 \\ \varepsilon_1 & 0 \end{pmatrix}\right) - \left(\mathbf{A}' + \begin{pmatrix} 0 & 0 \\ \varepsilon_0 & 0 \end{pmatrix}\right)\mathbf{A} = \begin{pmatrix} 0 & 0 \\ \beta\varepsilon_1 - \alpha\varepsilon_0 & 0 \end{pmatrix}.$$

Consequently, the oracle O_{coeff} satisfies

$$O_{\mathrm{coeff}}(\varepsilon_0, \varepsilon_1) = \begin{cases} 0 & (\text{if } \alpha\varepsilon_0 = \beta\varepsilon_1) \\ 1 & (\text{otherwise}) \end{cases}. \tag{2}$$

Therefore, the oracle O_{coeff} can determine whether $\alpha/\beta = \varepsilon_1/\varepsilon_0$ or not.

We emphasize that, in the MOXZ setting, the degree of the hidden isogeny ϕ is known. By exploiting the Weil pairing e_A, we have

$$e_A(P_1, Q_1) = e_A(P_0, Q_0)^{\alpha\beta \deg \phi}.$$

Therefore, by solving the Discrete Logarithm problem between $e_A(P_0, Q_0)$ and $e_A(P_1, Q_1)$, one can compute the product $\alpha\beta \in (\mathbb{Z}/A\mathbb{Z})^\times$. Therefore, by replacing (P_1, Q_1) with $\frac{1}{\sqrt{\alpha\beta}}(P_1, Q_1)$, we may assume without loss of generality that the matrix $\mathbf{A}$ is of the form

$$\mathbf{A} = \frac{1}{\sqrt{\alpha\beta}} \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} = \begin{pmatrix} \gamma & 0 \\ 0 & \gamma^{-1} \end{pmatrix},$$

where $\gamma = \sqrt{\alpha/\beta}$.

Let $A = 2^a$, and let $\gamma_0, \ldots, \gamma_{a-1}$ be elements in $\{0, 1\}$ such that

$$\gamma = \sum_{i=0}^{a-1} \gamma_i 2^i.$$

We also define $\gamma^{(j)}$ for $j = 0, \ldots, a-1$ by

$$\gamma^{(j)} := \sum_{i=0}^{j} \gamma_i 2^i.$$

For $j = 2, \ldots, a$, it holds that

$$\gamma + \gamma^{(j-1)} \equiv 2 \pmod 4.$$

By combining this congruence with

$$\gamma - \gamma^{(j-1)} \equiv \gamma_j 2^j \pmod{2^{j+1}},$$

we have, for $j = 2, \ldots, a-2$,

$$2^{a-j-2}\gamma^2 - 2^{a-j-2}\left(\gamma^{(j-1)}\right)^2 = \gamma_j 2^{a-1}.$$

Hence, the oracle O_{coeff} recovers the coefficient γ_j from $\gamma^{(j-1)}$ via

$$\gamma_j = O_{\mathrm{coeff}}\left(2^{a-j-2},\, 2^{a-j-2}\left(\gamma^{(j-1)}\right)^2\right).$$

Since $\gamma \in (\mathbb{Z}/A\mathbb{Z})^\times$, it follows that $\gamma_0 = 1$. Moreover, upon replacing γ with $-\gamma$ if necessary, we may assume without loss of generality that $\gamma_1 = 0$. Consequently, an inductive procedure on j reveals the coefficients $\gamma_0, \ldots, \gamma_{a-2}$. Finally, the remaining coefficient γ_{a-1} can be recovered by a brute-force search. □

Algorithm 1 summarizes the above procedure to obtain the secret scalar γ.

Remark 10. Although the original paper does not explicitly discuss the applicability of the MOXZ attack beyond the case where A is a power of 2, the attack naturally extends to the more general setting.

By the Chinese remainder theorem, it suffices to treat the case where A is a power of an odd prime. Therefore, we let ℓ be an odd prime and $e \geq 1$ an integer, and set $A = \ell^e$.

