SQIAsignHD: SQIsignHD Adaptor Signature

Farzin Renan[1] and Péter Kutas[2]

[1] *Middle East Technical University, Ankara, Turkey*
farzin.renan@gmail.com

[2] *University of Birmingham, UK*
[2] *Eötvös Loránd University, Budapest, Hungary*
p.kutas@bham.ac.uk

Abstract

Adaptor signatures can be viewed as a generalized form of standard digital signature schemes by linking message authentication to the disclosure of a secret value. As a recent cryptographic primitive, they have become essential for blockchain applications, including cryptocurrencies, by reducing on-chain costs, improving fungibility, and enabling off-chain payments in payment-channel networks, payment-channel hubs, and atomic swaps. However, existing adaptor signature constructions are vulnerable to quantum attacks due to Shor's algorithm. In this work, we introduce SQIAsignHD, a new quantum-resistant adaptor signature scheme based on isogenies of supersingular elliptic curves, using SQIsignHD - as the underlying signature scheme - and exploiting the idea of the artificial orientation on the supersingular isogeny Diffie-Hellman key exchange protocol, SIDH, to define the underlying hard relation. We, furthermore, provide a formal security proof for our proposed scheme.

Keywords: Post-quantum Cryptography, Blockchain, Isogeny-based Cryptography, Adaptor Signature, Payment Channel Network

Acknowledgements

We express our gratitude to Luca De Feo, Andrea Basso, Simon-Philipp Merz, and the cryptography research group at IBM for their valuable feedback. We also thank the anonymous reviewers for their insightful comments, which helped improve this paper.

1 Introduction

Blockchain technology, introduced anonymously in 2009 [1], revolutionized digital payments by enabling decentralized financial transactions recorded in a distributed ledger. Each transaction is validated by network nodes through a

consensus protocol, forming the backbone of cryptocurrencies such as Bitcoin and Ethereum. However, executing transactions on-chain incurs fees based on storage and computational costs, making frequent transactions expensive. To address this, off-chain solutions were explored to reduce on-chain fees while preserving security. In this context, Andrew Poelstra introduced the concept of scriptless scripts [2], which was later formalized as adaptor signatures by [3] and [4], providing a more efficient mechanism for conditional payments without relying on complex on-chain scripts.

1.1 Adaptor Signature

An adaptor signature is a novel cryptographic primitive that builds upon the concept of a standard digital signature. It has emerged as a key tool for blockchain applications, such as cryptocurrencies, to reduce on-chain costs, improve fungibility, and support off-chain payment methods in payment-channel networks (PCNs), payment-channel hubs (PCHs), and atomic swaps. Adaptor signatures also play a crucial role in Anonymous Multihop Locks (AMHLs), enabling secure and private conditional transfers by embedding a secret within the signature. This feature ensures that transactions in AMHLs remain atomic and conditional on the revelation of the secret [5].

Technically, an adaptor signature conceals secret randomness by embedding it within the signature during the signing process. This randomness is revealed once the signature is created. Specifically, the typical procedure involves constructing a pre-signature in the first phase, converting it into a full signature using secret randomness in the second phase, and finally extracting the secret randomness from the signature using cryptographic processing. Furthermore, the signature produced by an adaptor signature can be verified using the verification algorithm of the underlying signature scheme.

An adaptor signature also possesses features that ensure its security. A signer with a secret key can create a pre-signature for any message, which can then be converted into a full signature if and only if the user possesses a valid witness to the statement. Furthermore, anyone with access to both the pre-signature and the corresponding full signature can extract the witness and reveal the hard relation.

1.2 Related Work and Our Contribution

Several works have explored adaptor signatures and their applications. Aumayr et al. [3] provide a formalization of adaptor signatures, applying them to ECDSA and Schnorr-based schemes. Malavolta et al. [5] analyze secure and privacy-preserving PCNs, identifying a new attack that affects major PCNs, such as the Lightning Network. They also define Anonymous Multihop Locks (AMHLs) and demonstrate how they can be constructed for PCNs using linear homomorphic one-way functions. Moreno-Sanchez et al. [6] show an instance of adaptor signatures applied to Monero's linkable ring signature scheme to improve scalability and address other issues. Tairi et al. [7] introduce the

PCHs protocol, with a provably secure instantiation based on adaptor signatures. However, these constructions are vulnerable to quantum adversaries due to Shor's algorithm [8].

The security of blockchain technologies largely depends on digital signature schemes built on Elliptic Curve Cryptography (ECC) to authenticate transactions. ECC's security relies on the intractability of the discrete logarithm problem, which is secure against classical computers. However, Shor's algorithm enables quantum computers to efficiently compute discrete logarithms in polynomial time. Additionally, due to Grover's algorithm [9], quantum attackers could potentially replace valid blocks with falsified ones, making blockchains susceptible to quantum attacks. In the case of Bitcoin, for instance, this could allow attackers to double-spend or steal assets from other users. As a result, post-quantum cryptography has gained increasing attention and has become a critical area of research. To secure cryptosystems against quantum adversaries, the underlying hard problems must remain intractable in the quantum setting.

In the realm of post-quantum cryptography, the first post-quantum adaptor signature, LAS [10], was established using lattice-based assumptions such as Module-LWE and Module-SIS, with a simplified form of Dilithium [11] as the underlying signature scheme. Applications using LAS require zero-knowledge proofs to ensure the extracted witness satisfies the desired norm and the hard relation. However, the most efficient proof variant is 53KB [12], leading to significant off-chain communication costs. Moreover, LAS, when used in specific applications like PCNs, can leak non-trivial information, compromising the overall privacy of the architecture.

Another attempt at designing an adaptor signature, named SQI-AS, was introduced in [13], using SQISign [14] as the underlying signature. The authors rely on SIDH [15] to apply the corresponding hard relation in their design. However, due to devastating attacks [16, 17, 18] on SIDH, SQI-AS lost its security. This vulnerability arises because SQI-AS's adapting algorithm benefits from SIDH-like operations, requiring the publication of torsion point images as auxiliary information during the pre-signature phase. This SIDH-based information is critical for breaking SIDH security and exposing the secret key isogeny. Furthermore, SQI-AS suffers from structural flaws in its design and security proof. Specifically, the shifting of the signature with secret randomness does not occur correctly. In an adaptor signature, any two of the trio (witness, pre-signature, and signature) must generate the others; however, in SQI-AS, there is no mechanism to generate the pre-signature from the witness and signature. As a result, generating the pre-signature from the full signature and witness, which is necessary for the simulator $\mathcal{S}$ of NIZK to simulate oracle queries using the signing oracle $\mathsf{Sig}^{\mathsf{SQISign}}$ and the random oracle $\mathcal{H}^{\mathsf{SQISign}}$ for adversary $\mathcal{A}$, becomes inapplicable. Furthermore, in SQI-AS, the generated signature is not directly verifiable using the standard verification procedure of the underlying signature scheme.

The only secure isogeny-based adaptor signature scheme in the literature is IAS [19], which uses CSI-FiSh [20] as the underlying signature and relies on the security of the CSIDH key exchange protocol [21]. However, IAS's efficiency

is limited by the parameter sizes of CSI-FiSh. Specifically, CSI-FiSh operates with a maximum of CSIDH-512 parameters since knowledge of the class group structure is required to efficiently compute the class group action on random group elements. CSIDH-512 is relatively slow and vulnerable to quantum subexponential attacks. Recent quantum algorithms [22, 23] have demonstrated that the parameters of CSIDH-512 do not provide the required quantum security, leading to ongoing debates about their adequacy. A new isogeny-based group action, named SCALLOP and proposed by De Feo et al. [24], addresses the scaling problem with CSI-FiSh. SCALLOP simplifies the computation of the class group structure but requires more computations to execute the group action, making it slower than CSI-FiSh.

Contribution. In light of these challenges, this work introduces a new post-quantum adaptor signature based on SQIsignHD [25], the most compact post-quantum digital signature available. Compared to other isogeny-based signature schemes, SQIsignHD is generally faster and more flexible in its parameter sets. Therefore, unlike IAS, which is restricted to CSIDH-512 parameters and is susceptible to quantum subexponential attacks, our scheme scales well to higher security levels. The signature in our construction is approximately 1.26 KB in size for a $\lambda = 128$ security level.

The main technical challenges in constructing isogeny-based adaptor signatures stem from the fact that not all post-quantum digital signatures, particularly SQIsignHD, satisfy certain homomorphic properties. As shown by [26], signature schemes derived from identification (ID) schemes with homomorphic features can be generically transformed into adaptor signature schemes. To address this, we carefully apply the concept of "shifting the signature by secret randomness" using several techniques, allowing SQIsignHD to meet this requirement. We also leverage recent advances in SIDH attacks to recover the secret witness during the extraction phase of our construction.

1.3 Organization of the Paper

Section 2 provides the necessary preliminaries for the main sections, Sections 3 and 4. These preliminaries are divided into two parts: the mathematical prerequisites for our construction and the cryptographic background required for the next sections. Section 3 introduces the new adaptor signature SQIAsignHD and examines it in detail. Section 4 analyzes the security of SQIAsignHD, providing a formal proof of its security in the random oracle model.

2 Preliminaries

Notation. A *negligible* function $\mathsf{negl} : \mathbb{N} \to \mathbb{R}$ is a function that, for every $k \in \mathbb{N}$, admits $\mathcal{O}(n^{-k})$ as its upper bound, i.e., there exists $n_0 \in \mathbb{N}$ such that for every $n \geq n_0$, it holds that $\mathsf{negl}(n) \leq 1/n^k$. We denote the uniform sampling of the variable x from the set X by $x \xleftarrow{\$} X$. Moreover, we denote a probabilistic

polynomial-time (PPT) algorithm A on input y, producing output x, by $x \xleftarrow{\$} A(y)$. If the algorithm A is deterministic polynomial-time (DPT), it is denoted by $x := A(y)$.

2.1 Elliptic Curves and Isogenies

Elliptic Curves. Let $k := \mathbb{F}_q$ be a finite field where $q = p^n$ for some prime p and positive integer n, with $\mathrm{char}(k) = p \neq 2, 3$. An *elliptic curve* E, over a field k, is a smooth projective curve of genus 1, defined over k, with a distinguished k-rational point $\infty := [0 : 1 : 0]$. Every elliptic curve over field k can be uniquely represented (up to $\bar{k}$-isomorphism) by its j-invariant. For a positive integer l, the *l-torsion subgroup* of E is defined as $E[l] := \{P \in E(\bar{k}) \mid [l]P = \infty\}$. An elliptic curve E is said to be *supersingular* if it has no nontrivial p-torsion points over $\overline{\mathbb{F}}_p$, i.e., $E[p] = \{\infty\}$. If E is supersingular, then $\mathrm{char}(k) = p$ divides $|E(\mathbb{F}_q)| - q - 1$.

Isogenies. An *isogeny* $\varphi : E_1 \to E_2$ is a surjective morphism that maps the point at infinity of E_1 to the point at infinity of E_2. Two elliptic curves E_1 and E_2 are *isogenous* over $\mathbb{F}_q$ if there exists an isogeny between them over $\mathbb{F}_q$. Furthermore, Tate's theorem [27] says that E_1 and E_2 are isogenous over $\mathbb{F}_q$ if and only if $|E_1(\mathbb{F}_q)| = |E_2(\mathbb{F}_q)|$. The *degree* of isogeny φ is the degree of the field extension $[k(E_1) : \varphi^*(k(E_2))]$, where $k(E_i)$ is the function field of E_i, $i = 1, 2$, and φ^* is the pullback of φ defined as $\varphi^* : k(E_2) \to k(E_1)$, with $\varphi^*(f) := f \circ \varphi$ for $f \in k(E_2)$. The isogeny φ is called *separable* in case the field extension is separable. If $\gcd(\deg(\varphi), \mathrm{char}(k)) = 1$, then φ is necessarily separable. Since $\varphi(\infty_{E_1}) = \infty_{E_2}$, it follows that $\varphi : E_1(k) \to E_2(k)$ is a group homomorphism. If φ is separable, then $|\ker(\varphi)| = \deg(\varphi)$. Therefore, isogenies can be characterized by their kernel. In particular, there is a one-to-one correspondence between separable isogenies (up to isomorphism of the target curve) and finite subgroups of $E_1(k)$. Isogenies can be constructed from their kernels using Vélu's formulas [28]. Such an isogeny takes the form $E \to E/G$, where G is a finite subgroup of E, and the kernel of the constructed isogeny. Since the degree of a composition of isogenies equals the product of their degrees, for any isogeny ϕ of degree $l = \prod_{i=1}^{n} l_i$, ϕ can be factored as a composition of l_i-isogenies, where $1 \leq i \leq n$ and the integers l_i need not be coprime.

If the l_i's are pairwise coprime, then reordering the l_i's produces a different set of isogenies due to the non-commutative structure of isogenies of supersingular elliptic curves under composition. Suppose that l_1 and l_2 are two coprime integers and φ is an $l_1 l_2$-isogeny. Then, φ can be decomposed in two ways: $\varphi = \psi_2 \circ \varphi_1 = \psi_1 \circ \varphi_2$, as shown in Figure 1. In this case, ψ_1 (respectively ψ_2) is called the *push-forward* of φ_1 (respectively φ_2) through φ_2 (respectively φ_1), denoted by $\psi_1 = [\varphi_2]_*\varphi_1$ (respectively $\psi_2 = [\varphi_1]_*\varphi_2$). It can be shown that $\ker(\psi_1) = \varphi_2(\ker(\varphi_1))$, and $\ker(\psi_2) = \varphi_1(\ker(\varphi_2))$. Furthermore, φ_1 (respectively φ_2) is called the *pull-back* of ψ_1 (respectively ψ_2) through φ_2 (respectively φ_1), denoted by $\varphi_1 = [\varphi_2]^*\psi_1$ (respectively $\varphi_2 = [\varphi_1]^*\psi_2$).

For a given isogeny $\alpha : E_1 \to E_2$ of degree d, its (unique) *dual* is an isogeny

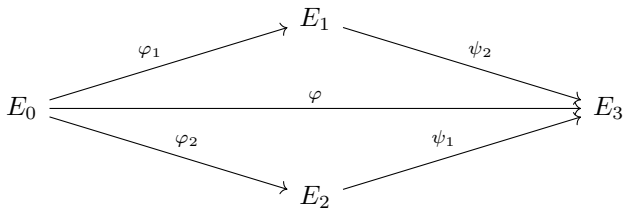

Figure 1: Commutative Isogeny Diagram.

$\hat{\alpha} : E_2 \to E_1$ of degree d such that $\alpha \circ \hat{\alpha} = [d] : E_2 \to E_2$, and $\hat{\alpha} \circ \alpha = [d] : E_1 \to E_1$. An isogeny from an elliptic curve E to itself is called an *endomorphism*. Notable examples of endomorphisms include the *multiplication-by-integer-m* map $[m] : P \mapsto m \cdot P$, and the *Frobenius* map $\pi : (x, y) \mapsto (x^q, y^q)$ of an elliptic curve defined over $E/\mathbb{F}_q$. The set of all endomorphisms on E, denoted by $\mathrm{End}(E)$, forms a ring under addition and composition, known as the *endomorphism ring* of E. Every supersingular elliptic curve in characteristic p is isomorphic to a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. This implies that each supersingular elliptic curve has an isomorphic representative defined over $\mathbb{F}_{p^2}$. For a prime $\ell \neq p$, the supersingular ℓ-isogeny graph is the graph whose vertices represent the supersingular j-invariants in $\mathbb{F}_{p^2}$, and whose edges correspond to the ℓ-isogenies between them. These graphs are connected [29], essentially undirected (since each ℓ-isogeny has a dual), $(\ell + 1)$-regular (since there are exactly $\ell + 1$ outgoing edges from each j-invariant), and Ramanujan [30].

