A Complete Security Proof of SQIsign

Marius A. Aardal[1], Andrea Basso[2], Luca De Feo[2], Sikhar Patranabis[3], and Benjamin Wesolowski[4]

[1] Aarhus University, Denmark; maardal@cs.au.dk
[2] IBM Research Europe, Zürich, Switzerland;
andrea.basso@ibm.com, crypto25@defeo.lu
[3] IBM Research India, Bangalore, India; sikhar.patranabis@ibm.com
[4] ENS de Lyon, CNRS, UMPA, UMR 5669, Lyon, France; benjamin.wesolowski@ens-lyon.fr

Abstract. SQIsign is the leading digital signature from isogenies. Despite the many improvements that have appeared in the literature, all its recents variants lack a complete security proof. In this work, we provide the first full security proof of SQIsign, as submitted to the second round of NIST's on-ramp track for digital signatures.
To do so, we introduce a new framework, which we call Fiat–Shamir with hints, that captures all those protocols where the simulator needs additional information to simulate a transcript. Using this framework, we show that SQIsign is EUF-CMA secure in the ROM, assuming the hardness of the One Endomorphism problem *with hints*, or the hardness of the Full Endomorphism Ring problem *with hints* together with a hint indistinguishability assumption; all assumptions, unlike previous ones in the literature, are non-interactive. Along the way, we prove several intermediate results that may be of independent interest.

Date of this document: February 27, 2025

1 Introduction

SQIsign is a digital signature candidate in NIST's Post-Quantum Cryptography Standardization process [CSD+23, AAA+25]. Based on the theory of isogenies between supersingular elliptic curves, its security is usually claimed to reduce to the *Supersingular Endomorphism Ring Problem* or, equivalently, to the *Supersingular Isogeny Path Problem* [Wes22]. Despite the claims, the literature only contains incomplete sketches of a security proof, glossing over key details, heuristics, and logical gaps. To complicate matters, since the scheme was first introduced in [DKL+20], several variants have appeared [DLLW23, DLRW24, NOC+24, BDD+24, DF24], each calling for different proof techniques.

In all its incarnations, SQIsign is based on a Σ-protocol which, informally, proves knowledge of the endomorphism ring of a supersingular curve E_{pk}. All variants follow the same pattern:[5]

1. The prover generates a random "commitment curve" E_{com};
2. The verifier challenges with a random isogeny $E_{\mathsf{pk}} \leftrightarrow E_{\mathsf{chl}}$;
3. The prover responds with a random isogeny $E_{\mathsf{com}} \leftrightarrow E_{\mathsf{chl}}$;
4. The verifier checks that the response is a valid isogeny connecting E_{com} to E_{chl} and rejects if that is not the case.

Such a protocol is 2-special sound: two responses $E_{\mathsf{com}} \leftrightarrow E_{\mathsf{chl}}$ and $E_{\mathsf{com}} \leftrightarrow E'_{\mathsf{chl}}$ to distinct challenges for the same commitment form a cycle $E_{\mathsf{pk}} \leftrightarrow E_{\mathsf{chl}} \leftrightarrow E_{\mathsf{com}} \leftrightarrow E'_{\mathsf{chl}} \leftrightarrow E_{\mathsf{pk}}$ of isogenies and thus an endomorphism of E_{pk}. Although this comes short of a full description of the endomorphism ring, Page and Wesolowski have given a polynomial time reduction from the Endomorphism Ring Problem to the problem of computing a single non-scalar endomorphism [PW24].

Zero knowledge is much more delicate and crucially depends on the distribution of the response. In the original version of SQIsign and in the first round NIST candidate [DKL+20, DLLW23, CSD+24], the response is a cyclic isogeny of fixed degree 2^x sampled in an algorithmically defined set. It is unknown how to efficiently sample from this distribution without knowledge of the secret key, thus the security proof simulates it with the uniform distribution on all cyclic isogenies of degree 2^x, conjecturing that the associated distinguishing problem is hard. This weak form of zero-knowledge has been the source of several issues, e.g. in [DKL+20, Sec. 6] and [BLL24].

[5] Older variants of SQIsign [DKL+20, CSD+24] swap the roles of E_{com} and E_{pk} in the challenge and response. Here we focus on the version encountered in the current NIST candidate.

In part to bypass the issue above, in part for efficiency, SQIsignHD [DLRW24] changed the definition of the response to be any isogeny $E_{\mathsf{com}} \leftrightarrow E_{\mathsf{chl}}$ of any degree up to a certain bound B. The same design was inherited by follow up works [NOC+24,BDD+24,DF24] and ultimately by the round-2 candidate [AAA+25]. Although sampling from the set of all isogenies up to a certain degree feels more natural and possibly easier to simulate, there is a catch: isogenies of large prime degree can only be efficiently represented using so-called higher-dimensional (HD) representations, which can only be efficiently produced when given a description of the endomorphism rings of the curves, i.e. of the secrets.

Thus, HD variants appear to be even worse than the original SQIsign in terms of zero-knowledge: there is no efficient algorithm to sample from a distribution even remotely similar to the response distribution, without knowledge of the secret. The way SQIsignHD and follow-ups get around this problem is by working with one or more interactive oracles that produce HD-representations of random isogenies of degree $\leq B$. This means that, either explicitly or implicitly, all HD variants have only proved security in an ad-hoc model where every party has access to these oracles producing HD-representations. These proofs do not produce any statement that applies in the more standard Random Oracle Model. Furthermore, these proofs limit themselves to showing special soundness and zero-knowledge of the underlying Σ-protocol: given the ad-hoc nature of the model, it is unclear whether the standard proof for the Fiat–Shamir transform [PS96] applies. Thus, all HD variants of SQIsign in the literature fall short of a meaningful security statement.

Contributions. In this work, we give the first full security proof of SQIsign[6] as of the second round of NIST's *on-ramp* track for digital signatures [AAA+25]. We innovate on several aspects:

- We introduce a new framework, which we call *Fiat–Shamir with hints*, that captures all those protocols where the simulator needs additional information to simulate a transcript. This framework can be applied to all HD variants of SQIsign, providing the first proof of security in the random oracle model (i.e., without any ad-hoc model) for all those protocols.
- To model the extra information needed by the simulator, previous works introduced interactive oracles to fill the gaps in the simulation. We replace the oracles with non-interactive *hints*, reducing the EUF-CMA security of SQIsign only to non-interactive assumption, whose hardness is easier to analyze. We also study different hints, identifying one type of hints that allows us to reduce the EUF-CMA security of SQIsign to a variant with hints of the endomorphism problem.
- We carefully account for losses in the reduction, thereby obtaining precise statements on the security of SQIsign's NIST parameters. Based on this, we also make suggestions that have negligible impact on the efficiency of the protocol while benefiting the strength of the security statements.
- Along our security proof, we obtain two smaller results that may be of independent interest: 1. we show that it is possible, in polynomial time, to sample from a distribution statistically close to the distribution of degrees of random isogenies of bounded degree; 2. we show there exists a tight quantum reduction (we only need a factoring oracle) from the endomorphism ring problem to the one endomorphism problem, reducing the runtime loss in the reduction from a factor approximately $2^{103}\lambda^{13}$ to a factor 24.