Write γ in base ℓ as

$$\gamma = \sum_{i=0}^{e-1} \gamma_i \ell^i,$$

where each $\gamma_i \in \{0, \ldots, \ell - 1\}$. For $0 \leq j \leq e-1$, define the truncated sum

$$\gamma^{(j)} := \sum_{i=0}^{j} \gamma_i \ell^i.$$

Algorithm 1 The MOXZ attack [18]

Require: $(E_0, (P_0, Q_0))$, $(E_1, (P_1, Q_1))$, $\mathcal{M}_{2^a}$, $\deg\phi$, and O_{coeff}
Ensure: $\gamma \in (\mathbb{Z}/2^a\mathbb{Z})^\times$
$\gamma^{(1)} \leftarrow 1$
for $j = 2, \ldots, a-2$ **do**
  $\gamma_j \leftarrow O_{coeff}(2^{a-j-2}, 2^{a-j-2}(\gamma^{(j-1)})^2)$
  $\gamma^{(j)} \leftarrow \gamma^{(j-1)} + \gamma_j 2^j$
end for
for $\gamma_{a-1} = 0, 1$ **do**
  $\gamma \leftarrow \gamma^{(a-2)} + \gamma_{a-1}2^{a-1}$
  $\mathbf{A} \leftarrow \begin{pmatrix} \gamma & 0 \\ 0 & \gamma^{-1} \end{pmatrix}$
  $\begin{pmatrix} P_1' \\ Q_1' \end{pmatrix} \leftarrow \mathbf{A}^{-1} \begin{pmatrix} P_1 \\ Q_1 \end{pmatrix}$
  if $\exists$ an isogeny $\phi\colon E_0 \to E_1$ such that $P_1' = \phi(P_0)$ and $Q_1' = \phi(Q_0)$ **then**
    return γ
  end if
end for
return $\perp$

Let $c \in \{0, \ldots, \ell-1\}$, and choose $\delta \in \{0, \ldots, \ell-1\}$ such that

$$2\gamma_0 \equiv \delta \pmod{\ell}.$$

A direct computation shows that for $j = 1, \ldots, e-1$,

$$\begin{aligned} &\ell^{e-j-1} \cdot \gamma^2 - \left(\ell^{e-j-1}(\gamma^{(j-1)})^2 + c\,\delta\,\ell^{e-1}\right) \\ &= \ell^{e-1}(\gamma + \gamma^{(j-1)})\gamma_j - c\,\delta\,\ell^{e-1} \\ &\equiv \ell^{e-1}\delta(\gamma_j - c) \pmod{\ell^e}. \end{aligned}$$

Consequently,

$$O_{\text{coeff}}\left(\ell^{e-j-1},\, \ell^{e-j-1}(\gamma^{(j-1)})^2 + c\,\delta\,\ell^{e-1}\right) = \begin{cases} 0 & (\text{if } \gamma_j = c) \\ 1 & (\text{otherwise}) \end{cases}.$$

Therefore, by testing all $c \in \{0, \ldots, \ell-1\}$, an adversary can recover γ_j from O_{coeff} together with $\gamma^{(j-1)}$. Hence, an adversary can compute γ by induction in polynomial-time for any $\mathcal{B}$-smooth integer A.

4 Our attack: solving the OA-MD-CIST problem

In this section, we present a new adaptive attack on variants of FESTA. In particular, our attack applies to several schemes whose security relies on the MD-CIST problem.

4.1 Proposed algorithm for the OA-MD-CIST problem

Our attack provides a polynomial-time algorithm to solve the OA-MD-CIST problem in specific cases.

We first introduce a mathematical problem, called the MD-SSI-T problem, whose hardness is closely related to our approach for solving the OA-MD-CIST problem.

Definition 11 (The MD-SSI-T problem). *The masked-degree supersingular isogeny with torsion problem (the MD-SSI-T problem) is formulated as the following computational problem:*

Given

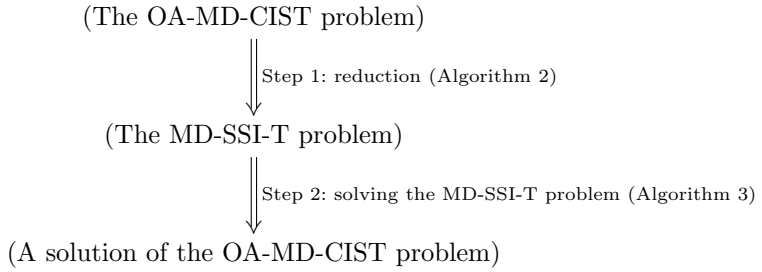
$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1)),\ \Delta,$$

find an isogeny $\phi\colon E_0 \to E_1$ *such that*

$$\deg \phi \in \Delta, \quad \textit{and} \quad \begin{pmatrix} P_1 \\ Q_1 \end{pmatrix} = \alpha \begin{pmatrix} \phi(P_0) \\ \phi(Q_0) \end{pmatrix},$$

for some $\alpha \in (\mathbb{Z}/A\mathbb{Z})^\times$.