2.2 Endomorphism Rings and Quaternion Orders

Quaternion Algebras. Let $a, b \in \mathbb{Q}^*$. A *quaternion algebra* $\mathcal{B}$ over $\mathbb{Q}$ is a four-dimensional central simple $\mathbb{Q}$-algebra defined as $\mathcal{B} := (\frac{a,b}{\mathbb{Q}}) = \mathbb{Q} + \mathbb{Q}i + \mathbb{Q}j + \mathbb{Q}k$, where $1, i, j, k$ form a basis satisfying $i^2 = a$, $j^2 = b$, and $k = ij = -ji$. Let l be a prime. The quaternion algebra $\mathcal{B}_l := \mathcal{B} \otimes_{\mathbb{Q}} \mathbb{Q}_l$ is obtained by extending the scalars of $\mathcal{B}$ from $\mathbb{Q}$ to $\mathbb{Q}_l$, where $\mathbb{Q}_l$ is the set of l-adic numbers (i.e., the fraction field of l-adic integers $\mathbb{Z}_l$ which is the localization of $\mathbb{Z}$ away from prime l). Also, we can define $\mathcal{B}_\infty := \mathcal{B} \otimes_{\mathbb{Q}} \mathbb{R}$. A quaternion algebra $\mathcal{B}$ is said to be ramified at l (including $l = \infty$) if $\mathcal{B}_l$ is a division algebra. We are only interested in $\mathcal{B}_{p,\infty}$ which is a quaternion algebra ramified at p and ∞. A *fractional ideal* I of $\mathcal{B}$ is a $\mathbb{Z}$-lattice of rank four, expressible as $I = \mathbb{Z}\alpha_1 + \mathbb{Z}\alpha_2 + \mathbb{Z}\alpha_3 + \mathbb{Z}\alpha_4$, for some $\mathbb{Q}$-basis $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ of $\mathcal{B}$.

Quaternionic Orders. An *order* is a fractional ideal that is also a subring of $\mathcal{B}$. An order $\mathcal{O}$ is *maximal* if it is not strictly contained in any other order. Let E be an elliptic curve defined over a field of characteristic p with no non-trivial p-torsion points, namely supersingular. The endomorphism algebra of such an elliptic curve is isomorphic to a quaternion algebra ramified at p and ∞, and

its endomorphism ring is isomorphic to a maximal order of the corresponding quaternion algebra, i.e., $\mathrm{End}^0(E) := \mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q} \cong \mathcal{B}_{p,\infty}$, and $\mathrm{End}(E) \cong \mathcal{O} \subset \mathcal{B}_{p,\infty}$. Conversely, for any maximal order in $\mathcal{B}_{p,\infty}$, there exists a supersingular elliptic curve over a field of characteristic p such that whose endomorphism ring is isomorphic to this maximal order. This correspondence, known as the *Deuring correspondence* [31], establishes a connection between supersingular elliptic curves and maximal orders in quaternion algebras. Specifically, given a fixed maximal order $\mathcal{O} \cong \mathrm{End}(E)$, there exists an equivalence between the category of supersingular elliptic curves (under isogenies) and the category of left fractional $\mathcal{O}$-ideals (under homomorphisms of $\mathcal{O}$-modules). Constructing a supersingular elliptic curve with a given maximal order as its endomorphism ring (one direction of the Deuring correspondence) is computationally feasible in polynomial time over carefully chosen base fields [32]. This procedure is known as the *constructive Deuring correspondence* [33]. Let $\mathcal{O} \subset \mathcal{B}_{p,\infty} \cong \mathrm{End}^0(E)$ be a maximal order, and let I be an integral left $\mathcal{O}$-ideal. The set of I-torsion points of E is defined as $E[I] := \{P \in E : \alpha(P) = 0, \text{ for all } \alpha \in I\}$, which corresponds to the kernel of I. For such an ideal I, the associated isogeny $\varphi_I : E \to E_I := \frac{E}{E[I]}$ is defined with kernel $E[I]$.

2.3 Artificial Orientation

Artificial orientation, introduced in [34], provides a method for securely computing SIDH-like operations to counteract current SIDH attacks. Let A and B be smooth, square-free, and relatively prime integers, and let p be a prime of the form $p = ABf - 1$, where f is a small cofactor. Let E be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$. An *artificial A-orientation* of E is a pair $\mathfrak{A} = (G_1, G_2)$, where G_1, G_2 are cyclic subgroups of $E[A]$ satisfying $|G_1| = |G_2| = A$ and $G_1 \cap G_2 = \{\infty\}$. A curve E equipped with $\mathfrak{A}$ is called an *artificially A-oriented curve*, denoted $(E, \mathfrak{A})$. For an artificially A-oriented curve $(E, \mathfrak{A})$, a range of isogenies can be constructed with kernels derived from $\mathfrak{A} = (G_1, G_2)$. Specifically, an isogeny ϕ is termed an *$\mathfrak{A}$-isogeny* if its kernel can be expressed as $\ker(\phi) = H_1 \oplus H_2$, where $H_1 \subseteq G_1$ and $H_2 \subseteq G_2$. Such an isogeny can be decomposed into two isogenies of relatively prime degrees as $\phi = \phi_2 \circ \phi_1$, where $\ker(\phi_1) = H_1 \subseteq G_1$ and $\ker(\phi_2) = \phi_1(H_2) \subseteq \phi_1(G_2)$.

However, as noted in [34], for a non-trivial $\mathfrak{A}$-isogeny $\phi : E \to E'$, the artificial A-orientation of E cannot be carried onto E' due to the possibility that $\phi(G_1)$ or $\phi(G_2)$ in $E'[A]$ has an order smaller than A. To address this, the degree of the isogeny must be relatively prime to A. The following definition formalizes this notion:

Definition 2.1. *For two artificially A-oriented curves $(E, \mathfrak{A})$ and $(E', \mathfrak{A}')$, and an integer B relatively prime to the A, the pairs is said to be B-isogenous if there exists a B-isogeny $\phi : E \to E'$ such that*

$$\mathfrak{A}' = (G_1', G_2') = \phi(G_1, G_2) = \phi(\mathfrak{A}).$$

With fixed generators $\langle P_1 \rangle = G_1$ and $\langle P_2 \rangle = G_2$, the subgroups G_1' and G_2'

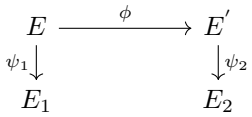

Figure 2: Parallel Isogenies

are represented as $[\alpha]\phi(P_1)$ and $[\beta]\phi(P_2)$, respectively, for $\alpha, \beta \in \mathbb{Z}/A\mathbb{Z}$. Although artificial orientations do not generate a commutative group action as in standard orientations [35], they provide sufficient structure for computing parallel isogenies. Concretely, given A-oriented curves $(E, \mathfrak{A})$ and $(E', \mathfrak{A}')$, connected by a $\mathfrak{B}$-isogeny $\phi : E \to E'$, where $\mathfrak{A} = (G_1, G_2)$ and $\mathfrak{A}' = (G_1', G_2')$, the isogenies $\psi_1 : E \to E_1$ and $\psi_2 : E' \to E_2$ are parallel, as depicted in Figure 2. Here, $E_1 = E/\langle [A_1]G_1 + [A_2]G_2 \rangle$ and $E_2 = E'/\langle [A_1]G_1' + [A_2]G_2' \rangle$, with $\ker(\psi_2) = \phi(\ker(\psi_1))$. The codomain curves E_1 and E_2 are B-isogenous, connected by the isogeny ϕ' with $\ker(\phi') = \psi_1(\ker(\phi))$. The isogenies ψ_1 and ψ_2 are thus characterized by the multiplicative decomposition $A = A_1A_2$. The properties of artificial orientation are leveraged in the pre-signature and adaptation phases of our scheme.

2.4 Computational Hardness Assumptions

The following computational hardness assumptions, which are derived from the generic problem of finding an isogeny between two isogenous elliptic curves defined over a field k, are presumed to be computationally infeasible. These assumptions underpin the security of our scheme and are employed throughout its construction.

Problem 2.2 (Supersingular Smooth Endomorphism Problem [14])**.** *Given a prime p and a supersingular elliptic curve $E/\mathbb{F}_{p^2}$, find a (non-trivial) cyclic endomorphism of E of smooth degree.*

Problem 2.3 (SSIP-A [34])**.** *Let $(E, \mathfrak{B})$ be an artificially B-oriented curve, and let A be an integer coprime to B. Let $\phi : E \to E'$ be a cyclic isogeny of degree A and let $\mathfrak{B}' = \phi(\mathfrak{B})$. Given $(E, \mathfrak{B})$ and $(E', \mathfrak{B}')$ and the degree A, compute ϕ.*

Problem 2.4 (SSIP-B [34])**.** *Let $(E, \mathfrak{B})$ be an artificially B-oriented curve, and let A be an integer coprime to B. Let $\psi : E \to E'$ be a cyclic $\mathfrak{B}$-isogeny of degree B, with $A < B$. Let P, Q be a basis of $E[A]$. Given $(E, \mathfrak{B})$, the points P, Q, and the curve E' with the points $\psi(P)$ and $\psi(Q)$, compute ψ.*

2.5 Adaptor Signature Scheme

We begin by recalling the definition of a cryptographically hard relation:

Definition 2.5 (Hard Relation)**.** *Let $\mathsf{R} \subseteq \mathsf{W} \times \mathsf{S}$ be a set of witness/statement pairs (w, s). The language of R is defined as: $\mathcal{L}_{\mathsf{R}} := \{\mathsf{s} \mid \exists\, \mathsf{w} \text{ s.t. } (\mathsf{w}, \mathsf{s}) \in \mathsf{R}\}$. The relation R is said to be a hard relation if the following conditions are satisfied:*

- *There exists a* PPT *algorithm* $\mathsf{GenR}(1^\lambda)$ *taking the security parameter* λ *as input, and outputs a witness/statement pair* $(\mathsf{w},\mathsf{s}) \in \mathsf{R}$.
- *The relation's validation is decidable in polynomial time.*
- *For any* PPT *adversary* $\mathcal{A}$*, a negligible function* negl *exists such that:*

$$Pr\left[(\mathsf{w}^*,\mathsf{s}) \in \mathsf{R} \;\middle|\; \begin{array}{c} (\mathsf{w},\mathsf{s}) \leftarrow \mathsf{GenR}(1^\lambda) \\ \mathsf{w}^* \leftarrow \mathcal{A}(\mathsf{s}) \end{array} \right] \leq \mathsf{Negl}(\lambda).$$

Non-interactive Proof System. Let $(\mathsf{w},\mathsf{s}) \in \mathsf{R}$ be cryptographically a hard relation, and $\mathcal{H}$ be a random oracle. A *non-interactive proof system* is a pair of PPT oracle algorithms (P,V), where:

- $\pi_\mathsf{w}/\perp \leftarrow \mathsf{P}^{\mathcal{H}}(\mathsf{w},\mathsf{s})$: A prover P takes a pair $(\mathsf{w},\mathsf{s}) \in \mathsf{R}$ as input and outputs a proof π_w of the statement s with witness w. If $(\mathsf{w},\mathsf{s}) \notin \mathsf{R}$, $\mathsf{P}^{\mathcal{H}}(\mathsf{w},\mathsf{s}) = \perp$.
- $0/1 \leftarrow \mathsf{V}^{\mathcal{H}}(\mathsf{s},\pi_\mathsf{w})$: A verifier V takes a pair $(\mathsf{s},\pi_\mathsf{w})$ and outputs whether the proof π_w for s is valid.

This system satisfies the following conditions:

i. Completeness: If $(\mathsf{w},\mathsf{s}) \in \mathsf{R}$ and $\pi_\mathsf{w} \leftarrow \mathsf{P}^{\mathcal{H}}(\mathsf{w},\mathsf{s})$, then there exists a negligible function negl such that $\Pr[\mathsf{V}^{\mathcal{H}} = 1] \geq 1 - \mathsf{negl}(\lambda)$.

ii. Zero-knowledge (NIZK): For a PPT algorithm $\mathcal{S}$, any (w,s), and a PPT algorithm $\mathcal{D}$, the following distributions are computationally indistinguishable:
   - $\pi_\mathsf{w} \leftarrow \mathsf{P}^{\mathcal{H}}(\mathsf{w},\mathsf{s})$ if $(\mathsf{w},\mathsf{s}) \in \mathsf{R}$ and $\pi_\mathsf{w} \leftarrow \perp$ otherwise. Output $\mathcal{D}^{\mathcal{H}}(\mathsf{w},\mathsf{s},\pi_\mathsf{w})$.
   - $\pi_\mathsf{w} \leftarrow \mathcal{S}(\mathsf{s},1)$ if $(\mathsf{w},\mathsf{s}) \in \mathsf{R}$ and $\pi_\mathsf{w} \leftarrow \mathcal{S}(\mathsf{s},0)$ otherwise. Output $\mathcal{D}^{\mathcal{H}}(\mathsf{w},\mathsf{s},\pi_\mathsf{w})$.

iii. Online-extractability: For a PPT algorithm $\mathcal{E}$, and any algorithm A, let $(\mathsf{s},\pi_\mathsf{w}) \leftarrow A^{\mathcal{H}}(\lambda)$ be the sequence of queries of A to $\mathcal{H}$, and H_A be the $\mathcal{H}$'s answers. Let $\mathsf{w} \leftarrow \mathcal{E}(s,\pi_w,H_A)$. Then it holds that

$$\Pr[(\mathsf{w},\mathsf{s}) \notin \mathsf{R} \wedge \mathsf{V}^{\mathcal{H}}(\mathsf{s},\pi_\mathsf{w}) = 1] \leq \mathsf{negl}(\lambda).$$

Digital Signature Scheme. We recall the definition of a digital signature scheme and the properties that a signature scheme must satisfy in order to be considered secure.

Definition 2.6 (Digitial Signature Scheme)**.** *A digital signature is a triple* $\Sigma = (\mathsf{KeyGen},\mathsf{Sig},\mathsf{Ver})$ *consisting of three polynomial-time algorithms:*

- $(\mathsf{sk},\mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$: *a* PPT *algorithm that takes security parameter* λ *as input, outputs a secret/public key pair* $(\mathsf{sk},\mathsf{pk})$.
- $\sigma \leftarrow \mathsf{Sig}(\mathsf{sk},m)$: *a* PPT *algorithm that takes a secret key* sk *and a message* $m \in \{0,1\}^*$ *as input, outputs a signature* σ *for the message* m.