1.1 Technical overview

Fiat–Shamir with hints. We would like to reduce the unforgeability of SQIsign to the $\mathsf{EndRing}_p$ problem. However, the protocol does not quite fit into the usual framework for proving security of Fiat–Shamir signatures. In particular, it has two shortcomings:

1. Σ_{SQI} is special sound, but not with respect to the original relation R_{SQI}.
2. We do not know how to construct a weak Honest-Verifier Zero-Knowledge simulator for Σ_{SQI}. Without knowledge of the endomorphism ring of a curve E, we can only efficiently evaluate random isogenies from E of smooth degree. The SQIsign protocol, however, uses isogenies of arbitrary degree in its response: thus, the degree of the simulated response isogenies could not follow the correct distribution.

While the first point only requires a little extra care, the second point is a significant obstacle. Notwithstanding, to the best of current knowledge the hardness of constructing isogenies of arbitrary degree seems independent of that of $\mathsf{EndRing}_p$. If we could give the simulator a little "extra help" to

[6] The round-2 SQIsign submission slightly diverges from the protocol we analyse: for efficiency reasons, it relies on certain algorithms that may fail with very small probabilities. We discuss this in Section 7.

produce these HD representations, then it could simulate the responses without needing to break the scheme.

In Section 3, we thus introduce a new framework, which we call *Fiat–Shamir with hints*, for proving EUF-CMA security of Fiat–Shamir signatures where the simulator needs access to some additional data, which allows us to capture protocols like SQIsign and prove their security. We remark that our approach is significantly different from previous literature: all variants of SQIsign that rely on higher-dimensional representations [DLRW24, BDD+24, NOC+24, DF24] solved the simulation issue by proving security, either explicitly or implicitly, in an ad-hoc model that provides one or more oracles to compute isogenies of arbitrary degree. In contrast, we aim to prove security in the random oracle model, without any additional oracles.

EUF-CMA security of SQIsign. In Section 4, we analyze the EUF-CMA security of SQIsign in the Fiat–Shamir with hints framework, using a hint distribution that we call $\mathcal{H}^{\mathsf{sim}}$. We show that Σ_{SQI} has high commitment min-entropy, is hint-assisted wHVZK, and is special sound with respect to some soundness relation. This gives us our first result, namely that SQIsign is EUF-CMA secure in the ROM, assuming the hardness of the OneEnd problem (i.e. computing one non-trivial endomorphism) with hints.

However, we want to show that SQIsign is EUF-CMA secure as long as the EndRing problem (i.e. computing *all* endomorphisms) is hard since the EndRing problem is a much more natural and well-studied problem [EHL+18, EHL+20, Wes22, PW24]. In Section 5, we obtain a similar result by relying on a variant *with hints* of the EndRing problem. To do so, however, we cannot rely on the $\mathcal{H}^{\mathsf{sim}}$ hint distribution used so far. The reduction from the EndRing problem to the OneEnd problem requires translating endomorphisms from one curve to another, and the hint distribution $\mathcal{H}^{\mathsf{sim}}$ enforces a specific distribution that is hard to translate from one curve to another.

We sidestep this issue by introducing a new hint distribution $\mathcal{H}^{\mathsf{unif}}$ that is more suitable for the reduction. In particular, we show that the new hint distribution is *pushable*: given a 2^n-isogeny[7] $\sigma : E \to E'$ and a hint $h \leftarrow \mathcal{H}^{\mathsf{unif}}_E$ for E, we can push it through σ to get a hint for E' distributed according to $\mathcal{H}^{\mathsf{unif}}_{E'}$. We need to introduce a new assumption to switch between distributions (which we argue is hard in Remark 5.1), but the pushability property implies that the new hint distribution $\mathcal{H}^{\mathsf{unif}}$ has several advantages over $\mathcal{H}^{\mathsf{sim}}$:

- The hint formulation is simpler: the hint does not need to include what becomes the challenge isogeny in a SQIsign transcript since a hint from the public key curve can be pushed to the challenge curve.
- The OneEnd problem with $\mathcal{H}^{\mathsf{unif}}$ hints is equivalent to the EndRing problem with (the same) hints, which allows us to reduce the hardness of the EndRing problem with hints to the EUF-CMA security of SQIsign.
- A similar argument shows the EndRing problem with $\mathcal{H}^{\mathsf{unif}}$ hints is random self-reducible, and thus it admits an average-case to worst-case reduction.
- Lastly, the sampling of hints is conceptually easier, which allows us to provide an argument for why we do not expect these hints to make the EndRing problem easier.

Putting everything together, we obtain a proof of the EUF-CMA security of SQIsign in the ROM, based on the hardness of a variant of the endomorphism ring problem with hints and of the hint indistinguishability problem. When the reduction is classical, the runtime loss in the reduction is about the same as in [PW24], which is polynomial but considerably large. To obtain a tighter proof of security, we show that quantum reduction (or, equivalently, a classical reduction with access to a factoring oracle) has a runtime loss that is constant and small. This is our main result, which is summarized in Theorem 6.

2 Preliminaries

2.1 Notation

We let $\lambda \in \mathbb{N}$ denote the security parameter. All algorithms will (implicitly) take as input the unary encoding of the security parameter 1^λ. We refer to algorithms running in probabilistic polynomial time in the length of their inputs as PPT. We let $\mathsf{poly}(x_1, \ldots, x_n)$ denote an unspecified positive polynomial in $x_1, \ldots, x_n$. Similarly, $\mathsf{negl}(\lambda)$ denotes an unspecified negligible function in λ. When an algorithm $\mathcal{A}$ has black-box query access to an oracle $\mathcal{O}$, we denote this as $\mathcal{A}^{\mathcal{O}}$.

[7] We use isogenies of degree 2^n because that is the degree of the challenge isogeny in SQIsign, but the distribution is pushable through any smooth-degree isogeny.

We write ln and log for the natural and the base-2 logarithm respectively. For $x, y \in \{0,1\}^*$, we write $x \parallel y$ for their concatenation and $|x|$ for the length of x.