Our attack does not solve the OA-MD-CIST problem directly. Instead, we decompose the attack into two steps. The first step, based on the MOXZ attack, reduces the OA-MD-CIST problem to the MD-SSI-T problem. The second step solves the MD-SSI-T problem in certain cases.



We call the first step the generalized MOXZ attack. From Equation (2), the oracle O_{coeff} provides information on $\gamma' := \alpha/\beta$, and thus the adversary can compute γ' by using O_{coeff} throughout this algorithm. If we have γ', then we can construct the tuple

$$(E_0, (P_0, Q_0)),\ (E_1, (\alpha\phi(P_0), \alpha\phi(Q_0))).$$

Consequently, this algorithm provides a reduction from the OA-MD-CIST problem to the MD-SSI-T problem.

The second step solves the MD-SSI-T problem in certain cases. This approach originates from previous studies showing that POKÉ is vulnerable when $\deg\phi$ takes one of the following forms:

$$\deg\phi = \begin{cases} q(A-q) & \text{(see [2, Section 5.2])}, \\ A^2 - q^2 & \text{(see [17])}, \end{cases}$$

where q is an integer.

Combining these two steps, we ultimately obtain an algorithm for solving the OA-MD-CIST problem in certain cases.

Remark 12. The above cases are relatively uncommon in the mathematical setting. Nevertheless, several isogeny-based protocols, *e.g.,* POKÉ-2D, PIKE, INKE, and LIT-SiGamal, are specifically designed for such cases, enabling efficient computations.

Step 1: the generalized MOXZ attack. Our approach is derived from a generalization of the MOXZ attack. A key distinction between our setting and that of MOXZ lies in the use of the degree of the target isogeny. Specifically, our attack does not require knowledge of $\deg\phi$, whereas the MOXZ attack assumes that this quantity is known.

In the original setting of the MOXZ attack, we assume that $\alpha\beta = 1$ for the target matrix $\mathbf{A}$. Under this assumption, knowledge of $\deg\phi$ is required in order to recover $\mathbf{A}$ via the MOXZ attack. In contrast, in the setting of the OA-MD-CIST problem, we cannot impose the condition $\alpha\beta = 1$, since the secret quantity $\deg\phi$ is unknown. Consequently, the MOXZ attack does not directly yield the matrix $\mathbf{A}$ itself.

We show that the MOXZ attack indeed yields a reduction from the OA-MD-CIST problem to the MD-SSI-T problem. More precisely, it recovers a quantity $\gamma' \in \mathbb{Z}/A\mathbb{Z}$ satisfying

$$\gamma' = \alpha/\beta,$$

thereby establishing a reduction between the two problems.

Theorem 13. *The OA-MD-CIST problem can be reduced to the MD-SSI-T problem via Algorithm 2 in time $O(\mathcal{B}\log A)$ using $O(\mathcal{B}\log A)$ queries to the oracle O.*

Proof. By the Chinese remainder theorem, it suffices to assume that A is a prime power. We therefore write $A = \ell^e$, where ℓ is a prime integer. We let $\gamma'_0, \ldots, \gamma'_{e-1}$ be elements in $\{0, \ldots, \ell-1\}$ satisfying

$$\gamma' = \sum_{i=0}^{e-1} \gamma'_i \ell^i,$$

and, for each integer $k = 0, \ldots, e-1$, define the truncated expansion

$$\gamma'^{(k)} = \sum_{i=0}^{k} \gamma'_i \ell^i.$$

Recall Equation (2), which implies that

$$O_{\mathrm{coeff}}(\varepsilon_0, \varepsilon_1) = \begin{cases} 0 & (\text{if } \gamma'\varepsilon_0 = \varepsilon_1) \\ 1 & (\text{otherwise}) \end{cases}.$$

A direct computation then shows that, for $j = 0, \ldots, e-1$,

$$O_{\mathrm{coeff}}\big(\ell^{e-j-1},\ \ell^{e-j-1}\gamma'^{(j-1)} + c\,\ell^{e-1}\big) = \begin{cases} 0 & (\text{if } \gamma'_j = c) \\ 1 & (\text{otherwise}) \end{cases},$$

where $c = 0, \ldots, \ell-1$, and we define $\gamma'^{(-1)} := 0$.