- $0/1 \leftarrow \mathsf{Ver}(\mathsf{pk}, m, \sigma)$: *a* DPT *algorithm that takes a public key* pk*, a message* $m \in \{0,1\}^*$*, and signature* σ *as input, outputs a bit* $b \in \{0,1\}$*.*

A signature scheme is *correct* if, for any security parameter $\lambda \in \mathbb{N}$, any key pair $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$, and for any message $m \in \{0,1\}^*$, the following holds:

$$\Pr\Big[\mathsf{Ver}(\mathsf{pk}, m, \mathsf{Sig}(\mathsf{sk}, m)) = 1 \;\Big|\; (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)\Big] = 1.$$

There are several security requirements for a signature scheme, one of the most common being *existential unforgeability under chosen message attacks* (EUF-CMA). This property ensures that forging a verifiable signature on a message m without knowledge of the secret key sk is infeasible, even if the PPT adversary has access to many valid signatures on messages of its choice but message m. The formal definition of this property is as follows:

Definition 2.7 (EUF-CMA Security)**.** *A signature scheme* Σ *is* EUF-CMA *secure if for every* PPT *adversary* $\mathcal{A}$*, there exists a negligible function* negl *such that*

$$Pr[\mathsf{SigForge}_{\mathcal{A},\Sigma}(\lambda) = 1] \leq \mathsf{negl}(\lambda),$$

where the experiment $\mathsf{SigForge}_{\mathcal{A},\Sigma}$ *is defined as follows:*

$\mathsf{SigForge}_{\mathcal{A},\Sigma}(\lambda)$	$\mathcal{O}_S(m)$
$\mathcal{Q} \leftarrow \emptyset$	$\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}, m)$
$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$	$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
$(m, \sigma) \leftarrow \mathcal{A}^{\mathcal{O}_S}(\mathsf{pk})$	**return** σ
return $(m \notin \mathcal{Q} \wedge \mathsf{Ver}(\mathsf{pk}, m, \sigma))$	

A stronger definition is *strong existential unforgeability under chosen message attacks* (SUF-CMA), which ensures the difficulty of transforming a valid signature on a message m into another valid signature on m. The formal definition is as follows:

Definition 2.8 (SUF-CMA Security)**.** *A signature scheme* Σ *is* SUF-CMA *secure if for every* PPT *adversary* $\mathcal{A}$*, there exists a negligible function* negl *such that*

$$Pr[\mathsf{StrongSigForge}_{\mathcal{A},\Sigma}(\lambda) = 1] \leq \mathsf{negl}(\lambda),$$

where the experiment $\mathsf{StrongSigForge}_{\mathcal{A},\Sigma}$ *is defined as follows:*

$\mathsf{StrongSigForge}_{\mathcal{A},\Sigma}(\lambda)$	$\mathcal{O}_S(m)$
$\mathcal{Q} \leftarrow \emptyset$	$\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}, m)$
$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$	$\mathcal{Q} := \mathcal{Q} \cup \{m, \sigma\}$
$(m, \sigma) \leftarrow \mathcal{A}^{\mathcal{O}_S}(\mathsf{pk})$	**return** σ
return $((m, \sigma) \notin \mathcal{Q} \wedge \mathsf{Ver}(\mathsf{pk}, m, \sigma))$	

Adaptor Signature Scheme. An adaptor signature is a cryptographic primitive that extends an ordinary digital signature. It hides secret randomness within the signature, which is only revealed once the signature is generated. The process begins with the generation of a pre-signature, which is then adapted into a full signature by applying secret randomness. In the final step, this secret randomness is extracted through cryptographic procedures. The signature produced is verifiable using the verification algorithm of the underlying signature scheme. An adaptor signature also has specific security properties. For any statement $\mathsf{s} \in \mathcal{L}_{\mathsf{R}}$, a signer with secret key sk can produce a pre-signature $\tilde{\sigma}$ on any message m. This pre-signature can be adapted into a full signature σ if and only if the user has a witness w to the statement s. Additionally, anyone with access to the pre-signature $\tilde{\sigma}$, (full) signature σ, and statement s can extract the witness w, thus revealing the hard relation.

The formal definition of an adaptor signature scheme and its properties are given as follows:

Definition 2.9 (Adaptor Signature Scheme)**.** *An adaptor signature scheme with respect to a hard relation* R *and a signature scheme* $\Sigma = (\mathsf{KeyGen}, \mathsf{Sig}, \mathsf{Ver})$ *is a quadruple* $\Xi_{\mathsf{R},\Sigma} = (\mathsf{PreSig}, \mathsf{PreVer}, \mathsf{Adapt}, \mathsf{Ext})$ *defined as:*

- $\tilde{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, \mathsf{s})$: *a* PPT *algorithm that takes a secret key* sk*, a message* $m \in \{0,1\}^*$*, and a statement* $\mathsf{s} \in \mathcal{L}_{\mathsf{R}}$*, outputs a pre-signature* $\tilde{\sigma}$*.*
- $0/1 \leftarrow \mathsf{PreVer}(\mathsf{pk}, m, \mathsf{s}, \tilde{\sigma})$: *a* DPT *algorithm that takes a public key* pk*, a message* $m \in \{0,1\}^*$*, a statement* $\mathsf{s} \in \mathcal{L}_{\mathsf{R}}$*, and a pre-signature* $\tilde{\sigma}$*, produces a bit* $b \in \{0,1\}$*.*
- $\sigma \leftarrow \mathsf{Adapt}(\tilde{\sigma}, \mathsf{w})$: *a* DPT *algorithm that takes a valid pre-signature* $\tilde{\sigma}$*, and a witness* w*, generates a signature* σ*.*
- $\mathsf{w}/\perp \leftarrow \mathsf{Ext}(\sigma, \tilde{\sigma}, \mathsf{s})$: *a* DPT *algorithm that takes a pre-signature* $\tilde{\sigma}$*, a corresponding signature* σ*, and a statement* $\mathsf{s} \in \mathcal{L}_{\mathsf{R}}$*, produces a witness* w *to the statement* s*, or* $\perp$*.*

In an adaptor signature scheme, $\Xi_{\mathsf{R},\Sigma}$, the algorithm GenR generates witness/statement pairs (w, s) based on the underlying hard relation R. As mentioned earlier, several properties ensure the security of an adaptor signature scheme. The first property is *pre-signature correctness*, which guarantees that an honestly generated pre-signature can be adapted into a valid signature.

Definition 2.10 (Pre-signature Correctness)**.** *An adaptor signature scheme* $\Xi_{\mathsf{R},\Sigma}$ *satisfies pre-signature correctness if for any* $\lambda \in \mathbb{N}$*, any message* $m \in \{0,1\}^*$*, and any witness/statement pair* (w, s)*, the following holds:*

$$Pr\left[\begin{array}{c} \mathsf{PreVer}(\mathsf{pk}, m, \mathsf{s}, \tilde{\sigma}) = 1 \\ \mathsf{Ver}(\mathsf{pk}, m, \sigma) = 1 \\ (\mathsf{w}', \mathsf{s}) \in \mathsf{R} \end{array} \middle| \begin{array}{c} (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ \tilde{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, \mathsf{s}) \\ \sigma := \mathsf{Adapt}(\tilde{\sigma}, \mathsf{w}) \\ \mathsf{w}' := \mathsf{Ext}(\sigma, \tilde{\sigma}, \mathsf{s}) \end{array} \right] = 1.$$

The second property of an adaptor signature is *pre-signature adaptability*. It states that any valid (though not necessarily honestly generated) pre-signature for a statement s can be adapted into a valid signature using a witness w such that $(\mathsf{w}, \mathsf{s}) \in \mathsf{R}$.

Definition 2.11 (Pre-signature Adaptability)**.** *An adaptor signature scheme $\Xi_{\mathsf{R},\Sigma}$ satisfies pre-signature adaptability if for any $\lambda \in \mathbb{N}$, message $m \in \{0,1\}^*$, witness/statement pair $(\mathsf{w}, \mathsf{s}) \in \mathsf{R}$, key pair $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$, and pre-signature $\tilde{\sigma} \leftarrow \{0,1\}^*$ such that $\mathsf{PreVer}(\mathsf{pk}, m, \mathsf{s}, \tilde{\sigma}) = 1$, the following holds:*

$$Pr[\mathsf{Ver}(\mathsf{pk}, m, \mathsf{Adapt}(\tilde{\sigma}, \mathsf{w})) = 1] = 1.$$

Another key property is *existential unforgeability under chosen message attack* (aEUF-CMA). This property states that even with access to a pre-signature on a message m with respect to a random statement $\mathsf{s} \in \mathcal{L}_\mathsf{R}$, it is computationally infeasible for an adversary to forge a valid signature σ for m.

Definition 2.12 (aEUF-CMA Security)**.** *An adaptor signature scheme $\Xi_{\mathsf{R},\Sigma}$ is* aEUF-CMA *secure if for any* PPT *adversary $\mathcal{A}$, there exists a negligible function* negl *such that*

$$Pr[\mathsf{aSigForge}_{\mathcal{A},\Xi_{\mathsf{R},\Sigma}}(\lambda) = 1] \leq \mathsf{negl}(\lambda),$$

where the experiment $\mathsf{aSigForge}_{\mathcal{A},\Xi_{\mathsf{R},\Sigma}}$ is defined as follows:

$\mathsf{aSigForge}_{\mathcal{A},\Xi_{\mathsf{R},\Sigma}}(\lambda)$

1: $\mathcal{Q} := \emptyset$
2: $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$
3: $m \leftarrow \mathcal{A}^{\mathcal{O}_S, \mathcal{O}_{pS}}(\mathsf{pk})$
4: $(\mathsf{w}, \mathsf{s}) \leftarrow \mathsf{GenR}(1^\lambda)$
5: $\tilde{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, \mathsf{s})$
6: $\sigma \leftarrow \mathcal{A}^{\mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\sigma}, \mathsf{s})$
7: **return** $m \notin \mathcal{Q} \wedge \mathsf{Ver}(\mathsf{pk}, m, \sigma)$

$\mathcal{O}_S(m)$

1: $\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}, m)$
2: $\mathcal{Q} := \mathcal{Q} \cup \{m\}$
3: **return** σ

$\mathcal{O}_{pS}(m, s)$

1: $\tilde{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, \mathsf{s})$
2: $\mathcal{Q} := \mathcal{Q} \cup \{m\}$
3: **return** $\tilde{\sigma}$

The fourth and last property is called *witness extractability*. This property guarantees that once a pre-signature is adapted into a (full) signature, it must not be the case that the witness for the original statement used to generate the pre-signature cannot be extracted.

Definition 2.13 (Witness Extractability)**.** *An adaptor signature scheme $\Xi_{\mathsf{R},\Sigma}$ is witness extractable if for any* PPT *adversary $\mathcal{A}$, there exists a negligible function* negl *such that the following holds:*

$$Pr[\mathsf{aWitExt}_{\mathcal{A},\Xi_{\mathsf{R},\Sigma}}(\lambda) = 1] \leq \mathsf{negl}(\lambda),$$

where the experiment $\mathsf{aWitExt}_{\mathcal{A},\Xi_{\mathsf{R},\Sigma}}$ is defined as follows:

$\mathsf{aWitExt}_{\mathcal{A},\Xi_{\mathsf{R},\Sigma}}(\lambda)$

1 : $\mathcal{Q} := \emptyset$
2 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$
3 : $(m, \mathsf{s}) \leftarrow \mathcal{A}^{\mathcal{O}_S, \mathcal{O}_{pS}}(\mathsf{pk})$
4 : $\tilde{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, \mathsf{s})$
5 : $\sigma \leftarrow \mathcal{A}^{\mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\sigma})$
6 : $\mathsf{w}^* := \mathsf{Ext}(\sigma, \tilde{\sigma}, \mathsf{s})$
7 : **return** $(m \notin \mathcal{Q} \wedge (\mathsf{w}^*, \mathsf{s}) \notin \mathsf{R} \wedge \mathsf{Ver}(\mathsf{pk}, m, \sigma))$

$\mathcal{O}_S(m)$

1 : $\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}, m)$
2 : $\mathcal{Q} := \mathcal{Q} \cup \{m\}$
3 : **return** σ

$\mathcal{O}_{pS}(m, \mathsf{s})$

1 : $\tilde{\sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, \mathsf{s})$
2 : $\mathcal{Q} := \mathcal{Q} \cup \{m\}$
3 : **return** $\tilde{\sigma}$

In light of the above properties of the adaptor signature scheme, the following definition is established:

Definition 2.14 (Secure Adaptor Signature Scheme)**.** *An adaptor signature scheme $\Xi_{\mathsf{R},\Sigma}$ is secure if it is* aEUF-CMA *secure, pre-signature adaptable, and witness extractable.*

2.6 SQIsignHD

SQIsignHD [25] is a post-quantum digital signature scheme derived from SQISign [14], incorporating recent advancements stemming from attacks [16, 17, 18] on SIDH. These advancements enable efficient representation of isogenies of arbitrary degrees. In comparison to SQISign, SQIsignHD provides improved scalability for higher security levels, greater simplicity and efficiency, and smaller signature sizes. The protocol is outlined as follows:

Let $D_\varphi := \prod_{i=1}^{n} \ell_i^{e_i}$ be a smooth integer and $\mu(D_\varphi) := \prod_{i=0}^{n} \ell_i^{e_i - 1}(\ell_i + 1)$. Also, let $\Phi_{D_\varphi}(E, h)$ be an arbitrary function that maps an integer $h \in [1, \mu(D_\varphi)]$ to a non-backtracking isogeny of degree D_φ starting at E. Consider a hash function $\mathsf{H} : \{0,1\}^* \to [1, \mu(D_\varphi)]$ which is cryptographically secure.

Setup. Choose a prime p and supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$ with known endomorphism ring $\mathcal{O}_0 \cong \mathrm{End}(E_0)$, where E_0 has smooth torsion defined over a small extension of $\mathbb{F}_{p^2}$ of degree 1 or 2.

KeyGen. Generate a random secret isogeny $\tau : E_0 \to E_A$ of fixed smooth degree D_τ. The secret/public key pair is $(\mathsf{sk}, \mathsf{pk}) := (\tau, E_A)$.

Sign. Generate a random (secret) commitment isogeny $\psi : E_0 \to E_1$. For signing a message m, build the isogeny $\Phi_{D_\varphi}(E_A, h) = \varphi : E_A \to E_2$, where $h = \mathsf{H}(j(E_1), m)$. From the knowledge of the secret key τ, and isogenies φ, ψ, construct an efficient representation $R = (\sigma(P_1), \sigma(P_2), q)$ given by the image of torsion points by a response isogeny $\sigma : E_1 \to E_2$ and return the pair $\Sigma := (E_1, R)$ as a signature.

Verify. Upon receiving a signature $\Sigma = (E_1, R)$ associated with the message m and public key E_A, the verifier recovers $h = H(j(E_1), m)$ and then

computes $\varphi = \Phi(E_A, h) : E_A \to E_2$. Finally, the verifier checks that R represents correctly an isogeny $\sigma : E_1 \to E_2$ by computing a higher dimensional isogeny, as described in SQIsignHD.