For a probability distribution D, we write $x \leftarrow D$ for sampling x from D. For a finite set S, we denote the uniform distribution over S by $\mathcal{U}(S)$ and write $x \xleftarrow{\$} S$ for sampling x from $\mathcal{U}(S)$. For two distributions D_0, D_1 over S, the statistical distance between D_0 and D_1 is

$$\Delta(D_0, D_1) \coloneqq \frac{1}{2} \sum_{s \in S} \left| \Pr\left[D_0 = s\right] - \Pr\left[D_1 = s\right] \right|$$

For an algorithm $\mathcal{A}$ outputting a bit b, we define its advantage in distinguishing between D_0 and D_1 as

$$\mathsf{Adv}^{\mathsf{dist}}\left[D_0, D_1\right](\mathcal{A}) \coloneqq \left| \Pr\left[1 \leftarrow \mathcal{A}(x) \mid x \leftarrow D_0\right] - \Pr\left[1 \leftarrow \mathcal{A}(x) \mid x \leftarrow D_1\right] \right|.$$

Throughout this document, p is a prime congruent to 3 mod 4 of cryptographic size ($p \approx 2^{2\lambda}$). E_A denotes the Montgomery curve $y^2 = x^3 + Ax^2 + x$. We write $\mathsf{Supersingular}_p$ for the set of supersingular Montgomery curves E_A with $A \in \mathbb{F}_{p^2}$ and $j(\mathsf{Supersingular}_p)$ for their j-invariants.

2.2 Probability distributions

The statistical distance satisfies several well known properties.

Proposition 2.1. *Let D_0 and D_1 be random variables over a finite set S.*

(i) $\Delta(D_0, D_1)$ is a metric on distributions.

(ii) For every subset $T \subseteq S$,

$$\Pr\left[D_0 \in T\right] \leq \Pr\left[D_1 \in T\right] + \Delta(D_0, D_1).$$

(iii) Let F be a random variable over functions $f : S \to T$ for some finite set T. Then

$$\Delta(F(D_0), F(D_1)) \leq \Delta(D_0, D_1).$$

There is equality if and only if every possible f is injective.

Proposition 2.2. *Let $X_1, \ldots, X_n, Y_1, \ldots Y_n$ be mutually independent random variables, where for each $i = 1, \ldots, n$, X_i and Y_i are defined over a finite set S_i. Then the statistical distance between the joint distributions $(X_1, \ldots, X_n)$ and $(Y_1, \ldots, Y_n)$ satisfies*

$$\Delta((X_1, \ldots, X_n), (Y_1, \ldots, Y_n)) \leq \sum_{i=1}^{n} \Delta(X_i, Y_i).$$

Proposition 2.3. *Let D_0 and D_1 be random variables over a finite set S. Let $\mathcal{A}$ be an algorithm taking as input an element from S and outputting a single bit. Then*

$$\mathsf{Adv}^{\mathsf{dist}}\left[D_0, D_1\right](\mathcal{A}) = \Delta(\mathcal{A}(D_0), \mathcal{A}(D_1)) \leq \Delta(D_0, D_1).$$

Proof. If we fix the random tape of $\mathcal{A}$, it becomes a deterministic function $S \to \{0,1\}$. Therefore, $\mathcal{A}$ can be seen as a random variable over functions $S \to \{0,1\}$.

$$\begin{aligned}
\Delta(\mathcal{A}(D_0), \mathcal{A}(D_1)) &= \frac{1}{2} \left| \Pr\left[\mathcal{A}(D_0) = 0\right] - \Pr\left[\mathcal{A}(D_1) = 0\right] \right| \\
&\quad + \frac{1}{2} \left| \Pr\left[\mathcal{A}(D_0) = 1\right] - \Pr\left[\mathcal{A}(D_1) = 1\right] \right| \\
&= \frac{1}{2} \left| \left(1 - \Pr\left[\mathcal{A}(D_0) = 1\right]\right) - \left(1 - \Pr\left[\mathcal{A}(D_1) = 1\right]\right) \right| \\
&\quad + \frac{1}{2} \left| \Pr\left[\mathcal{A}(D_0) = 1\right] - \Pr\left[\mathcal{A}(D_1) = 1\right] \right| \\
&= \mathsf{Adv}^{\mathsf{dist}}\left[D_0, D_1\right](\mathcal{A})
\end{aligned}$$

The inequality $\Delta(\mathcal{A}(D_0), \mathcal{A}(D_1)) \leq \Delta(D_0, D_1)$ follows by Proposition 2.1. □

Experiment 1: Weak honest-verifier zero-knowledge

$\underline{\mathsf{Real}_\Sigma(1^\lambda)}$

1: $(x, w) \leftarrow \mathsf{Gen}_R(1^\lambda)$
2: $(\mathsf{com}, \mathsf{state}) \leftarrow \mathcal{P}_1(x, w)$
3: $\mathsf{chl} \xleftarrow{\$} \mathcal{C}$
4: $\mathsf{rsp} \leftarrow \mathcal{P}_2(\mathsf{state}, \mathsf{chl})$
5: **return** $(x, \mathsf{com}, \mathsf{chl}, \mathsf{rsp})$

$\underline{\mathsf{Sim}_\Sigma(1^\lambda, \mathcal{S})}$

1: $(x, w) \leftarrow \mathsf{Gen}_R(1^\lambda)$
2: $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp}) \leftarrow \mathcal{S}(x)$
3: **return** $(x, \mathsf{com}, \mathsf{chl}, \mathsf{rsp})$

2.3 Relations and Σ-protocols

A binary relation R is a set of tuples $(x, w) \in \mathcal{X} \times \mathcal{W} \subseteq \{0,1\}^* \times \{0,1\}^*$. We refer to x as the *statement* and w as the *witness*. All relations in this paper are NP relations. Given (x, w), we can check if it is in R in polynomial time in $|x|$.

An instance generator Gen_R for R is a PPT algorithm which on input 1^λ outputs a pair (x, w) such that $(x, w) \in R$ and $|x| = \mathsf{poly}(\lambda)$. All relations we will consider come equipped with an instance generator.

Definition 2.1 (Hard relation). *Let R be a relation. We say that R is a hard relation if for all PPT algorithms $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathsf{rel}}_R(\mathcal{A}) := \Pr\left[(x, w^*) \in R \;\middle|\; (x, w) \leftarrow \mathsf{Gen}_R(1^\lambda),\ w^* \leftarrow \mathcal{A}(x)\right] = \mathsf{negl}(\lambda).$$

Definition 2.2 (Σ-protocol). *A Σ-protocol for a relation R is a 3-message interactive protocol between a prover and a verifier, given by a tuple of PPT algorithms $(\mathcal{P}_1, \mathcal{P}_2, \mathcal{V})$ as follows:*

1. *On input $(x, w) \in R$, the prover runs $(\mathsf{com}, \mathsf{state}) \leftarrow \mathcal{P}_1(x, w)$, obtaining a commitment com and the state state needed for the response computation. It sends com to the verifier.*
2. *The verifier samples a challenge chl uniformly from the challenge space $\mathcal{C}$ and sends it to the prover.*
3. *The prover replies to the verifier with the response $\mathsf{rsp} \leftarrow \mathcal{P}_2(\mathsf{state}, \mathsf{chl})$.*
4. *Finally, the verifier runs the verification algorithm on input the statement x and the transcript $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$ to obtain a bit $b \leftarrow \mathcal{V}(x, \mathsf{com}, \mathsf{chl}, \mathsf{rsp})$. It accepts if and only if $b = 1$.*

The Σ-protocol should always have *correctness*. If the prover and verifier run the protocol honestly, the verifier should accept except with negligible probability in λ. In addition, there are several other security properties of interest.