Hence, by induction, we can recover γ' in time $O(e\ell)$ using $e\ell$ queries to the oracle O. We summarize the algorithm in Algorithm 2.

Write

$$A = \prod_{i=1}^{t} \ell_i^{e_i},$$

where $\ell_1, \ldots, \ell_t$ are distinct primes less than $\mathcal{B}$. Then, from the above discussion, the generalized MOXZ attack has time complexity

$$O\left(\sum_{i=1}^{t} e_i \ell_i\right) = O(\mathcal{B} \log A),$$

and requires $O(\mathcal{B} \log A)$ queries to the oracle O. This completes the proof of the assertion. □

Algorithm 2 The generalized MOXZ attack

Require: O_{coeff}
Ensure: $\gamma' \in (\mathbb{Z}/\ell^e\mathbb{Z})^\times$

```
γ'^(-1) ← 0
for j = 0, ..., e − 1 do
    for c = 0, ..., ℓ − 1 do
        i ← O_coeff(ℓ^{e−j−1}, ℓ^{e−j−1} γ'^(j−1) + c ℓ^{e−1})
        if i = 0 then
            γ'^(j) ← γ'^(j−1) + c · ℓ^j
        end if
    end for
end for
return γ'^(e−1)
```

Step 2: solving the MD-SSI-T problem. We next explain how to solve the MD-SSI-T problem. That is, we compute the secret isogeny ϕ from the tuple

$$(E_0, (P_0, Q_0)),\ (E_1, (P_1, Q_1)),\ \Delta,$$

where $P_1 := \alpha\phi(P_0)$, $Q_1 := \alpha\phi(Q_0)$, and $\deg\phi \in \Delta$.

When $\#\Delta = 1$, the MD-SSI-T problem reduces to the SSI-T problem, since the value $\alpha \in \mathbb{Z}/A\mathbb{Z}$ can be recovered by solving the Discrete Logarithm problem of $e_A(P_0, Q_0)$ and $e_A(P_1, Q_1) = e_A(P_0, Q_0)^{\alpha^2 \deg\phi}$. Consequently, the SIDH attacks also yields a solution to the MD-SSI-T problem. Furthermore, when $\#\Delta = 1$, the OA-MD-CIST problem coincides with the OA-CIST problem. Therefore, our attack applies to all FESTA variants for which the MOXZ attack is effective.

We now turn to the case where $\#\Delta$ is sufficiently large, namely, the setting in which the degree of the secret isogeny is genuinely masked.

It is known that the MD-SSI-T problem admits a polynomial-time solution in certain special cases even if $\#\Delta \gg 1$. For example, in the POKÉ-2D setting, such an algorithm is presented in [2, Section 6.1]. In this work, we generalize this approach and extend it to a broader class of settings.

Definition 14 (Computation-friendly representation). *Let d and A be positive integers where A is $\mathcal{B}$-smooth, and let $\phi\colon E_0 \to E_1$ be an isogeny with $\deg\phi = d$.*

We say that ϕ admits a computation-friendly representation with respect to A if there exist positive integers d_1, d_2, and m such that

$$\gcd(d_1, d_2) = 1, \quad mA = d_1 + d_2, \quad d_1 d_2 = d,$$

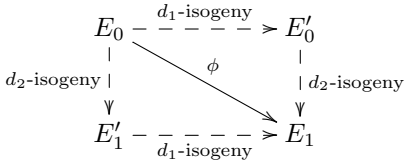
and $m = \mathrm{poly}(\lambda)$.

Suppose that an isogeny $\phi\colon E_0 \to E_1$ admits a computation-friendly representation with respect to A. Then ϕ can be evaluated via a 2-dimensional isogeny

$$\Phi\colon E_0 \times E_1 \longrightarrow V, \qquad \ker\Phi = \langle (d_1 P, \phi(P)) \mid P \in E_0[A] \rangle,$$

where V is an abelian variety of dimension 2.

Indeed, there exist elliptic curves E_0' and E_1' fitting into the following commutative diagram.