The public parameters for SQIsignHD are easy to generate. Specifically, the underlying prime is of the form $p = c\ell^f \ell'^{f'} - 1$, where ℓ and ℓ' are distinct primes (in practice, $\ell = 2$ and $\ell' = 3$), $c \in \mathbb{N}^*$ is a small cofactor, and $\ell^f \approx \ell'^{f'} \approx p^{1/2}$. This ensures sufficient accessible torsion for isogeny computations. This flexibility allows replacing ℓ^f and $\ell'^{f'}$ with a collection of small primes, as discussed in Section 3.1, providing a suitable setting for applying artificial orientation in our construction.

The signature, as shown in the protocol, is the data $(E_1, \sigma(P_1), \sigma(P_2), q)$, with $q \approx p^{1/2}$, $\sigma : E_1 \to E_2$ a q-isogeny, and (P_1, P_2) a basis of $E_1[\ell^f]$. This data is based on the following definition:

Definition 2.15 ([25]). *Suppose that* A *is an algorithm and* $\varphi : E \to E'$ *is an* $\mathbb{F}_q$*-rational isogeny. Then, an efficient representation of isogeny* φ *(with respect to* A*) is some data* $\mathsf{D} \in \{0,1\}^*$ *such that:*

1. D *has polynomial size in* $\log(\deg(\varphi))$ *and* $\log(q)$.
2. *On input* D *and* $P \in E(\mathbb{F}_{q^k})$, A *returns* $\varphi(P)$ *in polynomial time in* $k \log(q)$ *and* $\log(\deg(\varphi))$.

3 New Adaptor Signature Construction

In this section, we present a new post-quantum adaptor signature scheme built upon SQIsignHD [25] as the underlying signature scheme. To incorporate the associated hard relation, we utilize the hybrid variant of binSIDH, denoted as $\mathsf{binSIDH}^{\mathsf{hyb}}$, introduced in Section 5 of [34]. This variant combines oriented and non-oriented approaches, wherein one party computes binSIDH-like isogenies while the other performs SIDH-like isogenies.

Currently, the only secure post-quantum isogeny-based adaptor signature scheme is IAS, proposed in [19], which is built upon CSI-FiSh [20]. However, IAS faces efficiency limitations due to the parameter sizes required by CSI-FiSh. Specifically, CSI-FiSh operates at most on the CSIDH-512 parameters, as efficient computation of the class group action on uniformly random group elements necessitates prior knowledge of the class group structure. In the following, we provide a detailed description of our proposed post-quantum adaptor signature scheme and present the corresponding protocol in Algorithm 1.

3.1 Public Parameters

To deploy our protocol, we first establish a set of initial public parameters. These parameters are inspired by those employed in $\mathsf{binSIDH}^{\mathsf{hyb}}$ and SQIsignHD. The setup of our scheme is defined as follows.

We select a prime p of the form $p = ABCf - 1$, where $A = 2^a$, $B = \prod_{i=1}^{t} \ell_i$, and $C = 3^c$ are pairwise relatively prime integers. Here, f is a small cofactor, ℓ_i's represent distinct small primes, and the sizes of A, B, and C are chosen such that $A \approx C \approx p^{1/4}$ and $B \approx p^{1/2}$. Let $E_0/\mathbb{F}_{p^2}$ denote a supersingular elliptic curve with a known endomorphism ring $\mathrm{End}(E_0) \cong \mathcal{O}_0 \subset \mathcal{B}_{p,\infty}$, and assume $|E_0(\mathbb{F}_{p^2})| = (p+1)^2$. We define $\mathfrak{B} = (G_1, G_2)$ as an artificial B-orientation on E_0, and fix a basis $\langle P, Q \rangle = E_0[C]$. Additionally, we employ a cryptographically secure hash function $\mathsf{H} : \{0,1\}^* \to [1, \mu(D_\varphi)]$, analogous to the one used in SQIsignHD.

3.2 Key Generation & Hard Relation

The key generation procedure follows the standard process in SQIsignHD. Specifically, a random secret isogeny $\tau : E_0 \to E_\tau$ is generated, and the secret/public key pair is defined as $(\mathsf{sk}, \mathsf{pk}) := (\tau, E_\tau)$.

To define the hard relation in our scheme, we set the witness/statement pairs as follows:

$$\mathsf{R}_{\mathfrak{A}} := \left\{ \; (w, I_w := (E_w, w(\mathfrak{B}), \pi_w)) \;\middle|\; \begin{array}{c} w : E_0 \to E_w := E_0/\langle P + [\alpha]Q \rangle, \\ \text{where } \langle P, Q \rangle = E_0[C], \alpha \in \mathbb{Z}/C\mathbb{Z}. \\ (E_0, \mathfrak{B}) \text{ is artificially } B\text{-oriented.} \end{array} \right\},$$

where w denotes the secret witness isogeny with the artificially B-oriented curve $(E_0, \mathfrak{B})$ as its domain, while $(E_w, w(\mathfrak{B}))$ constitutes the statement, consisting of the target elliptic curve E_w and the image of the artificial B-orientation $\mathfrak{B} = (G_1, G_2)$ under the isogeny w. Additionally, π_w denotes a zero-knowledge proof that $(w, (E_w, w(\mathfrak{B})))$ is a valid instance of the hard relation $\mathsf{R}_{\mathfrak{A}}$.

3.3 Pre-signature

The pre-signing algorithm shares similarities with the signing procedure described in the SQIsignHD protocol but introduces notable differences, particularly in generating the commitment isogeny (and the corresponding curve) and incorporating additional elements required during the adaptation phase.

Unlike SQIsignHD, our scheme's pre-signature phase involves two (secret) commitment isogenies. The first serves a role similar to the commitment isogeny in SQIsignHD, while the second, generated in conjunction with the statement curve, lays the foundation for the adaptation phase. We now examine these components in detail.

Commitment $\boldsymbol{\psi}$. The first commitment isogeny, ψ, is a $\mathfrak{B}$-oriented isogeny $\psi : E_0 \to E_\psi$, generated by uniformly sampling a vector $\vec{b}$ from $\{1,2\}^t$ to compute

$$\ker(\psi) := \langle G_{b_1}^1, G_{b_2}^2, \ldots, G_{b_t}^t \rangle,$$

where $G_1 := \langle G_1^1, G_1^2, \ldots, G_1^t \rangle$ and $G_2 := \langle G_2^1, G_2^2, \ldots, G_2^t \rangle$, with $|G_1^i| = |G_2^i| = \ell_i$, for $1 \le i \le t$.

Furthermore, using the isogeny ψ, we compute the images of the publicly given points P and Q under ψ. These images are denoted as $S := (\psi(P), \psi(Q))$.

Commitment $\boldsymbol{\psi'}$. After parsing I_w as $(E_w, w(\mathfrak{B}), \pi_w)$ and verifying that $1 = \mathsf{NIZK.V}(E_w, \pi_w)$, the second commitment isogeny, ψ', is derived by pushing forward the first commitment isogeny, ψ, through the witness $w : E_0 \to E_w$ using the component $w(\mathfrak{B})$ of the public statement. Here, $w(\mathfrak{B})$ represents the image of the artificially B-orientation $\mathfrak{B}$ under the witness isogeny w. Formally, this is defined as $\psi' := [w]_*\psi : E_w \to E_1$.

As a result, we obtain the second commitment curve, E_1, whose j-invariant is used to compute the challenge isogeny. Finally, we compute a zero-knowledge proof, $\pi_{\psi'}$, to demonstrate that E_1 is the codomain of the isogeny parallel to ψ.

Now, the challenge and pre-signature isogenies are constructed as follows:

Challenge $\boldsymbol{\varphi}$. To generate the challenge isogeny, the j-invariant of the second commitment curve E_1 is combined with a message m to produce an isogeny starting at the public key E_τ. Specifically, for $h := \mathsf{H}(j(E_1), m)$, the challenge isogeny is defined as $\varphi := \Phi(E_\tau, h) : E_\tau \to E_2$.

Pre-signature $\tilde{\boldsymbol{\sigma}}$. To complete the pre-signing phase for a message m, given knowledge of the endomorphism ring $\mathrm{End}(E_0) \cong \mathcal{O}_0$ and the isogenies τ, φ, and ψ, an efficient representation $\mathcal{R}_{\tilde{\sigma}} := (\tilde{\sigma}(R_1), \tilde{\sigma}(R_2), \deg(\tilde{\sigma}))$ is constructed. This representation is derived from the images of a canonically determined basis $\langle R_1, R_2 \rangle$ of $E_\psi[A]$ under the pre-signature isogeny $\tilde{\sigma} : E_\psi \to E_2$.

Thus, the pre-signature tuple is defined as $\tilde{\Sigma} := (E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}})$, and the pre-signing algorithm is formally expressed as

$$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\mathsf{sk}, m, \mathsf{s}) = \mathsf{PreSig}(\tau, m, I_w).$$

3.4 Pre-verification

The pre-verification process begins by parsing $S = (\psi(P), \psi(Q))$ and checking the equality of the Weil pairings: $e_C(\psi(P), \psi(Q)) = e_C(P, Q)^B$. Next, using the statement curve E_w, extracted from $I_w = (E_w, w(\mathfrak{B}), \pi_w)$, and the commitment curve E_1, the proof $\pi_{\psi'}$ is verified by ensuring that: $1 = \mathsf{NIZK.V}((E_w, E_1), \pi_{\psi'})$, which confirms that the isogeny ψ' is an isogeny from the statement curve E_w to the curve E_1, parallel to the isogeny $\psi : E_0 \to E_\psi$. Subsequently, the challenge isogeny $\varphi = \Phi(E_\tau, h) : E_\tau \to E_2$ is recovered, where $h = \mathsf{H}(j(E_1), m)$. Finally, using the canonical basis $\langle R_1, R_2 \rangle = E_\psi[A]$, it is verified that the representation $\mathcal{R}_{\tilde{\sigma}} = (\tilde{\sigma}(R_1), \tilde{\sigma}(R_2), \deg(\tilde{\sigma}))$ correctly represents an isogeny $\tilde{\sigma} : E_\psi \to E_2$ by computing a higher-dimensional isogeny, as outlined in SQIsignHD. If any of these conditions are not met, the process aborts. The pre-verification algorithm is thus defined as follows:

$$0/1 \leftarrow \mathsf{PreVer}(\mathsf{pk}, m, \mathsf{s}, \tilde{\Sigma}) = \mathsf{PreVer}\left(E_\tau, m, I_w, (E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}})\right).$$

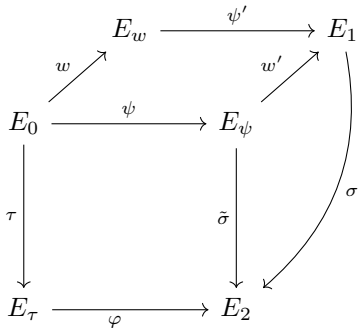

Figure 3: SQIAsignHD Protocol

3.5 Adaptation

To adapt the pre-signature into a (full) signature, the parallel isogeny w' to the witness isogeny w is first computed using the additional information $S = (\psi(P), \psi(Q))$. This ensures that the resulting second commitment curve, E_1, coincides with the codomain of the w', i.e., $w' := [\psi]_* w : E_\psi \to E_1$, as depicted in Figure 3.

Next, an efficient representation of the (full) signature isogeny $\sigma := \tilde{\sigma} \circ \hat{w'} : E_1 \to E_2$ is constructed by employing the algorithm A, as described in Definition 2.15. The steps are as follows:

1. Determine a canonical basis $\langle P_0, Q_0 \rangle := E_1[AC]$.
2. Compute $\hat{w'}(P_0)$ and $\hat{w'}(Q_0)$, where $\hat{w'} : E_1 \to E_\psi$ is the dual of w'.
3. Evaluate $\mathsf{A}(\mathcal{R}_{\tilde{\sigma}}, \hat{w'}(P_0)) =: \sigma(P_0)$ and $\mathsf{A}(\mathcal{R}_{\tilde{\sigma}}, \hat{w'}(Q_0)) =: \sigma(Q_0)$.
4. Construct the efficient representation of the isogeny $\sigma : E_1 \to E_2$:

$$\mathcal{R}_\sigma := \big(\sigma(P_0), \sigma(Q_0), \deg(\sigma)\big).$$

The signature is defined as $\Sigma := (E_1, \mathcal{R}_\sigma)$. Accordingly, the adaptation algorithm is specified as follows:

$$\Sigma := (E_1, \mathcal{R}_\sigma) \leftarrow \mathsf{Adapt}(\tilde{\Sigma}, \mathsf{w}) = \mathsf{Adapt}\big((E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}}), w\big).$$

3.6 Extraction

In the final phase of our scheme, the goal is to extract the secret witness isogeny w using the publicly known pre-signature $\tilde{\Sigma}$ and signature Σ. This is achieved through two computational approaches: one involves computing the discrete logarithm (of modulus a sufficiently smooth integer), denoted by $\mathcal{A}_{\mathsf{DLP}}$, and the other is an attack for key recovery of an isogeny satisfying $n^2 > 4d$ via the SIDH attack [16], denoted by $\mathcal{A}_{\mathsf{SIDH}}$, where d is the degree of the isogeny and

n is the order of the given torsion points information. Additionally, we utilize the algorithm A, as defined in Definition 2.15. The extraction process proceeds with the following steps:

1. Determine a canonical basis $\langle P_1, Q_1 \rangle = E_1[N]$ such that $4C < N^2$.
2. Set $P' := \mathsf{A}(\mathcal{R}_\sigma, P_1)$, $Q' := \mathsf{A}(\mathcal{R}_\sigma, Q_1)$, where $P', Q' \in E_2[N]$.
3. Define $X := \hat{w'}(P_1)$ and $Y := \hat{w'}(Q_1)$ as unknowns, for which we seek to determine their values. Then, X and Y can be written as
$$X = [a]P_\psi + [b]Q_\psi, \qquad Y = [c]P_\psi + [d]Q_\psi,$$
for some unknown values $a, b, c, d \in \mathbb{Z}/N\mathbb{Z}$, where $\langle P_\psi, Q_\psi \rangle = E_\psi[N]$.
4. From the action of the isogeny $\tilde{\sigma}$ on X and Y, we have
$$\tilde{\sigma}(X) = \tilde{\sigma}([a]P_\psi + [b]Q_\psi) = [a]\tilde{\sigma}(P_\psi) + [b]\tilde{\sigma}(Q_\psi),$$
$$\tilde{\sigma}(Y) = \tilde{\sigma}([c]P_\psi + [d]Q_\psi) = [c]\tilde{\sigma}(P_\psi) + [d]\tilde{\sigma}(Q_\psi),$$
which gives the following system of equations:
$$[a]\tilde{\sigma}(P_\psi) + [b]\tilde{\sigma}(Q_\psi) = P',$$
$$[c]\tilde{\sigma}(P_\psi) + [d]\tilde{\sigma}(Q_\psi) = Q',$$
where P' and Q' were obtained in step 2.
5. Set initial values for a and c (we let $a = c = 1$). Using the Discrete Logarithm (DL) algorithm, $\mathcal{A}_{\mathsf{DLP}}$, the values of b and d can be determined. This allows us to determine the action of $\hat{w'}$ on P_1 and Q_1, i.e., X and Y, respectively.
6. Apply the SIDH attack, $\mathcal{A}_{\mathsf{SIDH}}$, to find the kernel of the isogeny $\hat{w'}$. Then, compute the dual of $\hat{w'}$, which is the isogeny $w' : E_\psi \to E_1$.
7. Uncover the secret witness $\alpha \in \mathbb{Z}/C\mathbb{Z}$ by expressing the $\ker(w')$ in terms of the already given torsion basis $S = (\psi(P), \psi(Q))$ on $E_\psi[C]$, that is, $\ker(w') = \langle \psi(P) + [\alpha]\psi(Q) \rangle$. This is sufficient to recover the witness isogeny via $\ker(w) = \langle P + [\alpha]Q \rangle$, where P and Q are public.