Definition 2.3 (wHVZK). *Σ is weak honest-verifier zero-knowledge (wHVZK) if there exists a PPT algorithm $\mathcal{S}$, called the simulator, such that for all PPT algorithms $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathsf{wHVZK}}_{\Sigma,\mathcal{S}}(\mathcal{A}) := \mathsf{Adv}^{\mathsf{dist}}\left[\mathsf{Real}_\Sigma(1^\lambda), \mathsf{Sim}_\Sigma(1^\lambda, \mathcal{S})\right](\mathcal{A}) = \mathsf{negl}(\lambda),$$

where $\mathsf{Real}_\Sigma(1^\lambda)$ and $\mathsf{Sim}_\Sigma(1^\lambda, \mathcal{S})$ are the distributions defined in Experiment 1.

Definition 2.4 (MinEnt). *Σ has high commitment min-entropy if it holds that*

$$\mathsf{MinEnt}(\Sigma) := \max_{(x,w)} \max_{\mathsf{com}} \Pr\left[\mathsf{com} = \mathsf{com}' \;\middle|\; (\mathsf{com}', \mathsf{state}) \leftarrow \mathcal{P}_1(x, w)\right] = \mathsf{negl}(\lambda),$$

where the first max ranges over the pairs that might be output by $\mathsf{Gen}_R(1^\lambda)$.

Note that this definition is slightly different from previous works (e.g. [AABN02]). The actual min-entropy of the commitment is $-\log(\mathsf{MinEnt}(\Sigma))$.

Definition 2.5 (Special soundness). *Σ is special sound if, given a statement x and two accepting transcripts $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$ and $(\mathsf{com}, \mathsf{chl}', \mathsf{rsp}')$ for x with $\mathsf{chl} \neq \mathsf{chl}'$, we can compute a witness s.t. $(x, w) \in R$ in polynomial time in $|x|$.*

Game 1: $\mathsf{IMP\text{-}PA}_\Sigma(\mathcal{A}_1, \mathcal{A}_2)$

$(x, w) \leftarrow \mathsf{Gen}_R(1^\lambda)$
$(\mathsf{com}^*, \mathsf{state}^*) \leftarrow \mathcal{A}_1^{\mathsf{OTrans}_{x,w}}(x)$
$\mathsf{chl}^* \xleftarrow{\$} \mathcal{C}$
$\mathsf{rsp}^* \leftarrow \mathcal{A}_2(\mathsf{state}^*, \mathsf{chl}^*)$
return $\mathsf{Ver}_\Sigma(x, \mathsf{com}^*, \mathsf{chl}^*, \mathsf{rsp}^*)$

$\mathsf{OTrans}_{x,w}()$:
$\mathsf{com}, \mathsf{state} \leftarrow \mathcal{P}_1(x, w)$
$\mathsf{chl} \xleftarrow{\$} \mathcal{C}$
$\mathsf{rsp} \leftarrow \mathcal{P}_2(x, w, \mathsf{state})$
return $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$

Definition 2.6 (Knowledge soundness). *Σ is knowledge sound with knowledge error $\kappa : \mathbb{N} \to [0, 1]$ if there exists a knowledge extractor $\mathcal{E}$, defined as follows: On input a statement x and given rewindable black-box query-access to a prover $\mathcal{P}^*$, it runs in an expected polynomial number of steps in $|x|$ (counting each invocation of $\mathcal{P}^*$ as a single step), and finds a witness for x with probability*

$$\Pr\left[(x, w^*) \in R \;\middle|\; w^* \leftarrow \mathcal{E}^{\mathcal{P}^*}(x)\right] \geq \varepsilon(\mathcal{P}^*) - \kappa(|x|),$$

where $\varepsilon(\mathcal{P}^*)$ is the probability that the verifier accepts in the protocol with $\mathcal{P}^*$.

We say that Σ is *knowledge sound* when $\kappa(|x|) = \mathsf{negl}(|x|)$.

Lemma 2.1 ([ACK21]). *If Σ is special sound, then it is knowledge sound with knowledge error $\kappa = 1/|\mathcal{C}|$. In expectation, the extractor invokes the prover at most 2 times.*

A Σ-protocol can be seen as an identification scheme [AABN02]. The prover is trying to convince the verifier that it knows a secret w associated to the identity x. We consider security against an attacker that might have eavesdropped on honest runs of the protocol before it attempts to impersonate the identity x.

Definition 2.7 (IMP-PA [AABN02]). *We say that Σ is secure under passive impersonation attacks (IMP-PA) if for all PPT two-stage algorithms $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ it holds that*

$$\mathsf{Adv}_\Sigma^{\mathsf{IMP\text{-}PA}}(\mathcal{A}) \coloneqq \Pr\left[\mathsf{IMP\text{-}PA}_\Sigma(\mathcal{A}) = 1\right] = \mathsf{negl}(\lambda),$$

where $\mathsf{IMP\text{-}PA}_\Sigma$ is the game presented in Game 1.

If Σ is a Σ-protocol for a hard relation R that is special sound, wHVZK and has $1/|\mathcal{C}| = \mathsf{negl}(\lambda)$, then it can be shown that Σ is IMP-PA-secure. In fact, we prove a generalization of this result in Section 3.1.

2.4 Signature schemes in the ROM

In the random oracle model (ROM) all algorithms are given black-box query access to the random oracle RO, implementing a uniformly random function $\{0, 1\}^* \to \mathcal{C}$. We usually omit the dependency on the random oracle from our notation, writing $\mathcal{A}$ instead of $\mathcal{A}^{\mathsf{RO}}$.

Definition 2.8 (Signature scheme). *Let M be some set. A signature scheme in the ROM for a message space M is a tuple of PPT algorithms $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ defined as follows.*

- $\mathsf{Gen}(1^\lambda)$*: On input the security parameter λ, the key generation algorithm outputs the public key* pk *and the secret key* sk*.*
- $\mathsf{Sign}(\mathsf{sk}, \mathsf{msg})$*: On input the secret key* sk *and a message* $\mathsf{msg} \in M$*, the signing algorithm outputs a signature* sig*.*
- $\mathsf{Ver}(\mathsf{pk}, \mathsf{msg}, \mathsf{sig})$*: On input the public key* pk*, a message* msg *and a signature* sig*, the verification algorithm outputs 1 (accept) or 0 (reject).*

A signature scheme should satisfy two security properties, correctness and EUF-CMA. The former says that the verification of a correctly generated signature should only fail with negligible probability in λ. The latter is defined as follows.