Thus, by Kani's lemma, there exists an mA-isogeny

$$\Psi\colon E_0 \times E_1 \longrightarrow E_0' \times E_1'$$

that is the composition of Φ and an m-isogeny. Consequently, the isogeny ϕ can be evaluated by computing Φ together with all m-isogenies from the abelian variety V.

According to [14], the complexity of computing Φ is

$$O(\mathcal{B}^2 \cdot \log A \cdot \mathrm{poly}(\log p)),$$

while the complexity of computing an m-isogeny is

$$O(m^2 \cdot \mathrm{poly}(\log p)).$$

Moreover, the number of m-isogenies from V is approximately m^3. Therefore, the overall complexity of computing Ψ is

$$O\left((\mathcal{B}^2 \cdot \log A + m^5) \cdot \mathrm{poly}(\log p)\right).$$

Remark 15. In our setting, we only consider the cases $m = 1$ and $m = 2$. Thus, computing an m-isogeny is computationally feasible.

We now show that the MD-SSI-T problem can be solved if the solution ϕ admits a computation-friendly representation.

Theorem 16. *The MD-SSI-T problem can be solved in time polynomial in λ if its solution admits a computation-friendly representation with respect to A.*

Proof. From the above discussion, if we can compute

$$G := \langle (d_1 P, \phi(P)) \mid P \in E_0[A] \rangle,$$

then we can evaluate the target isogeny ϕ.

Using the Weil pairing e_A and solving the Discrete Logarithm problem, we can compute $\alpha^2 d \pmod{A}$, since

$$e_A(P_1, Q_1) = e_A(P_0, Q_0)^{\alpha^2 d}.$$

This computation can be done in time polynomial in λ because A is a $\mathcal{B}$-smooth integer. Recalling that $d_2 \equiv -d_1 \pmod{A}$, we obtain

$$\alpha^2 d = \alpha^2 d_1 d_2 \equiv -(\alpha d_1)^2 \pmod{A}.$$

Hence, $\alpha d_1 \pmod{A}$ can be recovered. We note that there are approximately 2^t candidates for αd_1.

If we collect the correct αd_1, then we can compute G, because

$$G = \langle (\alpha d_1 P_0, P_1), (\alpha d_1 Q_0, Q_1) \rangle.$$

Therefore, given G and m, we can compute the isogeny ϕ. If G is not the correct one, the computation fails. In this case, we consider another candidate for αd_1 and repeat the computation until we obtain the desired isogeny ϕ. Therefore, the overall complexity of this procedure is

$$O\left(2^t \cdot (\mathcal{B}^2 \cdot \log A + m^5) \cdot \mathrm{poly}(\log p)\right) = \mathrm{poly}(\lambda).$$

This completes the proof of Theorem 16. □

We summarize our algorithm to solve the MD-SSI-T problem in Algorithm 3.

Remark 17. Theorem 16 can be viewed as a generalization of [2, Section 5.2] and [17, Theorem 7]. It unifies the settings considered in these works and provides a unified approach to solving the MD-SSI-T problem.

Algorithm 3 Soving the MD-SSI-T problem (in specific cases)

Require: $(E_0, (P_0, Q_0))$, $(E_1, (P_1, Q_1))$, and m
Ensure: $\phi\colon E_0 \to E_1$ such that $P_1 = \alpha\phi(P_0)$ and $Q_1 = \alpha\phi(Q_0)$
1: $\mu_0 \leftarrow e_A(P_0, Q_0)$
2: $\mu_1 \leftarrow e_A(P_1, Q_1)$
3: Compute $c \in \mathbb{Z}/A\mathbb{Z}$ satisfying $\mu_1 = \mu_0^c$
4: **for all** c' satisfying $c'^2 = c$ **do**
5: $G \leftarrow \langle (c'P_0, P_1), (c'Q_0, Q_1) \rangle$
6: Compute ϕ from G and m
7: **if** ϕ is the desired isogeny **then**
8: **return** ϕ
9: **end if**
10: **end for**
11: **return** $\perp$

4.2 Attack on actual FESTA variants

We now discuss the impact of our attack on several FESTA variants to which the MOXZ attack does not apply. In particular, we focus on POKÉ-2D, INKE, PIKE, LIT-SiGamal, and POKÉ-4D.

We conclude that all of these schemes, except for POKÉ-4D, are vulnerable to our adaptive attack. Therefore, these schemes should be combined with an appropriate transform, such as the Fujisaki–Okamoto transform [7], in order to achieve resistance against adaptive attacks.