Thus, the extraction algorithm is defined as follows:

$$w/\perp \leftarrow \mathsf{Ext}(\Sigma, \tilde{\Sigma}, \mathsf{s}) = \mathsf{Ext}\Big((E_1, \mathcal{R}_\sigma), (E_1, \pi_{\psi'}, E_\psi, S, R_{\tilde{\sigma}}), I_w\Big).$$

Algorithm 1 SQIAsignHD : Adaptor Signature $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$

Public Parameters. A prime $p = ABCf - 1$, where $A = 2^a$, $B = \prod_{i=1}^{t} \ell_i$, and $C = 3^c$ are pairwise coprime integers, f is some (small) cofactor, ℓ_i's are distinct small primes, $A \approx C \approx p^{1/4}$, and $B \approx p^{1/2}$. A supersingular elliptic curve $E_0/\mathbb{F}_{p^2}$ with known $\mathrm{End}(E_0) \cong \mathcal{O}_0 \subset \mathcal{B}_{p,\infty}$, and $|E_0(\mathbb{F}_{p^2})| = (p+1)^2$. An artificial B-orientation $\mathfrak{B} = (G_1, G_2)$ on E_0, and a torsion basis $\langle P, Q\rangle = E_0[C]$. A secure hash function $\mathsf{H} : \{0,1\}^* \to [1, \mu(D_\varphi)]$.
Procedure $\mathsf{PreSig}(\mathsf{sk}, \mathsf{m}, \mathsf{s})$
  Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$.
  Verify that $1 = \mathsf{NIZK.V}(E_w, \pi_w)$.
  Compute a secret isogeny $\psi : E_0 \to E_\psi$.
  Compute the image of P, Q under ψ, and set $S := (\psi(P), \psi(Q))$.
  Compute the push-forward $\psi' := [w]_*\psi : E_w \to E_1$ via $w(\mathfrak{B})$.
  Compute the zero-knowledge $\pi_{\psi'}$ showing that E_1 is honestly generated.
  Compute $\varphi := \Phi(E_\tau, h) : E_\tau \to E_2$, where $h := \mathsf{H}(j(E_1), m)$.
  Compute $\mathcal{R}_{\tilde{\sigma}} := (\tilde{\sigma}(R_1), \tilde{\sigma}(R_2), \tilde{q})$ where $\tilde{\sigma} : E_\psi \to E_2$ of degree $\tilde{q}$.
  Return $\tilde{\Sigma} := (E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}})$
Procedure $\mathsf{PreVer}(\mathsf{pk}, \mathsf{m}, \mathsf{s}, \tilde{\Sigma})$
  Parse $\tilde{\Sigma}$ as $(E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}})$.
  Parse S as $(\psi(P), \psi(Q))$.
  Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$.
  Check that $e_C(\psi(P), \psi(Q)) = e_C(P, Q)^B$.
  Verify that $1 = \mathsf{NIZK.V}((E_w, E_1), \pi_{\psi'})$.
  Recompute $h = \mathsf{H}(j(E_1), m)$ and recover $\varphi := \Phi(E_\tau, h) : E_\tau \to E_2$.
  Check that $\mathcal{R}_{\tilde{\sigma}}$ correctly represent $\tilde{\sigma} : E_\psi \to E_2$.
  Return $0/1$.
Procedure $\mathsf{Adapt}(\tilde{\Sigma}, \mathsf{w})$
  Compute push-forward $w' := [\psi]_* w : E_\psi \to E_1$ via S.
  Determine a canonical basis $\langle P_0, Q_0\rangle := E_1[AC]$.
  Compute $\sigma(P_0) := \mathsf{A}(\mathcal{R}_{\tilde{\sigma}}, \hat{w'}(P_0))$, and $\sigma(Q_0) := \mathsf{A}(\mathcal{R}_{\tilde{\sigma}}, \hat{w'}(Q_0))$.
  Set $\mathcal{R}_\sigma := (\sigma(P_0), \sigma(Q_0), q)$ where $\sigma : E_1 \to E_2$, and $q := \deg(\sigma)$.
  Return $\Sigma := (E_1, \mathcal{R}_\sigma)$
Procedure $\mathsf{Ext}(\tilde{\Sigma}, \Sigma, \mathsf{s})$
  Parse Σ as $(E_1, \mathcal{R}_\sigma)$.
  Recover $\hat{w'} : E_1 \to E_\psi$ via $\mathcal{A}_{\mathsf{DLP}}$ and $\mathcal{A}_{\mathsf{SIDH}}$, and compute $\ker(w')$.
  Represent $\ker(w')$ in terms of the already given basis $S = (\psi(P), \psi(Q))$.
  Extract the witness $\alpha \in \mathbb{Z}/C\mathbb{Z}$ for which $\ker(w') = \langle \psi(P) + [\alpha]\psi(Q)\rangle$.
  Return $\perp$ / w

3.7 Parameter Setting

We follow the parameterization strategy established in SQIsignHD and binSIDH, including its hybrid variant $\mathsf{binSIDH}^{\mathsf{hyb}}$, to select the underlying prime p in the form $p = ABCf - 1$, where $A = 2^a$, $B = \prod_{i=1}^{t} \ell_i$, and $C = 3^c$. The parameters are carefully chosen to satisfy $A \approx C \approx p^{1/4}$ and $B \approx p^{1/2}$. Here, the ℓ_i's

represent distinct small primes greater than 3, and f is a small cofactor.

Signature Size. To achieve 128-bit post-quantum security, the parameter configuration inspired by binSIDH sets B as the product of the first 134 primes greater than 3, i.e., $t = 134$. Under this selection, the prime p has an approximate bit length of $|p| \approx 2128$. Moving forward, we examine the detailed structure of the signature. A signature is represented as $\Sigma = (E_1, \mathcal{R}_\sigma)$, where E_1 denotes a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$, and $\mathcal{R}_\sigma = (\sigma(P_0), \sigma(Q_0), \deg(\sigma))$ encodes the image of a canonical torsion basis (P_0, Q_0) of $E_1[AC]$ under the isogeny σ, along with the degree of the isogeny. The size of the components of the signature is outlined as follows:

- **Representation of E_1:** The elliptic curve E_1 is uniquely determined by its j-invariant. For $j(E_1) = a + ib \in \mathbb{F}_{p^2}$, storing $j(E_1)$ requires approximately $2\log_2(p)$ bits.
- **Isogeny degree:** The degree of the signature isogeny σ satisfies $\deg(\sigma) = \deg(\tilde{\sigma}) \cdot \deg(w) \approx p^{3/4}$. Therefore, approximately $\frac{3}{4}\log_2(p)$ bits are required to store the degree.
- **Isogeny action on torsion basis:** The images of the torsion points (P_0, Q_0) under σ are given by

$$\sigma(P_0) = a_1 Q_1 + b_1 Q_2 \quad \text{and} \quad \sigma(Q_0) = a_2 Q_1 + b_2 Q_2,$$

  where $\langle Q_1, Q_2 \rangle = E_2[AC]$ is a canonical basis of the torsion subgroup, and $a_i, b_i \in \mathbb{Z}/AC\mathbb{Z}$ for $i = 1, 2$. Storing these four coefficients requires a total of $4\log_2(AC) = 2\log_2(p)$ bits.

Summing the contributions, the total signature size amounts to

$$2\log_2(p) + \frac{3}{4}\log_2(p) + 2\log_2(p) = \frac{19}{4}\log_2(p) \text{ bits.}$$

In our setting, this evaluates to approximately 1.26 KB. To support higher security levels, such as $\lambda \in \{192, 256\}$, one may adopt the parameter scaling strategy suggested by [34], where the number of small primes used in the construction of B is increased proportionally. In particular, it is reasonable to set $t = \lambda$, thereby ensuring that the calculation is made while maintaining a balance among the parameters similar to the 128-bit configuration.

Remark 3.1. *The pre-signature $\tilde{\Sigma}$ incorporates a zero-knowledge proof for the commitment isogeny. Although the pre-signature is inherently ephemeral, the size of the zero-knowledge proof remains an important consideration, primarily influenced by the underlying isogeny structure. For artificially oriented curves, the construction adapts the zero-knowledge proof for masked public keys from [36] to accommodate independently scaled points. While this adaptation preserves the desired security properties, its efficiency—particularly in terms of proof size and computational cost—remains an area for improvement. Investigating more compact encoding methods or alternative proof techniques may enhance the overall practicality and scalability of the scheme.*

4 Security Proof

In this section, we analyze and formally prove the security of the proposed adaptor signature scheme, denoted as $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$, as introduced in Algorithm 1. We demonstrate that $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$ satisfies the properties of pre-signature correctness, pre-signature adaptability, aEUF-CMA, and witness extractability. Verifying these properties is sufficient to prove Theorem 4.11.

Lemma 4.1. *The adaptor signature $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$, as presented in Algorithm 1, is pre-signature correct.*

Proof. First, let $(\mathsf{w},\mathsf{s}) := (w, I_w = (E_w, w(\mathfrak{B}), \pi_w)) \xleftarrow{\$} \mathsf{GenR}(1^\lambda)$ represent a fixed witness/statement pair for the defined hard relation $\mathsf{R}_{\mathfrak{A}}$. Here, w denotes an isogeny from E_0 to the target elliptic curve E_w, $w(\mathfrak{B})$ represents the image of the B-orientation $\mathfrak{B}$ under the witness isogeny w, and π_w is a zero-knowledge proof for the pair $(w,(E_w,(\mathfrak{B})))$. Additionally, let $(\mathsf{sk},\mathsf{pk}) := (\tau, E_\tau) \xleftarrow{\$} \mathsf{KeyGen}(1^\lambda)$ be a fixed secret/public key pair.
Assume that, for a message $m \in \{0,1\}^*$, the pre-signature $\tilde{\Sigma} = (E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}})$ is generated via the PreSig algorithm, i.e., $\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m, I_w)$. In this case, the verification algorithm yields $1 \leftarrow \mathsf{PreVer}(E_\tau, m, I_w, \tilde{\Sigma})$. This holds because: (1) $\mathcal{R}_{\tilde{\sigma}}$ is a correct efficient representation of an isogeny $\tilde{\sigma}$ from E_ψ to E_2, constructed using knowledge of $\mathrm{End}(E_0)$ and the isogenies τ, ψ, and φ; and (2) the isogeny $\varphi : E_\tau \to E_2$ depends on the message m and the j-invariant of the (second) commitment curve E_1. The curve E_1 is obtained by pushing forward the commitment isogeny ψ through the witness isogeny w, i.e., $[w]_*\psi : E_w \to E_1$. By the correctness of NIZK, we have $1 = \mathsf{NIZK.V}(E_1, \pi_{\psi'})$. Furthermore, for $S = (\psi(P), \psi(Q))$, the equality of the Weil pairings $e_C(\psi(P),\psi(Q)) = e_C(P,Q)^B$ holds.
Next, consider the (full) signature $\Sigma = (E_1, \mathcal{R}_\sigma)$ produced by the adaptation algorithm, i.e., $\Sigma \leftarrow \mathsf{Adapt}(\tilde{\Sigma}, w)$. The verification algorithm Ver of $\Sigma_{\mathsf{SQIsignHD}}$ returns $1 \leftarrow \mathsf{Ver}(E_\tau, m, \mathsf{Adapt}(\tilde{\Sigma}, w))$, since $\mathcal{R}_\sigma$ is an efficient representation of the signature isogeny

$$\sigma := \tilde{\sigma} \circ \widehat{[\psi]_* w} = \tilde{\sigma} \circ \hat{w}' : E_1 \to E_2,$$

where E_1 is derived by pushing forward the witness isogeny w through ψ using S, and E_2 is the codomain of isogeny φ induced by $j(E_1)$ and the message m. Using the pre-signature $\tilde{\Sigma} = (E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}})$ and the signature $\Sigma = (E_1, \mathcal{R}_\sigma)$, we can exploit the discrete logarithm algorithm $\mathcal{A}_{\mathsf{DLP}}$ and the SIDH attack $\mathcal{A}_{\mathsf{SIDH}}$ to extract the isogeny $w' : E_\psi \to E_1 = E_\psi/\langle \psi(P) + [\alpha]\psi(Q)\rangle$, as described in Section 3.6. The secret value α then suffices to construct the witness isogeny $w : E_0 \to E_w = E_0/\langle P + [\alpha]Q\rangle$. Consequently, $w \leftarrow \mathsf{Ext}(\Sigma, \tilde{\Sigma}, I_w)$ can be successfully executed to recover the secret witness isogeny w. Therefore, the adaptor signature $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$ satisfies the pre-signature correctness property. □

Lemma 4.2. *The adaptor signature* $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$*, as depicted in Algorithm 1, is pre-signature adaptable.*

Proof. Let us define a fixed witness/statement pair $(\mathsf{w},\mathsf{s}) := (w, I_w) \in \mathsf{R}_{\mathfrak{A}}$, a fixed public key $\mathsf{pk} = E_\tau$, a pre-signature $\tilde{\Sigma}$, and a message $m \in \{0,1\}^*$, as in Lemma 4.1.
We aim to prove that any verifiably valid (though not necessarily honestly generated) pre-signature $\tilde{\Sigma} = (E_1, \pi_{\psi'}, E_\psi, S, \mathcal{R}_{\tilde{\sigma}})$ that passes the PreVer procedure can be adapted into a valid (full) signature Σ.
Assuming $\mathsf{PreVer}(E_\tau, m, I_w, \tilde{\Sigma}) = 1$, it follows from the pre-verification procedure that $\mathsf{NIZK.V}(E_1, \pi_{\psi'}) = 1$, the equality $e_C(\psi(P), \psi(Q)) = e_C(P,Q)^B$ of Weil pairings holds, and $\mathcal{R}_{\tilde{\sigma}}$ represents an isogeny from E_ψ to E_2, where E_2 is the target curve of φ, derived from (the hash of) the message m and the j-invariant of commitment curve E_1. By the correctness property established in Lemma 4.1, and given the presence of the witness w corresponding to the statement I_w, the adaptation algorithm Adapt necessarily produces a full signature Σ by first computing the push-forward $w' = [\psi]_* w : E_\psi \to E_1$ using $S = (\psi(P), \psi(Q))$, and then computing the composition $\sigma = \tilde{\sigma} \circ \hat{w}' : E_1 \to E_2$ to produce the efficient representation $\mathcal{R}_\sigma$. Consequently, the verification algorithm Ver of $\Sigma_{\mathsf{SQIsignHD}}$ necessarily accepts the signature $\Sigma = (E_1, \mathcal{R}_\sigma)$, i.e., $1 \leftarrow \mathsf{Ver}(E_\tau, m, \mathsf{Adapt}(\tilde{\Sigma}, w))$. □

Lemma 4.3. *Assuming that the SQIsignHD signature scheme* $\Sigma_{\mathsf{SQIsignHD}}$ *is* SUF-CMA*-secure, that* $\mathsf{R}_{\mathfrak{A}}$ *is a hard relation, and that Problem 2.3 and Problem 2.4 are computationally hard, then the* SQIAsignHD *adaptor signature scheme* $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$*, as given in Algorithm 1, is* aEUF-CMA*-secure.*

Proof. We begin our proof by reducing the unforgeability of the $\mathsf{SQIAsignHD}$ adaptor signature scheme to the strong unforgeability of the SQIsignHD signature scheme. Specifically, we consider an adversary $\mathcal{A}$ who plays a series of games, starting with the $\mathsf{aSigForge}$ game as defined in Definition 2.12. We then construct a simulator $\mathcal{S}$ who plays the strong unforgeability experiment $\mathsf{StrongSigForge}$, as defined in Definition 2.8 for the SQIsignHD signature scheme. The simulator $\mathcal{S}$ leverages $\mathcal{A}$'s forgery in $\mathsf{aSigForge}$ to win its own experiment. In this setting, $\mathcal{S}$ has access to both the signing oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ and the random oracle $\mathcal{H}^{\mathsf{SQIsignHD}}$, which it uses to simulate oracle queries for $\mathcal{A}$: specifically, the random oracle $\mathcal{H}$, the signing queries $\mathcal{O}_S$, and the pre-signing queries $\mathcal{O}_{pS}$.