Definition 2.9 (EUF-CMA). *Let $\mathsf{S} = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ be a signature scheme. We say that S satisfies existential unforgeability under chosen message attacks (EUF-CMA) if it holds for all PPT random-oracle algorithms $\mathcal{A}$ that*

$$\mathsf{Adv}_\mathsf{S}^{\mathsf{EUF\text{-}CMA}}(\mathcal{A}) \coloneqq \Pr\left[\mathsf{EUF\text{-}CMA}_\mathsf{S}(\mathcal{A}) = 1\right] = \mathsf{negl}(\lambda),$$

where $\mathsf{EUF\text{-}CMA}_\mathsf{S}$ is the game presented in Game 2.

Game 2: $\mathsf{EUF\text{-}CMA}_{\mathsf{S}}(\mathcal{A})$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$
$\mathcal{Q} := \varnothing$
$(\mathsf{msg}^*, \mathsf{sig}^*) \leftarrow \mathcal{A}^{\mathsf{OSign}}(\mathsf{pk})$
if $\mathsf{Ver}(\mathsf{pk}, \mathsf{msg}^*, \mathsf{sig}^*) = 1 \ \wedge \ \mathsf{msg}^* \notin \mathcal{Q}$ **then**
    return 1
else
    return 0

$\underline{\mathsf{OSign}(\mathsf{msg})}$
$\mathsf{sig} \leftarrow \mathsf{Sign}(\mathsf{sk}, \mathsf{msg})$
$\mathcal{Q} := \mathcal{Q} \cup \{\mathsf{msg}\}$
return sig

Σ-protocols can be used to construct signature schemes in the ROM, using the *Fiat-Shamir transform*. The Fiat-Shamir transform $\mathsf{FS}[\Sigma]$ of Σ is the non-interactive protocol where the challenge is obtained as $\mathsf{chl} \leftarrow \mathsf{RO}(\mathsf{com} \parallel x)$.

Definition 2.10 (Fiat-Shamir signature). *Let Σ be a Σ-protocol for a relation R and let $M \subseteq \{0,1\}^*$. We define the signature scheme $\mathsf{SIG}[\Sigma]$ as follows.*

- $\mathsf{Gen}(1^\lambda)$*: Sample* $(x, w) \leftarrow \mathsf{Gen}_R(1^\lambda)$*. Set* $\mathsf{pk} := x$ *and* $\mathsf{sk} := (x, w)$*, and output* $(\mathsf{pk}, \mathsf{sk})$*.*
- $\mathsf{Sign}(\mathsf{sk}, \mathsf{msg})$*: Generate a transcript* $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$ *using the prover algorithm in* $\mathsf{FS}[\Sigma]$*, but compute the challenge as* $\mathsf{chl} \leftarrow \mathsf{RO}(\mathsf{com} \parallel x \parallel \mathsf{msg})$*. Output* $\mathsf{sig} := (\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$*.*
- $\mathsf{Ver}(\mathsf{pk}, \mathsf{msg}, \mathsf{sig})$*: Accept if and only if* $\mathsf{sig} = (\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$ *such that* $\mathcal{V}(x, \mathsf{com}, \mathsf{chl}, \mathsf{rsp}) = 1$ *and* $\mathsf{chl} = \mathsf{RO}(\mathsf{com} \parallel x \parallel \mathsf{msg})$*.*

Observe that $\mathsf{SIG}[\Sigma]$ is essentially the Fiat-Shamir transform of Σ, but with the relation modified to $R^* = \{(x \parallel \mathsf{msg}, w) \mid (x, w) \in R, \mathsf{msg} \in M\}$.

If Σ is a IMP-PA-secure identification scheme and has high commitment min-entropy, then $\mathsf{SIG}[\Sigma]$ is EUF-CMA-secure.

Lemma 2.2 ([AABN02, Lemma 3.5]). *For any PPT algorithm $\mathcal{A}$ attacking the* EUF-CMA *of* $\mathsf{SIG}[\Sigma]$*, there exists a PPT two-stage algorithm $\mathcal{B}$ such that*

$$\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma]}(\mathcal{A}) \leq (q+1) \cdot \mathsf{Adv}^{\mathsf{IMP\text{-}PA}}_{\Sigma}(\mathcal{B}) + (q+s+1)s \cdot \mathsf{MinEnt}(\Sigma),$$

where q and s are upper bounds on the number of queries that $\mathcal{A}$ makes to RO *and* OSign*, respectively. Furthermore, $\mathcal{B}$ makes at most s queries to the transcript oracle* OTrans*.*

2.5 Computing isogenies

SQIsign uses three distinct ways to encode and compute with isogenies. It is useful to define a computational abstraction.

Definition 2.11 ([BDD+24]). *Let $\mathbb{F}_q$ be a finite field. An isogeny evaluator $\mathscr{E}$ is a pair of polynomial time algorithms:*

- $\mathscr{E}.\mathsf{valid}\,(D)$*: On input a string $D \in \{0,1\}^*$ it outputs $\perp$ or a triple (E, E', d). In the latter case, E and E' are elliptic curves defined over $\mathbb{F}_q$ and D represents an isogeny $\varphi : E \to E'$ of degree d.*
- $\mathscr{E}.\mathsf{eval}\,(D, P)$*: On input a string $D \in \{0,1\}^*$ and a point $P \in E(\mathbb{F}_{q^k})$, it outputs the image point $\varphi(P) \in E'(\mathbb{F}_{q^k})$ if $\mathscr{E}.\mathsf{valid}\,(D) = (E, E', d)$, otherwise the output is undefined.*

In the case that $\mathscr{E}.\mathsf{valid}\,(D) \neq \perp$ and D is of size polynomial in $\log(q)$ and $\log(d)$, we say that D is an efficient representation of φ (with respect to $\mathscr{E}$).

The first representation used in SQIsign is restricted to isogenies of degree 2^n, represented as 2-*isogeny walks*: the isogeny is stored as a chain $\varphi = \varphi_1 \circ \varphi_2 \circ \cdots \circ \varphi_n$ of isogenies of degree 2. It is clear this representation is efficient.

The second is the *ideal representation*. If $\mathcal{O} \simeq \mathrm{End}(E)$ is a maximal order of the quaternion algebra ramified at p and ∞, the *Deuring correspondence* establishes a bijection between left ideals of $\mathcal{O}$ and isogenies with domain E. Algorithmically, given a basis of a left ideal of $\mathcal{O}$, there are PPT algorithms to compute the degree and the image of the associated isogeny and to evaluate it on arbitrary points [EHL+18, Wes22, Ler22]. This efficient representation intrinsically makes use of the secret key in SQIsign, and is thus only used internally by the prover.

The final representation is the so-called *higher-dimensional (HD) representation*. Following Robert [Rob22], any isogeny between elliptic curves can be "embedded" into an isogeny of higher dimensional abelian varieties using Kani's lemma [Kan97]. SQIsign only uses embeddings into chains of $(2,2)$-isogenies of (2-dimensional) abelian surfaces, which can be efficiently stored and evaluated. Although these yield efficient representations of any isogeny, they are in general difficult to produce in the first place. In SQIsign they are easily computed given knowledge of the secret key, but the difficulty of computing them in general is the key obstacle to proving security which we address in Section 3.