POKÉ-2D, INKE, and PIKE. We first discuss the cases of POKÉ-2D [2], INKE [12], and PIKE [4].

Although these schemes have different structures, the core components underlying the derivation of the OA-MD-CIST problem are essentially identical. In particular, they all employ the same set Δ, defined by

$$\Delta := \{q(A-q) \mid q \in \{1, \ldots, A-1\},\ \gcd(q, A) = 1\}.$$

The target isogeny ϕ admits a computation-friendly representation with respect to A by setting

$$d_1 := q, \quad d_2 := A - q.$$

Therefore, our adaptive attack applies to these schemes.

(MD-)LIT-SiGamal. In the setting of LIT-SiGamal [19], the degree of the target isogeny is not hidden, and therefore the MOXZ attack applies directly to LIT-SiGamal. However, [19, Remark 2] also proposes a variant of LIT-SiGamal in which $\deg\phi$ is hidden, rendering the MOXZ attack inapplicable. In what follows, we focus on this variant, which we refer to as MD-LIT-SiGamal.

In the setting of MD-LIT-SiGamal, the target isogeny ϕ is chosen so that its degree belongs to

$$\Delta := \{A^2 - q^2 \mid q \in \{1, \ldots, A-1\},\ \gcd(q, A) = 1\}.$$

We focus on the integers $A+q$ and $A-q$. A direct computation shows that

$$\gcd(A+q, A-q) = \begin{cases} 2 & \text{if } A \text{ and } q \text{ are odd} \\ 1 & \text{otherwise} \end{cases}.$$

Suppose first that $\gcd(A+q, A-q) = 1$. In this case, ϕ admits a computation-friendly representation with respect to A, obtained by setting

$$d_1 := A+q, \quad d_2 := A-q.$$

Therefore, MD-LIT-SiGamal is vulnerable to our attack if $\gcd(A+q, A-q) = 1$, which includes the original LIT-SiGamal setting.

Next, suppose that $\gcd(A+q, A-q) = 2$. Let ϕ' be the isogeny satisfying

$$\phi = \phi' \circ \phi_4,$$

where ϕ_4 is an isogeny of degree 4. Then ϕ' admits a computation-friendly representation with respect to A by taking

$$d_1 := \frac{A+q}{2}, \quad d_2 := \frac{A-q}{2}.$$

Moreover, the isogeny ϕ_4 can be recovered by an exhaustive search among every 4-isogenies mapping from E_0. Hence, MD-LIT-SiGamal remains vulnerable to our attack in this case as well.

POKÉ-4D. POKÉ-4D appears in [2, Section 4.4], where four-dimensional isogenies are employed for ciphertext decryption.

In the POKÉ-4D setting, the set Δ is defined as

$$\Delta := \{A^2 - q_1^2 - q_2^2 \mid q_1, q_2 \in \{1, \ldots, A-1\},\ \gcd(q_1^2 + q_2^2, A) = 1\}.$$

We first show that the isogeny ϕ rarely admits a computation-friendly representation with respect to A. Suppose that ϕ admits such a representation. Then there exist integers d_1, d_2, and m satisfying

$$A^2 - q_1^2 - q_2^2 = d_1 d_2, \quad d_1 + d_2 = mA.$$

Hence,

$$d_1^2 - mAd_1 + (A^2 - q_1^2 - q_2^2) = 0.$$

It follows that d_1 is rational if and only if

$$m^2A^2 - 4(A^2 - q_1^2 - q_2^2)$$

is a perfect square. Heuristically, for $A^2 - q_1^2 - q_2^2 \in \Delta$, the probability of this event is $O(1/A)$. Therefore, one does not expect the isogeny ϕ to admit a computation-friendly representation with respect to A.

Consequently, POKÉ-4D does not fall within the framework in which our adaptive attack applies. However, this does not rule out the possibility of an adaptive attack against POKÉ-4D, as there may exist fundamentally different approaches for constructing such an attack.

We now discuss why designing an adaptive attack on POKÉ-4D could be difficult. More precisely, we explain the approach in the proof of Theorem 16 may not work for solving the MD-SSI-T problem with the above choice of Δ.