The primary challenges in simulating oracles arise when handling $\mathcal{O}_{pS}$ queries. Since $\mathcal{S}$ can only obtain full signatures from its signing oracle, it requires a method to transform these full signatures into pre-signatures suitable for $\mathcal{A}$. This transformation process presents two main difficulties: (1) $\mathcal{S}$ must learn the witness w corresponding to the statement I_w for which the pre-signature is to be generated, and (2) $\mathcal{S}$ must simulate the zero-knowledge proof $\pi_{\psi'}$ associated with a secret parallel isogeny ψ' of the commitment isogeny ψ, ensuring consistency in the randomness within the pre-signature.

More specifically, upon receiving a $\mathcal{O}_{pS}$ query from $\mathcal{A}$, which includes a message m and an instance $I_w = (E_w, w(\mathfrak{B}), \pi_w)$, the simulator $\mathcal{S}$ queries its signing

oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ to obtain a full signature on m. Furthermore, the simulator must learn a witness w such that $(w, I_w) \in \mathsf{R}_{\mathfrak{A}}$ in order to convert the full signature into a pre-signature for $\mathcal{A}$. To this end, we utilize the extractability property of the zero-knowledge proof π_w, which allows us to extract w and, in turn, transform the full signature into a valid pre-signature. Additionally, since a valid pre-signature includes a zero-knowledge proof $\pi_{\psi'}$, the simulator must simulate this proof without knowledge of the corresponding secret. To achieve this, we rely on the zero-knowledge property, which enables the simulation of a proof for a statement without requiring access to the associated witness.

Game_0. This game corresponds to the $\mathsf{aSigForge}$ experiment, as per Definition 2.12, where the adversary $\mathcal{A}$ has access to a random oracle $\mathcal{H}$ in the random oracle model, as well as many previously produced valid pre-signatures and signatures through the pre-signing oracle $\mathcal{O}_{pS}$ and the signing oracle $\mathcal{O}_S$, respectively, for messages of its choice, except for a message m. The adversary then attempts to forge a verifiable signature Σ^* on m. Since we are working within the random oracle model, we explicitly write the random oracle code $\mathcal{H}$ via $\mathcal{H}^{\mathsf{SQIsignHD}}$. Thus, it follows that

$$\Pr[\mathsf{Game}_0 = 1] = \Pr[\mathsf{aSigForge}_{\mathcal{A},\Xi_{\mathsf{R}_{\mathfrak{A}}},\Sigma_{\mathsf{SQIsignHD}}}(\lambda) = 1].$$

Game_0	$\mathcal{H}(x)$
$\mathcal{Q} := \emptyset$	**if** $H[x] = \perp$
$H := [\perp]$	$H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$	**return** $H[x]$
$m \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(E_\tau)$	$\mathcal{O}_{pS}(m, I_w)$
$(w, I_w) \leftarrow \mathsf{GenR}(1^\lambda)$	$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m, I_w)$
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m, I_w)$	$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(\tilde{\Sigma}, I_w)$	**return** $\tilde{\Sigma}$
$b := \mathsf{Ver}(E_\tau, m, \Sigma^*)$	$\mathcal{O}_S(m)$
return $m \notin \mathcal{Q} \wedge b$	$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
	$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
	return Σ

Game_1	$\mathcal{H}(x)$
$\mathcal{Q} := \emptyset$	**if** $H[x] = \perp$
$H := [\perp]$	$H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$	**return** $H[x]$
$m^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(E_\tau)$	$\mathcal{O}_{pS}(m, I_w)$
$(w, I_w) \leftarrow \mathsf{GenR}(1^\lambda)$	$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m, I_w)$
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I_w)$	$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(\tilde{\Sigma}, I_w)$	**return** $\tilde{\Sigma}$
if $\mathsf{Adapt}(\tilde{\Sigma}, w) = \Sigma^*$	$\mathcal{O}_S(m)$
abort	$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$b := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$	$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $m^* \notin \mathcal{Q} \wedge b$	**return** Σ

Game_1. This game is analogous to Game_0, with the only difference being that if the valid signature Σ^*, forged by the adversary $\mathcal{A}$, matches the result of adapting the pre-signature into a signature using the corresponding witness w, then the game aborts.

Claim 4.4. *If* Bad_1 *is the event that* Game_1 *aborts, then we claim that for a negligible function* negl *in* λ, $Pr[\mathsf{Bad}_1] \leq \mathsf{negl}(\lambda)$.

Proof. We prove this claim by reducing it to the hardness of the relation $\mathsf{R}_{\mathfrak{Q}}$. To do this, we construct a simulator $\mathcal{S}$ that breaks the hardness of $\mathsf{R}_{\mathfrak{Q}}$ under the assumption that it has access to an adversary $\mathcal{A}$ that causes Game_1 to abort with non-negligible probability. The simulator receives a challenge $\mathsf{s}^* := I^*_{w^*}$, upon which it generates a secret/public key pair $(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$ to simulate $\mathcal{A}$'s queries to the oracles $\mathcal{H}$, $\mathcal{O}_{pS}$ and $\mathcal{O}_S$. The simulation of the oracles proceeds as described in Game_1.

Upon receiving the challenge message m^* from $\mathcal{A}$, $\mathcal{S}$ computes a pre-signature $\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I^*_{w^*})$ and returns the pair $(\tilde{\Sigma}, I^*_{w^*})$ to the adversary, who forges a signature using the returned pair. Assuming that Bad_1 occurred (i.e., $\mathsf{Adapt}(\tilde{\Sigma}, w) = \Sigma^*$). Since the $\Xi_{\mathsf{R}_{\mathfrak{Q}},\Sigma_{\mathsf{SQIsignHD}}}$ is pre-signature correct by Lemma 4.1, the simulator can extract w^* via $\mathsf{Ext}(\Sigma^*, \tilde{\Sigma}, I^*_{w^*})$ to obtain a valid witness/statement pair such that $(w^*, I^*_{w^*}) \in \mathsf{R}_{\mathfrak{Q}}$. In this way, $\mathcal{S}$ breaks the security of the relation $\mathsf{R}_{\mathfrak{Q}}$.

We note that the view of $\mathcal{A}$ is indistinguishable from its view in Game_1, since the challenge $I^*_{w^*}$ is an instance of the hard relation $\mathsf{R}_{\mathfrak{Q}}$ and follows the same distribution as the public output of GenR. Therefore, the probability that $\mathcal{S}$ breaks the hardness of $\mathsf{R}_{\mathfrak{Q}}$ is equal to the probability that the event Bad_1 occurring, which is non-negligible by assumption. This contradicts the hardness of $\mathsf{R}_{\mathfrak{Q}}$. □

Since Game_1 and Game_0 are equivalent except when the event Bad_1 occurs, it follows that

$$\Pr[\mathsf{Game}_1 = 1] \leq \Pr[\mathsf{Game}_0 = 1] + \mathsf{negl}(\lambda).$$

Game_2. This game is similar to the previous game, with the only difference being a modification in the pre-signing oracle $\mathcal{O}_{pS}$. Specifically, in this game, we apply the extractor algorithm $\mathcal{E}$, taking the statement $(E_w, w(\mathfrak{B}))$, the proof π_w, and the list of random oracle queries H as input to extract a witness w. The game aborts if $(w, (E_w, w(\mathfrak{B}), \pi_w)) \notin \mathsf{R}_{\mathfrak{A}}$.

Claim 4.5. *If* Bad_2 *is the event that* Game_2 *aborts during an* $\mathcal{O}_{pS}$ *execution, then it holds that* $Pr[\mathsf{Bad}_2] \leq \mathsf{negl}(\lambda)$*, for a negligible function* negl *in* λ*.*

Proof. By the online extractor property of the NIZK, for a witness w extracted from a proof π_w of the statement $(E_w, w(\mathfrak{B}))$ such that $\mathsf{NIZK.V}(E_w, w(\mathfrak{B}), \pi_w) = 1$, it follows that $(w, I_w) \in \mathsf{R}_{\mathfrak{A}}$, except with negligible probability in the security parameter λ. □

Therefore, since games Game_2 and Game_1 are equivalent except in case event Bad_2 happens, it follows that

$$Pr[\mathsf{Game}_2 = 1] \leq Pr[\mathsf{Game}_1 = 1] + \mathsf{negl}(\lambda).$$

Game_2

$\mathcal{Q} := \emptyset$
$H := [\bot]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$m^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(E_\tau)$
$(w, I_w) \leftarrow \mathsf{GenR}(1^\lambda)$
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I_w)$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\Sigma}, I_w)$
if $\mathsf{Adapt}(\tilde{\Sigma}, w) = \Sigma^*$
  abort
$b := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
return $m^* \notin \mathcal{Q} \wedge b$

$\mathcal{H}(x)$

if $H[x] = \bot$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m, I_w)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma}$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

Game_3

$\mathcal{Q} := \emptyset$
$H := [\perp]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$m^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(E_\tau)$
$(w, I_w) \leftarrow \mathsf{GenR}(1^\lambda)$
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I_w)$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(\tilde{\Sigma}, I_w)$
if $\mathsf{Adapt}(\tilde{\Sigma}, w) = \Sigma^*$
  abort
$b := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
return $m^* \notin \mathcal{Q} \wedge b$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

$\mathcal{H}(x)$

if $H[x] = \perp$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$

Game_3. This game extends the modifications to the pre-signing oracle $\mathcal{O}_{pS}$ introduced in the previous game. Specifically, it begins by generating a valid full signature $\Sigma = (E_1, \mathcal{R}_\sigma)$ through the execution of the Sig algorithm. Using the $\mathcal{A}_{\mathsf{SIDH}}$ algorithm, the isogeny σ, represented by $\mathcal{R}_\sigma$, is decomposed into $\sigma = \tilde{\sigma} \circ \hat{w}^*$, where $\hat{w}^*$ is a C-isogeny from E_1 to a curve E^*_ψ, and $\tilde{\sigma}$ is an isogeny from E^*_ψ to E_2. Subsequently, the efficient representation $\mathcal{R}_{\tilde{\sigma}}$ corresponding to the isogeny $\tilde{\sigma}$ is computed.

Now, in order to construct a C-torsion basis $\langle P^*, Q^* \rangle$ for $E^*_\psi[C]$ for which $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$, where α is the secret obtained from the extracted witness $w : E_0 \to E_0/\langle P + [\alpha]Q \rangle$, and w^* is the dual of $\hat{w}^*$, let $\ker(w^*) = R$. First, we select a point R' that is linearly independent of R to form a C-torsion basis for E^*_ψ, i.e., $\langle R, R' \rangle = E^*_\psi[C]$. Now, we seek values x_1, y_1, x_2, and y_2 such that

$$(x_1 R + y_1 R') + \alpha(x_2 R + y_2 R') = R.$$

This condition implies:

$$x_1 + \alpha x_2 = 1 \quad \text{and} \quad y_1 + \alpha y_2 = 0.$$

Finding a single solution for $(x_1, x_2), (y_1, y_2) \in (\mathbb{Z}/C\mathbb{Z}) \times (\mathbb{Z}/C\mathbb{Z})$, where $(x_i, y_i) \neq (0, 0)$ for $i = 1, 2$, is sufficient to determine the pair $S^* = (P^*, Q^*)$ by

setting:

$$P^* := x_1 R + y_1 R' \quad \text{and} \quad Q^* := x_2 R + y_2 R'.$$

Finally, before forming the pre-signature, the $\mathcal{S}$ simulates a proof $\pi^*_{\psi'}$ for the statement E_1 without any knowledge of the corresponding secret isogeny ψ'. The pre-signature is then defined as $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$. We see that this game is indistinguishable from the previous one, and it follows that

$$Pr[\mathsf{Game}_3 = 1] \leq Pr[\mathsf{Game}_2 = 1] + \mathsf{negl}(\lambda).$$

Game_4

$\mathcal{Q} := \emptyset$
$H := [\bot]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$m^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(E_\tau)$
$(w, I_w) \leftarrow \mathsf{GenR}(1^\lambda)$
$\Sigma \leftarrow \mathsf{Sig}(\tau, m^*)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\Sigma}, I_w)$
if $\mathsf{Adapt}(\tilde{\Sigma}, w) = \Sigma^*$
  abort
$b := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
return $m^* \notin \mathcal{Q} \wedge b$

$\mathcal{H}(x)$

if $H[x] = \bot$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract$(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

Game_4. In this game, upon receiving the challenge message m^* from $\mathcal{A}$, the game itself generates a signature Σ by running the Sig algorithm and converting the resulting signature into a valid pre-signature, as in the previous game during the $\mathcal{O}_{pS}$ execution. Consequently, the same indistinguishability argument as in

the previous game also holds. Therefore, it follows that

$$Pr[\mathsf{Game}_4 = 1] \leq Pr[\mathsf{Game}_3 = 1] + \mathsf{negl}(\lambda).$$

After establishing that the transition from the original aSigForge game (Game_0) to Game_4 is indistinguishable, it remains to demonstrate the existence of a simulator that perfectly emulates Game_4 and leverages $\mathcal{A}$ to succeed in the StrongSigForge game. Below, we provide a concise description of how the simulator responds to Oracle queries.

Simulation of Oracle Queries:

Signing queries. If the adversary $\mathcal{A}$ queries the signing oracle $\mathcal{O}_S$ on input m, $\mathcal{S}$ sends m to its oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ and forwards its response to $\mathcal{A}$.
Random Oracle queries. Based on $\mathcal{A}$ querying the oracle $\mathcal{H}$ on input x, in case $H[x] = \perp$, then $\mathcal{S}$ queries $\mathcal{H}^{\mathsf{SQIsignHD}}(x)$, otherwise the simulator outputs $H[x]$.
Pre-Signing queries. If $\mathcal{A}$ queries the pre-signing oracle $\mathcal{O}_{pS}$ on input (m, I_w):

1. The simulator extracts the witness isogeny w using the extractability property of NIZK. It then forwards the message m to the oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ and parses the generated signature Σ as $(E_1, \mathcal{R}_\sigma)$.