Both the ideal and the HD representation support an extended list of additional algorithms operating on them.

Lemma 2.3 (Algorithms on representations). *For each of the operations below there is a PPT algorithm which operates on ideal (resp. HD) representations:*

- ***Dual:*** *On input $\varphi : E_1 \to E_2$, compute $\widehat{\varphi} : E_2 \to E_1$.*
- ***Equality check:*** *On input $\varphi, \psi : E_1 \to E_2$, check whether $\varphi = \psi$.*
- ***Composition:*** *On input $\varphi : E_1 \to E_2$ and $\psi : E_2 \to E_3$, compute $\psi \circ \varphi : E_1 \to E_3$.*
- ***Splitting:*** *On input $\varphi : E_1 \to E_2$ and coprime n_1, n_2 s.t. $\deg(\varphi) = n_1 n_2$, find φ_1, φ_1' of degree n_1 and φ_2, φ_2' of degree n_2 s.t. $\varphi = \varphi_2' \circ \varphi_1 = \varphi_1' \circ \varphi_2$.*
- ***Division:*** *On input $\varphi : E_1 \to E_2$, $\varphi_l : E_1 \to E_2'$ and $\varphi_r : E_1' \to E_2$, check if there exists φ_l', φ_r' s.t. $\varphi = \varphi_r' \circ \varphi_l$ or $\varphi = \varphi_r \circ \varphi_l'$, and if so output them.*
- ***Pushforward:*** *On input $\varphi : E_1 \to E_2$ and $\psi : E_1 \to E_3$ with coprime degrees, compute a pushforward $\varphi' : E_3 \to E_{23}$ of φ by ψ, meaning that $\ker(\varphi') = \psi(\ker(\varphi))$.*
- ***Backtracking:*** *On input $\varphi : E_2 \to E_3$ and $\sigma : E_4 \to E_2$ with $\gcd(\deg\varphi, \deg\sigma)$ smooth, find ψ, σ', and φ' s.t. $\sigma = \widehat{\psi} \circ \sigma'$, $\varphi = \varphi' \circ \psi$, and $\ker(\psi) = \ker(\varphi) \cap \ker(\widehat{\sigma})$. We say that ψ is the* backtracking component *of φ and σ. If φ' and σ' are cyclic, then so is $\varphi' \circ \sigma'$.*

Proof. All of the above operations are discussed in the literature [Ler22, Rob24], with the exception of **Backtracking**, which we now detail.

We start with the ideal representation. Let I_φ and $I_{\hat{\sigma}}$ be the left $\mathrm{End}(E_2)$-ideals vanishing on $\ker(\varphi)$ and $\ker(\hat{\sigma})$ respectively. Then $I_\varphi + I_{\hat{\sigma}}$ is the vanishing ideal of $\ker\psi$ and thus an ideal representation of ψ. Now we can use **Division** to compute ideal representations of φ' and σ'.

For the HD representation we need more work. First, using **Splitting** and the fact that $\gcd(\deg\varphi, \deg\sigma)$ is smooth we reduce to the case where $\deg\varphi$ and $\deg\sigma$ are powers of a small prime ℓ.

Next, we factor the common factors $[\ell]$ from φ and σ, by constructing an HD representation of $[\ell]$ ([Rob24, Proposition 6.6]) and applying **Division** repeatedly. We thus reduce to the case where at least one of φ and σ is cyclic; without loss of generality, assume φ is.

Finally, we enumerate all ℓ-isogenies ψ_i from E_2 using Vélu's formulas, and use **Division** to factor $\varphi = \varphi_0 \circ \psi_i$, then again **Division** to factor $\hat{\sigma} = \hat{\sigma}_0 \circ \psi_i$, and continue recursively on φ_0 and σ_0. We stop when one of the **Division** attempts fails, at which point, if both φ and σ are cyclic, $\varphi \circ \sigma$ must also be. □

Computational assumptions. We recall two foundational computational problems in isogeny-based cryptography.

Problem 1 ($\mathsf{EndRing}_p$). *Given a curve $E \in \mathsf{Supersingular}_p$, find four endomorphisms in efficient representation that form a basis of $\mathrm{End}(E)$ as a lattice.*

Problem 2 (OneEnd_p). *Given a curve $E \in \mathsf{Supersingular}_p$, find an endomorphism in $\mathrm{End}(E) \setminus \mathbb{Z}$ in efficient representation.*

These problems are believed to computationally hard, even for quantum adversaries. Specifically, we assume that they have worst-case hardness, meaning that no algorithm can solve them in polynomial time in $\log p$ for all curves. This implies that the problems must be hard for uniformly random curves, by well-known worst-case to average-case reductions. Furthermore, a recent line of work [Wes22, PW24] shows that they are equivalent under polynomial-time reductions.

The stationary distribution S_j is the limit distribution of the random walking process which from a $j(E) \in j(\mathsf{Supersingular}_p)$ selects uniformly at random an ℓ-isogeny $\varphi : E \to E'$ among those that do not backtrack and moves to $j(E')$. [BCC+23, Theorem 11] proves the following fact.

Table 1: The public parameters of SQIsign

Parameter	p	e_{rsp}	e_{chl}	N_{mix}
Description	$p = c \cdot 2^e - 1$ prime, $p < 2^{2\lambda}$, $e \approx 2\lambda$	$\lceil \log_2(\sqrt{p}) \rceil$	$e - e_{\mathsf{rsp}}$	Smallest prime $> 2^{4\lambda}$

Proposition 2.4. *The* stationary distribution S_j *on* $j(\mathsf{Supersingular}_p)$ *is the distribution with*

$$\Pr\left[j(E) \leftarrow S_j\right] = \frac{24}{(p-1)|\mathrm{Aut}(E)|}.$$

Let $\ell \neq p$ *be prime. Let* π *be a distribution on* $j(\mathsf{Supersingular}_p)$ *and let* $D_\pi^{\ell^k}$ *be the distribution after* k *steps of a non-backtracking random walk, then*

$$\Delta(D_\pi^{\ell^k}, S_j) \leq \frac{(k+1)}{4}\sqrt{\frac{p}{\ell^k}}.$$

We define the stationary distribution S on $\mathsf{Supersingular}_p$ such that $j^* \leftarrow S_j$ and A is uniformly distributed among those such that $j(E_A) = j^*$.

2.6 The SQIsign signature scheme

We recap SQIsign, as submitted to the round-two NIST standardization process for additional post-quantum signatures [AAA+25]. It is a Fiat–Shamir signature constructed from a Σ-protocol Σ_{SQI}. We reproduce high-level pseudocode for the protocol; for a more detailed description see [AAA+25].