In the decryption procedure of POKÉ-4D, we consider the following commutative diagram, which yields a 4-dimensional isogeny via Kani's lemma:

$$\begin{array}{ccc} E_0 \times E_0 & \xrightarrow{\phi I_2} & E_1 \times E_1 \\ {\scriptstyle M_q}\downarrow & & \downarrow{\scriptstyle M_q} \\ E_0 \times E_0 & \xrightarrow[\phi I_2]{} & E_1 \times E_1. \end{array}$$

Here, M_q is the matrix

$$M_q := \begin{pmatrix} q_1 & -q_2 \\ q_2 & q_1 \end{pmatrix}.$$

More precisely, recovering the secret isogeny ϕ reduces to determining the subgroup

$$G := \left\langle \big((P,Q)\cdot M_q,\ (\phi(P),\phi(Q))\big) \;\middle|\; P,Q \in E_0[A] \right\rangle \subset E_0^2 \times E_1^2.$$

Therefore, one can solve the MD-SSI-T problem by computing G from the given instance of the problem.

If one can compute $\alpha M_q \pmod A$, then the subgroup G can be reconstructed. Consequently, the MD-SSI-T problem can be solved given

$$\alpha q_1 \pmod A \quad \text{and} \quad \alpha q_2 \pmod A.$$

However, obtaining these two values appears to be difficult for the following reason.

Using the Weil pairing e_A, one can compute

$$\alpha^2(q_1^2 + q_2^2) \pmod A.$$

Hence, one may attempt to recover $\alpha q_1 \pmod A$ and $\alpha q_2 \pmod A$ by solving the equation

$$x_1^2 + x_2^2 = \alpha^2(q_1^2 + q_2^2) \pmod A.$$

However, this equation admits approximately A solutions (see [25]), making it difficult to identify the correct pair

$$(\alpha q_1, \alpha q_2) \pmod A.$$

In summary, constructing an adaptive attack against POKÉ-4D within our framework appears to be difficult. This observation could provide evidence that POKÉ-4D is more secure than POKÉ-2D.

Remark 18. Snake Mackerel is an isogeny-based authenticated key encapsulation mechanism proposed in [9]. It incorporates POKÉ-4D as a core component due to its compactness. The security of Snake Mackerel relies on the hardness of the OA-CIST problem; consequently, POKÉ-2D cannot be substituted for POKÉ-4D within the scheme.

Table 1 summarizes Δ, (d_1, d_2), m, and the number of queries required for each FESTA variant, which characterize the performance of our attack.

Table 1. FESTA variants

Schemes	POKÉ-2D | INKE | PIKE	MD-LIT-SiGamal	POKÉ-4D
Δ	$\{q(A-q)\}$	$\{A^2-q^2\}$	$\{A^2-q_1^2-q_2^2\}$
(d_1, d_2)	$(q, A-q)$	$(A+q, A-q)$	–
m	1	2	–
Query count	$O(\mathcal{B}\log A)$	$O(\mathcal{B}\log A)$	–

5 Conclusion

In this work, we propose a novel GPST-type adaptive attack against several FESTA variants, which is an improvement of the MOXZ attack. An advantage of our attack is that our attack can apply to some FESTA invariants that mask the degree of the secret isogeny.

Our attack applies to POKÉ-2D, INKE, PIKE, and MD-LIT-SiGamal. In contrast, it does not appear to apply to POKÉ-4D, providing further evidence that POKÉ-4D is more secure than POKÉ-2D, as expected.

An interesting direction for future work is to investigate whether a similar adaptive attack exists for the other FESTA variants, including POKÉ-4D. However, our discussion of POKÉ-4D suggests that designing such an adaptive attack may be challenging.

Acknowledgements.

We thank the anonymous reviewers for their helpful feedback and constructive comments, which significantly improved the quality of this work. We made use of a generative AI tool to assist with refining the English writing.

This work was supported by JSPS KAKENHI Grant Number JP25K24402.

References

1. Marius A Aardal, Gora Adj, Diego F Aranha, Andrea Basso, Isaac Andrés Canales Martínez, Jorge Chávez-Saab, Maria Corte-Real Santos, Pierrick Dartois, Luca De Feo, Max Duparc, et al. SQIsign 2.0: Algorithm specifications and supporting documentation. 2025.