2. The simulator $\mathcal{S}$ constructs a pre-signature isogeny representation $\mathcal{R}_{\tilde{\sigma}}$, and torsion basis $S^* = (P^*, Q^*)$ by decomposing σ into $\tilde{\sigma} \circ \hat{w}^*$ using the algorithm $\mathcal{A}_{\mathsf{SIDH}}$, and the α which is obtained from the extracted witness $w : E_0 \to E_0/\langle P + [\alpha]Q \rangle$ via the online extractor property, respectively.

3. Finally, $\mathcal{S}$ simulates a zero-knowledge proof $\pi^*_{\psi'}$, for the statement E_1. The simulator outputs $\tilde{\Sigma} = (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$.

Challenge phase:

1. When $\mathcal{A}$ outputs the message m^* as the challenge message, $\mathcal{S}$ generates $(w, I_w) \leftarrow \mathsf{GenR}(1^\lambda)$, forwards m^* to the oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$, and parses the generated signature Σ as $(E_1, \mathcal{R}_\sigma)$.

2. $\mathcal{S}$ generates the required pre-signature $\tilde{\Sigma}$ in the same manner as it does during $\mathcal{O}_{pS}$ queries.

3. When $\mathcal{A}$ outputs a forgery Σ^*, the simulator outputs (m^*, Σ^*) as its own forgery.

We highlight that the primary difference between the simulation and Game_4 is syntactical. Specifically, rather than generating the secret/public keys and executing the algorithms Sig and $\mathcal{H}$, the simulator $\mathcal{S}$ utilizes its oracles $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ and $\mathcal{H}^{\mathsf{SQIsignHD}}$. It remains to demonstrate that the forgery produced by $\mathcal{A}$ can be used by the simulator to win the StrongSigForge game.

Claim 4.6. (m^*, Σ^*) *constitutes a valid forgery in the* StrongSigForge *game.*

Proof. To prove this claim, we must show that the pair (m^*, Σ^*) has not been previously output by the oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$. Note that the adversary $\mathcal{A}$ has not made a query on the challenge message m^* to either $\mathcal{O}_S$ or $\mathcal{O}_{pS}$. Therefore, $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ is only queried on m^* during the challenge phase. As demonstrated in the game Game_1, the adversary produces a forgery Σ^*, which matches the signature Σ output by $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ during the challenge phase with only negligible probability. Consequently, the oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ has never output Σ^* on the query m^* before, establishing that (m^*, Σ^*) is a valid forgery for the $\mathsf{StrongSigForge}$ game. □

$\mathcal{S}^{\mathsf{Sig}^{\mathsf{SQIsignHD}}, \mathcal{H}^{\mathsf{SQIsignHD}}}(E_\tau)$

$\mathcal{Q} := \emptyset$
$H := [\perp]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$m^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(E_\tau)$
$(w, I_w) \leftarrow \mathsf{GenR}(1^\lambda)$
$\Sigma \leftarrow \mathsf{Sig}^{\mathsf{SQIsignHD}}(m^*)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract$(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\Sigma}, I_w)$
return (m^*, Σ^*)

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}^{\mathsf{SQIsignHD}}(m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

$\mathcal{H}(x)$

if $H[x] = \perp$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract$(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$

From the game Game_0 to the game Game_4, we have that

$$\Pr[\mathsf{Game}_0 = 1] \leq \Pr[\mathsf{Game}_4 = 1] + \mathsf{negl}(\lambda).$$

Due to a perfect simulation of Game_4, provided by the simulator $\mathcal{S}$, it follows

that

$$\begin{aligned}\mathsf{Adv}_{\mathcal{A}}^{\mathsf{aSigForge}} &= \Pr[\mathsf{Game}_0 = 1] \\ &\leq \Pr[\mathsf{Game}_4 = 1] + \mathsf{negl}(\lambda) \\ &\leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{StrongSigForge}} + \mathsf{negl}(\lambda).\end{aligned}$$

By assumption, as SQIsignHD is secure in ROM with $\mathcal{H}^{\mathsf{SQIsignHD}}$ programmed as a random oracle, it implies that our adaptor signature, $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$, is aEUF-CMA secure in ROM. This completes the proof of Lemma 4.3. □

Lemma 4.7. *Assuming that the SQIsignHD signature scheme* $\Sigma_{\mathsf{SQIsignHD}}$ *is* SUF-CMA*-secure, that* $\mathsf{R}_{\mathfrak{A}}$ *is a hard relation, and that Problem 2.3 and Problem 2.4 are computationally hard, then the* SQIAsignHD *adaptor signature scheme* $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$*, as given in Algorithm 1, is witness extractable.*

Proof. We begin by outlining the primary intuition behind the proof of witness extractability. The proof of this lemma closely follows the proof of Lemma 4.3. Specifically, we prove this lemma by reducing the witness extractability of $\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}$ to the strong unforgeability of the SQIsignHD signature scheme, $\Sigma_{\mathsf{SQIsignHD}}$. To do so, let $\mathcal{A}$ be a PPT adversary that wins the aWitExt game. We then construct another PPT adversary, $\mathcal{S}$, so that it wins the StrongSigForge game.

Analogous to the proof of Lemma 4.3, the primary challenge lies in simulating the pre-signing queries. Consequently, the simulation process is carried out exactly as in the proof of Lemma 4.3.

The key distinction in this case, however, is that in the aWitExt game, the adversary $\mathcal{A}$ outputs the statement I_w for the relation $\mathsf{R}_{\mathfrak{A}}$ along with the challenge message m^*. This implies that the pair (w, I_w) is not predetermined by the game. As a result, $\mathcal{S}$ cannot convert a valid signature into a pre-signature since it does not have access to the witness w. However, w can be extracted from the zero-knowledge proof embedded in I_w. Once w is extracted, then S can simulate the pre-signing queries as in Lemma 4.3. We, now, begin with designing a series of games required for the proof.

Game_0. This game corresponds to the original aWitExt game, as per Definition 2.13, where the adversary $\mathcal{A}$ must produce a valid signature Σ for a message m of its choice, given a pre-signature $\tilde{\Sigma}$ and a witness/statement pair (w, I_w), while having access to the oracles $\mathcal{H}$, $\mathcal{O}_{pS}$ and $\mathcal{O}_S$. The adversary $\mathcal{A}$ succeeds if $(\mathsf{Ext}(\tilde{\Sigma}, \Sigma, I_w), I_w) \notin \mathsf{R}_{\mathfrak{A}}$. Since we are in the random oracle model, we explicitly write the random oracle code $\mathcal{H}$. It then trivially follows that:

$$\Pr[\mathsf{Game}_0 = 1] = \Pr[\mathsf{aWitExt}_{\mathcal{A},\Xi_{\mathsf{R}_{\mathfrak{A}},\Sigma_{\mathsf{SQIsignHD}}}}(\lambda) = 1].$$

Game_0

$\mathcal{Q} := \emptyset$
$H := [\bot]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$(m^*, I_w) \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(E_\tau)$
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I_w)$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(\tilde{\Sigma})$
$w^* := \mathsf{Ext}(\tilde{\Sigma}, \Sigma^*, I_w)$
$b_1 := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
$b_2 := m^* \notin \mathcal{Q}$
$b_3 := (w^*, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
return $b_1 \wedge b_2 \wedge b_3$

$\mathcal{H}(x)$

if $H[x] = \bot$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m, I_w)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma}$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

Game_1. This game is the same as Game_0, except that some changes are applied to the pre-signing oracle $\mathcal{O}_{pS}$. More specifically, during the $\mathcal{O}_{pS}$ queries, this game extracts a witness w by executing the online extractor algorithm $\mathcal{E}$ on the inputs: the statement $(E_w, w(\mathfrak{B}))$, the proof π_w, and the list of random oracle queries H. The game aborts if the extracted witness w does not satisfy $(w, I_w) \in \mathsf{R}_{\mathfrak{A}}$.

Game_1

$\mathcal{Q} := \emptyset$
$H := [\bot]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$(m^*, I_w) \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(E_\tau)$
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I_w)$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(\tilde{\Sigma})$
$w^* := \mathsf{Ext}(\tilde{\Sigma}, \Sigma^*, I_w)$
$b_1 := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
$b_2 := m^* \notin \mathcal{Q}$
$b_3 := (w^*, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
return $b_1 \wedge b_2 \wedge b_3$

$\mathcal{H}(x)$

if $H[x] = \bot$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m, I_w)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma}$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

Claim 4.8. *If* Bad_1 *is the event that* Game_1 *aborts while the execution of* $\mathcal{O}_{pS}$*, then it holds that* $Pr[\mathsf{Bad}_1] \leq \mathsf{negl}(\lambda)$.

Proof. By the online extractor property of NIZK, if a witness w is extracted from a proof π_w for the statement $(E_w, w(\mathfrak{B}))$ such that $\mathsf{NIZK.V}((E_w, w(\mathfrak{B})), \pi_w) = 1$, it follows that $(w, I_w) \in \mathsf{R}_{\mathfrak{A}}$, except with negligible probability. □

It follows that Game_1 and Game_0 are equivalent, except when the event Bad_1 occurs. Thus, we have:

$$\Pr[\mathsf{Game}_0 = 1] \leq \Pr[\mathsf{Game}_1 = 1] + \mathsf{negl}(\lambda).$$

Game_2

$\mathcal{Q} := \emptyset$
$H := [\bot]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$(m^*, I_w) \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(E_\tau)$
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I_w)$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\Sigma})$
$w^* := \mathsf{Ext}(\tilde{\Sigma}, \Sigma^*, I_w)$
$b_1 := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
$b_2 := m^* \notin \mathcal{Q}$
$b_3 := (w^*, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
return $b_1 \wedge b_2 \wedge b_3$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

$\mathcal{H}(x)$

if $H[x] = \bot$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$

Game_2. This game extends the modifications to the pre-signing oracle $\mathcal{O}_{pS}$ from the previous game. It generates a valid signature $\Sigma = (E_1, \mathcal{R}_\sigma)$ using the Sig algorithm and decomposes the isogeny σ into $\sigma = \tilde{\sigma} \circ \hat{w}^*$ using $\mathcal{A}_{\mathsf{SIDH}}$. Here, $\hat{w}^*$ is a C-isogeny from E_1 to a curve E^*_ψ, and $\tilde{\sigma}$ is an isogeny from E^*_ψ to E_2. Thereby, the efficient representation $\mathcal{R}_{\tilde{\sigma}}$ for $\tilde{\sigma}$ is computed.

To construct a C-torsion basis $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$ with α derived from the witness w, a basis $\langle R, R' \rangle = E^*_\psi[C]$ is formed by selecting a point R' linearly independent of $R = \ker(w^*)$. The coefficients x_1, y_1, x_2, y_2 are determined, where $(x_1, x_2), (y_1, y_2) \in (\mathbb{Z}/C\mathbb{Z}) \times (\mathbb{Z}/C\mathbb{Z})$ and $(x_i, y_i) \neq (0, 0)$ for $i = 1, 2$, to satisfy:

$$(x_1R + y_1R') + \alpha(x_2R + y_2R') = R.$$

These coefficients define the C-torsion basis for E^*_ψ by setting $P^* = x_1R + y_1R'$ and $Q^* = x_2R + y_2R'$.

Finally, the simulator $\mathcal{S}$ generates a proof $\pi^*_{\psi'}$ for E_1 without knowledge of the isogeny ψ', that is computationally indistinguishable from an honest proof, and the pre-signature is constructed as $\tilde{\Sigma} = (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$. The game remains indistinguishable from the previous one, ensuring:

$$\Pr[\mathsf{Game}_1 = 1] \leq \Pr[\mathsf{Game}_2 = 1] + \mathsf{negl}(\lambda).$$

Game_3

$\mathcal{Q} := \emptyset$
$H := [\perp]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$(m^*, I_w) \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(E_\tau)$
Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\tilde{\Sigma} \leftarrow \mathsf{PreSig}(\tau, m^*, I_w)$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\Sigma})$
$w^* := \mathsf{Ext}(\tilde{\Sigma}, \Sigma^*, I_w)$
$b_1 := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
$b_2 := m^* \notin \mathcal{Q}$
$b_3 := (w^*, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
return $b_1 \wedge b_2 \wedge b_3$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

$\mathcal{H}(x)$

if $H[x] = \perp$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$

Game_3. In this game, for the challenge phase, we apply the identical modifications implemented in Game_1's $\mathcal{O}_{pS}$ oracle. In the challenge phase, a witness w is extracted by the online extractor algorithm $\mathcal{E}$ taking the statement $(E_w, w(\mathfrak{B}))$, the proof π_w, and the list of random oracle queries H as inputs. In case for the extracted witness w, the relation $(w, I_w) \in \mathsf{R}_{\mathfrak{A}}$ is not satisfied, then the game aborts.

Claim 4.9. *If* Bad_2 *is the event that* Game_3 *aborts during the challenge phase, then it follows* $Pr[\mathsf{Bad}_2] \leq \mathsf{negl}(\lambda)$.

Proof. The arguments presented in the proof of Claim 4.8 apply similarly to prove this claim. □

Hence, Game_3 and Game_2 are equivalent except for the case that the event Bad_2 happens. Thus, we have

$$\Pr[\mathsf{Game}_2 = 1] \leq \Pr[\mathsf{Game}_3 = 1] + \mathsf{negl}(\lambda).$$

Game_4

$\mathcal{Q} := \emptyset$
$H := [\bot]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$(m^*, I_w) \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(E_\tau)$
Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H}, \mathcal{O}_S, \mathcal{O}_{pS}}(\tilde{\Sigma})$
$w^* := \mathsf{Ext}(\tilde{\Sigma}, \Sigma^*, I_w)$
$b_1 := \mathsf{Ver}(E_\tau, m^*, \Sigma^*)$
$b_2 := m^* \notin \mathcal{Q}$
$b_3 := (w^*, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
return $b_1 \wedge b_2 \wedge b_3$

$\mathcal{H}(x)$

if $H[x] = \bot$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}(\tau, m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

Game_4. In this game, the challenge phase employs similar modifications implemented in Game_2 for the $\mathcal{O}_{pS}$ oracle. Specifically, the game begins by generating a valid full signature Σ using the Sig algorithm and subsequently converts Σ into a pre-signature with the help of the extracted witness w and $\mathcal{A}_{\mathsf{SIDH}}$. Additionally, the game computes the zero-knowledge proof in the same manner as described in Game_2. Consequently, the same arguments apply here as well.

Thus, this game is indistinguishable from the previous one, and it follows that

$$\Pr[\mathsf{Game}_3 = 1] \leq \Pr[\mathsf{Game}_4 = 1] + \mathsf{negl}(\lambda).$$

Having established that the transformation of the original aWitExt game into Game_4 is indistinguishable, it remains to demonstrate the existence of a simulator that perfectly simulates Game_4 while leveraging the adversary $\mathcal{A}$ to win the StrongSigForge game. Below, we provide a concise description of the simulator's implementation.