Public parameters. For each security level λ, SQIsign defines some public parameters pp, presented in Table 1. We denote the algorithm that deterministically outputs pp by $\mathsf{PublicParam}_{\mathsf{SQI}}(1^\lambda)$. In addition, E_0 denotes the starting curve with j-invariant 1728 and known endomorphism ring $\mathrm{End}(E_0) \cong \mathcal{O}_0$.

Algorithmic building blocks. We highlight a few recurring algorithmic building blocks in Table 2.

Table 2: Algorithmic building blocks

Algorithm	**Inputs**	**Outputs**
DeterministicBasis	$E \in \mathsf{Supersingular}_p$	A deterministically computed basis (P, Q) for $E[2^e]$.
IdealToIsogeny	Left $\mathcal{O}_0$-ideal I	E_I and (P_I, Q_I), where $\varphi_I : E_0 \to E_I$ corresponds to I, and $P_I, Q_I = \varphi_I(P_0), \varphi_I(Q_0)$, $(P_0, Q_0) = \mathsf{DeterministicBasis}(E_0)$.
IsogenyEval2-2	$E, E_{12} \in \mathsf{Supersingular}_p$, K_1 and K_2 in $E \times E_{12}$ isotropic[8] of order 2^{f+2}, a list eval-pts of points in $E \times E_{12}$.	E_1, E_2 and $[\Phi(P) \mid P \in \text{eval-pts}]$, where $\Phi : E \times E_{12} \to E_1 \times E_2$ is a deterministically computed $(2^f, 2^f)$-isogeny with $\ker(\Phi) = \langle [4]K_1, [4]K_2 \rangle$.
MontgomeryRandomize	$A \in \mathbb{F}_{p^2}$	$A' \in \mathbb{F}_{p^2}$ uniformly random s.t. $j(E_A) = j(E_{A'})$.
RandomIdealGivenNorm	$N \in \mathbb{N}$ s.t. $p \nmid N$.	Uniformly random primitive left $\mathcal{O}_0$-ideal I of norm N.

[8] Two points of order n are said isotropic if their Weil pairing of order n is trivial.

Relation. The relation of Σ_{SQI} is

$$R_{\mathsf{SQI}} = \left\{ ((\mathsf{pp}, A_{\mathsf{pk}}), I_{\mathsf{sk}}) \;\middle|\; \begin{array}{c} \mathsf{pp} \text{ public parameters,} \\ A_{\mathsf{pk}} \in \mathbb{F}_{p^2},\ E_{\mathsf{pk}} \coloneqq E_{A_{\mathsf{pk}}} \in \mathsf{Supersingular}_p, \\ I_{\mathsf{sk}} \text{ is a left } \mathcal{O}_0\text{-ideal and} \\ \mathcal{O}_R(I_{\mathsf{sk}}) \cong \mathrm{End}(E_{\mathsf{pk}}) \end{array} \right\}.$$

We refer to the Montgomery A-invariant A_{pk} as the public key, and I_{sk} as the secret key. We can view R_{SQI} as a relation for the $\mathsf{EndRing}_p$ problem, since a basis for $\mathrm{End}(E_{\mathsf{pk}})$ can be efficiently recovered from I_{sk}.

The instance generator $\mathsf{Gen}_{R_{\mathsf{SQI}}}$ is defined in Algorithm 1. The prime N_{mix} is chosen so that $\Delta(j(E_{\mathsf{pk}}), S) \leq 2^{-\lambda}$. In addition, A_{pk} is randomized, to prevent any leakage from the choice of Montgomery coefficient.

The Σ-protocol Σ_{SQI}. The commitment, challenge and response algorithms are presented in Algorithm 2. The verification Algorithm 3 checks that the response encodes an efficient representation of an isogeny $E_{\mathsf{pk}} \to E_{\mathsf{com}}$. For an overview of the isogenies involved in the protocol, see Fig. 1.

Remark 2.1. We have modified the description of Σ_{SQI} slightly by adding the lines 24-25 in the response algorithm of Algorithm 2. These steps normalize the points $P_{\mathsf{chl}}, Q_{\mathsf{chl}}$ by making a deterministic choice of sign. This makes the distribution easier to simulate. However, in the optimized implementation of SQIsign, the points are represented by their x-coordinate. Since P_{chl} and $-P_{\mathsf{chl}}$ have the same x-coordinate, this means that the point normalization is not needed there.

Signature scheme. The SQIsign signature scheme is a standard Fiat–Shamir transform of Σ_{SQI}. We denote it $\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]$:

- Key generation is identical to $\mathsf{Gen}_{R_{\mathsf{SQI}}}$.
- The signature is generated by running Commitment to generate E_{com}, setting the challenge as $\mathsf{chl} \leftarrow \mathsf{RO}(j(E_{\mathsf{com}}) \parallel A_{\mathsf{pk}} \parallel \mathsf{msg})$ and finally running Response to get rsp. The signature is $(\mathsf{chl}, \mathsf{rsp})$.
- The verifier recovers $E_{\mathsf{com}} \simeq F_1$ from the signature[9], checks that it defines a valid isogeny $E_{\mathsf{pk}} \to E_{\mathsf{com}}$ and that $\mathsf{chl} = \mathsf{RO}(j(E_{\mathsf{com}}) \parallel A_{\mathsf{pk}} \parallel \mathsf{msg})$.

Algorithm 1: The instance generator $\mathsf{Gen}_{R_{\mathsf{SQI}}}(1^\lambda)$

1: $\mathsf{pp} \leftarrow \mathsf{PublicParams}_{\mathsf{SQI}}(1^\lambda)$.
2: $I_{\mathsf{sk}} \leftarrow \mathsf{RandomIdealGivenNorm}(N_{\mathsf{mix}})$.
3: $E_A, \_ \coloneqq \mathsf{IdealToIsogeny}(I_{\mathsf{sk}})$.
4: $A_{\mathsf{pk}} \coloneqq \mathsf{MontgomeryRandomize}(A)$.
5: $x \coloneqq (\mathsf{pp}, A_{\mathsf{pk}})$ and $w \coloneqq I_{\mathsf{sk}}$.
6: **return** (x, w).

[9] SQIsign is *commitment recoverable with perfect uniqueness*, so dropping the commitment from the signature does not affect security [BBSS18].

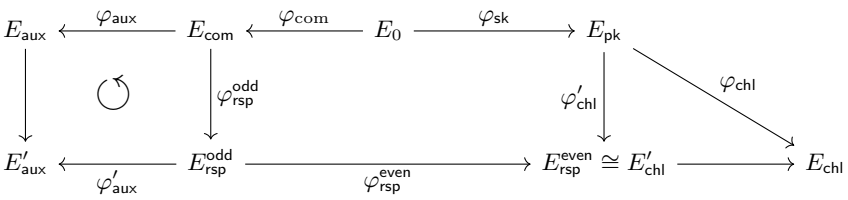

Fig. 1: Diagram of Σ_{SQI}.