2. Andrea Basso and Luciano Maino. POKÉ: A compact and efficient PKE from higher-dimensional isogenies. In *Advances in Cryptology – EUROCRYPT 2025*, pages 94–123. Springer, 2025.
3. Andrea Basso, Luciano Maino, and Giacomo Pope. FESTA: Fast encryption from supersingular torsion attacks. In *Advances in Cryptology – ASIACRYPT 2023*, pages 98–126. Springer, 2023.
4. Shiping Cai, Mingjie Chen, Yi-Fu Lai, and Kaizhan Lin. PIKE: Faster isogeny-based public key encryption with pairing-assisted decryption. In *Public-Key Cryptography – PKC 2026*, pages 58–91. Springer, 2026.
5. Wouter Castryck and Thomas Decru. An efficient key recovery attack on SIDH. In *Advances in Cryptology – EUROCRYPT 2023*, pages 423–447. Springer, 2023.
6. Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: compact post-quantum signatures from quaternions and isogenies. In *Advances in Cryptology – ASIACRYPT 2020*, pages 64–93. Springer, 2020.
7. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In *Advances in Cryptology – CRYPTO'99*, pages 537–554. Springer, 1999.
8. Steven D Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. In *Advances in Cryptology – ASIACRYPT 2016*, pages 63–91. Springer, 2016.
9. Jonas Janneck, Jonas Meers, Massimo Ostuzzi, and Doreen Riepel. Snake Mackerel: An isogeny-based AKEM leveraging randomness reuse, 2025. `https://eprint.iacr.org/2025/1474`.
10. David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *Post-Quantum Cryptography – PQCrypto 2011*, pages 19–34. Springer, 2011.
11. Ernst Kani. The number of curves of genus two with elliptic differentials. 1997.
12. Hyeonhak Kim, Seokhie Hong, Suhri Kim, and Sangjin Lee. INKE: Fast isogeny-based PKE using intermediate curves, 2025. `https://eprint.iacr.org/2025/1458`.
13. David Lim and Yan Bo Ti. Adaptive attack on static POKÉ keys, 2025. `https://eprint.iacr.org/2025/1541`.
14. David Lubicz and Damien Robert. Computing isogenies between abelian varieties. *Compositio Mathematica*, 148(5):1483–1515, 2012.
15. Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, and Benjamin Wesolowski. A direct key recovery attack on SIDH. In *Advances in Cryptology – EUROCRYPT 2023*, pages 448–471. Springer, 2023.
16. Tomoki Moriya. IS-CUBE: An isogeny-based compact KEM using a boxed SIDH diagram, 2023. `https://eprint.iacr.org/2023/1506`.
17. Tomoki Moriya. On weak keys of POKÉ, 2026. `https://eprint.iacr.org/2026/1002`.
18. Tomoki Moriya, Hiroshi Onuki, Maozhi Xu, and Guoqing Zhou. Adaptive attacks against FESTA without input validation or constant-time implementation. In *International Conference on Post-Quantum Cryptography — PQCRYPTO 2024*, pages 3–19. Springer, 2024.
19. Tomoki Moriya and Miha Stopar. LIT-SiGamal: An efficient isogeny-based PKE based on a LIT diagram. In *Advances in Cryptology – ASIACRYPT 2025*, pages 275–306. Springer, 2025.
20. David Mumford, Chidambaran Padmanabhan Ramanujam, and Jurij Ivanovič Manin. *Abelian varieties*, volume 3. Oxford university press Oxford, 1974.

21. Kohei Nakagawa and Hiroshi Onuki. QFESTA: Efficient algorithms and parameters for FESTA using quaternion algebras. In *Annual International Cryptology Conference – CRYPTO 2024*, pages 75–106. Springer, 2024.
22. National Institute of Standards and Technology. Post-quantum cryptography: Additional digital signature schemes. `https://csrc.nist.gov/projects/pqc-dig-sig/round-2-additional-signatures`.
23. Damien Robert. Breaking SIDH in polynomial time. In *Advances in Cryptology – EUROCRYPT 2023*, pages 472–503. Springer, 2023.
24. Joseph Silverman. *The arithmetic of elliptic curves*, volume 106. Springer Science & Business Media, 2009.
25. László Tóth. Counting solutions of quadratic congruences in several variables revisited. *Journal of Integer Sequences*, 17(11), 2014. Article 14.11.6.
26. Jacques Vélu. Isogénies entre courbes elliptiques. *CR Acad. Sci. Paris Sér. A*, 273(5):238–241, 1971.