$\mathcal{S}^{\mathsf{Sig}^{\mathsf{SQIsignHD}},\mathcal{H}^{\mathsf{SQIsignHD}}}(E_\tau)$

$\mathcal{Q} := \emptyset$
$H := [\bot]$
$(\tau, E_\tau) \leftarrow \mathsf{KeyGen}(1^\lambda)$
$(m^*, I_w) \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(E_\tau)$
Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}^{\mathsf{SQIsignHD}}(m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$
$\Sigma^* \leftarrow \mathcal{A}^{\mathcal{H},\mathcal{O}_S,\mathcal{O}_{pS}}(\tilde{\Sigma})$
return (m^*, Σ^*)

$\mathcal{H}(x)$

if $H[x] = \bot$
  $H[x] \leftarrow \mathcal{H}^{\mathsf{SQIsignHD}}(x)$
return $H[x]$

$\mathcal{O}_{pS}(m, I_w)$

Parse I_w as $(E_w, w(\mathfrak{B}), \pi_w)$
$w := \mathcal{E}(E_w, w(\mathfrak{B}), \pi_w, H)$
if $(w, I_w) \notin \mathsf{R}_{\mathfrak{A}}$
  abort
$\Sigma \leftarrow \mathsf{Sig}^{\mathsf{SQIsignHD}}(m)$
Parse Σ as $(E_1, \mathcal{R}_\sigma)$
Extract $(E^*_\psi, \mathcal{R}^*_{\tilde{\sigma}})$ by $\mathcal{A}_{\mathsf{SIDH}}$ s.t.
  $\sigma = \tilde{\sigma} \circ \hat{w}^*$ with $\deg(\hat{w}^*) = C$
Extract α from
  $\ker(w) = \langle P + [\alpha]Q \rangle$
Find $\langle P^*, Q^* \rangle = E^*_\psi[C]$ for which
  $\ker(w^*) = \langle P^* + [\alpha]Q^* \rangle$
Set $S^* := (P^*, Q^*)$
$\pi^*_{\psi'} \leftarrow \mathcal{S}((E_w, E_1), 1)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\tilde{\Sigma} := (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$

$\mathcal{O}_S(m)$

$\Sigma \leftarrow \mathsf{Sig}^{\mathsf{SQIsignHD}}(m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return Σ

Simulation of Oracle Queries:

Signing queries. If the adversary $\mathcal{A}$ queries the signing oracle $\mathcal{O}_S$ with input m, the simulator $\mathcal{S}$ forwards m to its oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ and then sends the response back to $\mathcal{A}$.

Random Oracle queries. If $\mathcal{A}$ queries the oracle $\mathcal{H}$ with input x, and if $H[x] = \perp$, the simulator $\mathcal{S}$ queries $\mathcal{H}^{\mathsf{SQIsignHD}}(x)$. Otherwise, it returns $H[x]$.
Pre-Signing queries. When $\mathcal{A}$ submits a query (m, I_w) to the pre-signing oracle $\mathcal{O}_{pS}$,

1. The simulator uses the extractability property of NIZK to extract the witness isogeny w. It then sends the message m to the oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ and parses the resulting signature Σ as $(E_1, \mathcal{R}_\sigma)$.
2. The simulator $\mathcal{S}$ constructs the pre-signature isogeny representation $\mathcal{R}_{\tilde{\sigma}}$ and the torsion basis $S^* = (P^*, Q^*)$ by decomposing σ into $\tilde{\sigma} \circ \hat{w}^*$ using the algorithm $\mathcal{A}_{\mathsf{SIDH}}$, and by extracting the value α from the witness $w : E_0 \to E_0/\langle P + [\alpha]Q \rangle$ obtained from the online extractor property.
3. The simulator $\mathcal{S}$ generates a zero-knowledge proof $\pi^*_{\psi'}$ for the statement E_1. It then outputs the pre-signature $\tilde{\Sigma} = (E_1, \pi^*_{\psi'}, E^*_\psi, S^*, \mathcal{R}^*_{\tilde{\sigma}})$.

Challenge phase:

1. When $\mathcal{A}$ outputs the challenge message (m^*, I_w), the simulator $\mathcal{S}$ extracts w using the extractability property of NIZK, sends m^* to the oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$, and parses the generated signature as $\Sigma = (E_1, \mathcal{R}_\sigma)$.
2. $\mathcal{S}$ constructs the required pre-signature $\tilde{\Sigma}$ in the same way it does for $\mathcal{O}_{pS}$ queries.
3. When $\mathcal{A}$ produces a forgery Σ^*, the simulator returns (m^*, Σ^*) as its own forgery.

The key distinction between the simulation and Game_4 is syntactical. Instead of generating the secret/public keys and executing the algorithms Sig and $\mathcal{H}$, the simulator $\mathcal{S}$ relies on its oracles $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ and $\mathcal{H}^{\mathsf{SQIsignHD}}$. It remains to show that the forgery produced by $\mathcal{A}$ can be used by the simulator to win the $\mathsf{StrongSigForge}$ game.

Claim 4.10. (m^*, Σ^*) *constitutes a valid forgery in the* $\mathsf{StrongSigForge}$ *game.*

Proof. It suffices to demonstrate that the pair (m^*, Σ^*) has not been previously output by the oracle $\mathsf{Sig}^{\mathsf{SQIsignHD}}$. Note that neither $\mathcal{O}_{pS}$ nor $\mathcal{O}_S$ has received a query from adversary $\mathcal{A}$ on the challenge message m^*. Consequently, $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ is queried on m^* only during the challenge phase. If adversary $\mathcal{A}$ produces a forgery Σ^* that matches the signature Σ generated by $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ during the challenge phase, the extracted w would satisfy the relation with the corresponding statement I_w. Therefore, $\mathsf{Sig}^{\mathsf{SQIsignHD}}$ has never previously output Σ^* on query m^*. Thus, (m^*, Σ^*) constitutes a valid forgery in the $\mathsf{StrongSigForge}$ game. □

From Game_0 to Game_4, we have

$$\Pr[\mathsf{Game}_0 = 1] \leq \Pr[\mathsf{Game}_4 = 1] + \mathsf{negl}(\lambda).$$

Since $\mathcal{S}$ perfectly simulates Game_4, it follows that we obtain:

$$\begin{aligned}\mathsf{Adv}_{\mathcal{A}}^{\mathsf{aWitExt}} &= \Pr[\mathsf{Game}_0 = 1] \\ &\leq \Pr[\mathsf{Game}_4 = 1] + \mathsf{negl}(\lambda) \\ &\leq \mathsf{Adv}_{\mathcal{S}}^{\mathsf{StrongSigForge}} + \mathsf{negl}(\lambda).\end{aligned}$$

Since SQIsignHD is secure in the random oracle model with $\mathcal{H}^{\mathsf{SQIsignHD}}$ modeled as a random oracle, it follows that the adaptor signature scheme $\Xi_{\mathsf{R}_{\mathfrak{R}},\Sigma_{\mathsf{SQIsignHD}}}$ achieves witness extractability in the random oracle model. This completes the proof of Lemma 4.7. □

Theorem 4.11. *If the SQIsignHD signature scheme, $\Sigma_{\mathsf{SQIsignHD}}$, is* SUF-CMA*-secure, $\mathsf{R}_{\mathfrak{R}}$ is a hard relation, Problem 2.3 and Problem 2.4 are computationally hard, then the* SQIAsignHD *adaptor signature scheme, $\Xi_{\mathsf{R}_{\mathfrak{R}},\Sigma_{\mathsf{SQIsignHD}}}$, introduced in Algorithm 1, is secure in the random oracle model.*

Proof. By Lemmas 4.1, 4.2, 4.3, and 4.7, we have demonstrated that the adaptor signature scheme $\Xi_{\mathsf{R}_{\mathfrak{R}},\Sigma_{\mathsf{SQIsignHD}}}$ satisfies the properties of pre-signature correctness, pre-signature adaptability, aEUF-CMA security, and witness extractability. The verification of these properties completes the proof of Theorem 4.11. □

Conclusion

Adaptor signatures, an extension of standard digital signatures, are a vital cryptographic primitive for blockchain applications, helping to reduce costs, enhance fungibility, and support off-chain payments within payment-channel networks and hubs. In the present work, we have introduced SQIAsignHD, a new adaptor signature construction with quantum-resistant security based on isogenies of supersingular elliptic curves. Thereby, it provides security and privacy concepts relevant to off-chain applications. In SQIAsignHD, we use SQIsignHD as the underlying signature scheme and make use of the idea of artificial orientation, on the supersingular isogeny Diffie-Hellman key exchange protocol (SIDH), to apply the hard relation. We also exploit the SIDH attacks as a generic algorithm in recovering the secret witness isogeny in the extraction phase of our scheme. The signature in SQIAsignHD is approximately 1.26 KB in size for $\lambda = 128$ security level. In contrast to the only isogeny-based adaptor signature construction, IAS, which operates on a maximum of the CSIDH-512 parameters, our scheme scales well to high-security levels. Thus, compared to IAS, SQIAsignHD significantly improves the security level and signature size. Providing a concrete and optimized implementation of SQIAsignHD is left for future work.

References

[1] Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2008)

[2] Poelstra, A.: Scriptless scripts. https://tinyurl.com/ludcxyz (2017)

[3] Aumayr, L., Ersoy, O., Erwig, A., Faust, S., Hostakova, K., Maffei, M., Moreno- Sanchez, P., Riahi, S.: Generalized bitcoin-compatible channels (2020)

[4] Fournier, L.: One-time verifiably encrypted signatures aka adaptor signatures (2019)

[5] Malavolta, G., Moreno-Sanchez, P., Schneidewind, C., Kate, A., Maffei, M.: Anonymous Multi-hop Locks for Blockchain Scalability and Interoperability. Cryptology ePrint Archive (2018)

[6] Moreno-Sanchez, P., Blue, A., Le, D.V., Noether, S., Goodell, B., Kate, A.: DLSAG: Non-interactive refund transactions for interoperable payment channels in Monero. In: Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24, pp. 325–345. Springer (2020)

[7] Tairi, E., Moreno-Sanchez, P., Maffei, M.: A^2L: Anonymous atomic locks for scalability in payment channel hubs. In: 2021 IEEE Symposium on Security and Privacy (SP), pp. 1834–1851. IEEE (2021)

[8] Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings 35th Annual Symposium on Foundations of Computer Science, pp. 124–134. IEEE (1994)

[9] Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing, pp. 212–219. (1996)

[10] Esgin, M.F., Ersoy, O., Erkin, Z.: Post-quantum adaptor signatures and payment channel networks. In: European Symposium on Research in Computer Security, pp. 378–397. Springer (2020)

[11] Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., Stehlé, D.: Crystals-Dilithium: A lattice-based digital signature scheme. IACR Transactions on Cryptographic Hardware and Embedded Systems, 238–268. (2018)

[12] Esgin, M.F., Nguyen, N.K., Seiler, G.: Practical exact proofs from lattices: New techniques to exploit fully-splitting rings. In: Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part II 26, pp. 259–288. Springer (2020)

[13] Gilchrist, V.: An isogeny-based adaptor signature using SQISign. Master's thesis. http://hdl.handle.net/10012/18157 (2022)

[14] De Feo, L., Kohel, D., Leroux, A., Petit, C., Wesolowski, B.: SQISign: com- pact post-quantum signatures from quaternions and isogenies. Cryptology ePrint Archive, Paper 2020/1240. https://eprint.iacr.org/2020/1240 (2020)

[15] De Feo, L., Jao, D., Plût, J.: Towards quantum resistant cryptosystems from supersingular elliptic curve isogenies. Cryptology ePrint Archive, Paper 2011/506. https://eprint.iacr.org/2011/506 (2011)

[16] Castryck, W., Decru, T.: An efficient key recovery attack on SIDH. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 423–447. Springer (2023)

[17] Maino, L., Martindale, C., Panny, L., Pope, G., Wesolowski, B.: A direct key recovery attack on SIDH. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 448–471. Springer (2023)

[18] Robert, D.: Breaking SIDH in polynomial time. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 472–503. Springer (2023)

[19] Tairi, E., Moreno-Sanchez, P., Maffei, M.: Post-quantum adaptor signature for privacy-preserving off-chain payments. Cryptology ePrint Archive (2020)

[20] Beullens, W., Kleinjung, T., Vercauteren, F.: CSI-FiSh: efficient isogeny based signatures through class group computations. In: Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8–12, 2019, Proceedings, Part I, pp. 227–247. Springer (2019)

[21] Castryck, W., Lange, T., Martindale, C., Panny, L., Renes, J.: CSIDH: An Efficient Post-Quantum Commutative Group Action. Cryptology ePrint Archive, Paper 2018/383. https://eprint.iacr.org/2018/383 (2018)

[22] Bonnetain, X., Schrottenloher, A.: Quantum security analysis of CSIDH. In: Advances in Cryptology–EUROCRYPT 2020: 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10–14, 2020, Proceedings, Part II 30, pp. 493–522. Springer (2020)

[23] Peikert, C.: He gives C-sieves on the CSIDH. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, pp. 463-492. Springer (2020)

[24] De Feo, L., Fouotsa, T.B., Kutas, P., Leroux, A., Merz, S., Panny, L., Wesolowski, B.: SCALLOP: scaling the CSI-FiSh. In: IACR International Conference on Public-Key Cryptography, pp. 345–375. Springer (2023)

[25] Dartois, P., Leroux, A., Robert, D., Wesolowski, B.: SQIsignHD: New dimensions in cryptography. Cryptology ePrint Archive (2023)

[26] Erwig, A., Faust, S., Hostáková, K., Maitra, M., Riahi, S.: Two-party adaptor signatures from identification schemes. In: Public-Key Cryptography–PKC 2021: 24th IACR International Conference on Practice and Theory of Public Key Cryptography, Virtual Event, May 10–13, 2021, Proceedings, Part I, pp. 451–480. Springer (2021)

[27] Silverman, J.H.: The Arithmetic of Elliptic Curves vol. 106. Springer (2009)

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

[29] Kohel, D.R.: Endomorphism rings of elliptic curves over finite fields. Ph.D. thesis, University of California, Berkeley. (1996)

[30] Pizer, A.K.: Ramanujan graphs and Hecke operators. Bulletin (New Series) of the American Mathematical Society 23(1), pp. 127–137. (1990)

[31] Deuring, M.: Die typen der multiplikatorenringe elliptischer funktionenkörper. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg 14, pp. 197–272. (1941)

[32] Kohel, D., Lauter, K., Petit, C., Tignol, J. P.: On the quaternion l-isogeny path problem. LMS Journal of Computation and Mathematics, 17(A):418–432. (2014)

[33] Eriksen, J. K., Panny, P., Sotáková J., Veroni, M.: Deuring for the people: Supersingular elliptic curves with pre-scribed endomorphism ring in general characteristic. IACR Cryptol. ePrint Arch., 2023:106. (2023)

[34] Basso, A., Fouotsa, T.B.: New SIDH countermeasures for a more efficient key exchange. Cryptology ePrint Archive (2023)

[35] Colò, L., Kohel, D.: Orienting supersingular isogeny graphs. Cryptology ePrint Archive, Paper 2020/985. https://eprint.iacr.org/2020/985 (2020)

[36] Basso, A.: A post-quantum round-optimal oblivious PRF from isogenies. International Conference on Selected Areas in Cryptography. pp. 147–168. Springer (2023)