Algorithm 2: The Σ-protocol Σ_{SQI}

Commitment($\mathsf{pp}, A_{\mathsf{pk}}, I_{\mathsf{sk}}$):

$I_{\mathsf{com}} \leftarrow \mathsf{RandomIdealGivenNorm}(N_{\mathsf{mix}})$.
$E_{\mathsf{com}}, (P_{\mathsf{com}}, Q_{\mathsf{com}}) \leftarrow \mathsf{IdealToIsogeny}(I_{\mathsf{com}})$.
$\mathsf{com} \coloneqq j(E_{\mathsf{com}})$; $\mathsf{state} \coloneqq (\mathsf{pp}, A_{\mathsf{pk}}, I_{\mathsf{sk}}, I_{\mathsf{com}}, E_{\mathsf{com}}, P_{\mathsf{com}}, Q_{\mathsf{com}})$.
return com, state.

Challenge:

$\mathsf{chl} \xleftarrow{\$} \{0, \dots, 2^{e_{\mathsf{chl}}} - 1\}$. // Defines the challenge isogeny $\varphi_{\mathsf{chl}} : E_{\mathsf{pk}} \to E_{\mathsf{chl}}$ with $\ker(\varphi_{\mathsf{chl}}) = \langle P_{\mathsf{pk}} + [\mathsf{chl}]Q_{\mathsf{pk}} \rangle$, where $(P_{\mathsf{pk}}, Q_{\mathsf{pk}}) \coloneqq \mathsf{DeterministicBasis}(E_{\mathsf{pk}})$.
return chl.

Response(state, chl):

Compute the left $\mathcal{O}_{\mathsf{pk}}$-ideal I_{chl} corresponding to φ_{chl}, where $\mathcal{O}_{\mathsf{pk}} = \mathcal{O}_R(I_{\mathsf{sk}})$.
Sample a uniformly random ideal J equivalent to $\overline{I_{\mathsf{com}}} \cdot I_{\mathsf{sk}} \cdot I_{\mathsf{chl}}$ of norm $< 2^{e_{\mathsf{rsp}}}$.
Decompose $J = m \cdot I_{\mathsf{rsp}}$ with I_{rsp} a primitive ideal.
// I_{rsp} corresponds to the cyclic response isogeny $\varphi_{\mathsf{rsp}} : E_{\mathsf{com}} \to E_{\mathsf{chl}}$.
Write $\mathrm{nrd}(I_{\mathsf{rsp}}) = 2^n d'$ with d' an odd integer.
Let n_{bt} be the largest integer s.t. $I_{\mathsf{chl}} \cdot \overline{I_{\mathsf{rsp}}} \subseteq 2^{n_{\mathsf{bt}}}\mathcal{O}_{\mathsf{pk}}$, and $r' \coloneqq n - n_{\mathsf{bt}}$.
// n_{bt} is the length of the part of $\widehat{\varphi_{\mathsf{rsp}}}$ that backtracks with φ_{chl}.
Factor I_{rsp} as $I_{\mathsf{rsp}}^{(1)} \cdot I_{\mathsf{rsp}}^{(0)} \cdot I'$ s.t. $\mathrm{nrd}(I_{\mathsf{rsp}}^{(1)}) = d'$, $\mathrm{nrd}(I_{\mathsf{rsp}}^{(0)}) = 2^{r'}$, $I_{\mathsf{chl}} \cdot \overline{I'} \subseteq 2^{n_{\mathsf{bt}}}\mathcal{O}_{\mathsf{pk}}$.
// $\varphi_{\mathsf{rsp}} : E_{\mathsf{com}} \xrightarrow{\varphi_{\mathsf{rsp}}^{\mathsf{odd}}} E_{\mathsf{rsp}}^{\mathsf{odd}} \xrightarrow{\varphi_{\mathsf{rsp}}^{\mathsf{even}}} E_{\mathsf{rsp}}^{\mathsf{even}} \xrightarrow{\psi} E_{\mathsf{chl}}$, with $\widehat{\psi}$ backtracking φ_{chl}.
$I''_{\mathsf{aux}} \leftarrow \mathsf{RandomIdealGivenNorm}(2^{e_{\mathsf{rsp}}-n} - d')$.
Let I_{aux} be the pushforward of I''_{aux} by I_{com}. // $\varphi_{\mathsf{aux}} : E_{\mathsf{com}} \to E_{\mathsf{aux}}$.
Let I'_{aux} be the pushforward of I_{aux} by $I_{\mathsf{rsp}}^{(1)}$. // $\varphi'_{\mathsf{aux}} : E_{\mathsf{rsp}}^{\mathsf{odd}} \to E'_{\mathsf{aux}}$.
$E'_{\mathsf{aux}}, (K_{\mathsf{aux},1}, K_{\mathsf{aux},2}) \leftarrow \mathsf{IdealToIsogeny}(I_{\mathsf{com}} \cdot I_{\mathsf{rsp}}^{(0)} \cdot I'_{\mathsf{aux}})$.
$K_1 \coloneqq ([2^{e-(e_{\mathsf{rsp}}-n+2)}d']P_{\mathsf{com}},\ [2^{e-(e_{\mathsf{rsp}}-n+2)}]K_{\mathsf{aux},1})$.
$K_2 \coloneqq ([2^{e-(e_{\mathsf{rsp}}-n+2)}d']Q_{\mathsf{com}},\ [2^{e-(e_{\mathsf{rsp}}-n+2)}]K_{\mathsf{aux},2})$.
$\mathsf{eval\text{-}pts} \coloneqq [(P_{\mathsf{com}}, 0), (Q_{\mathsf{com}}, 0)]$.
$E_{\mathsf{rsp}}^{\mathsf{odd}}, E_{\mathsf{aux}}, \mathsf{eval\text{-}pts}' \leftarrow \mathsf{IsogenyEval2\text{-}2}(E_{\mathsf{com}}, E'_{\mathsf{aux}}, K_1, K_2, \mathsf{eval\text{-}pts})$.
Parse $\mathsf{eval\text{-}pts}'$ as $[(R_{\mathsf{chl}}, R_{\mathsf{aux}}), (S_{\mathsf{chl}}, S_{\mathsf{aux}})]$.
$R_{\mathsf{chl}}, S_{\mathsf{chl}} \coloneqq \varphi_{\mathsf{rsp}}^{\mathsf{even}}(R_{\mathsf{chl}}), \varphi_{\mathsf{rsp}}^{\mathsf{even}}(S_{\mathsf{chl}})$.
Compute the non-backtracking part of the challenge isogeny $\varphi'_{\mathsf{chl}} : E_{\mathsf{pk}} \to E'_{\mathsf{chl}}$ with $\ker(\varphi'_{\mathsf{chl}}) = [2^{n_{\mathsf{bt}}}]\ker(\varphi_{\mathsf{chl}})$.
$R_{\mathsf{chl}}, S_{\mathsf{chl}} \coloneqq \iota_{\mathsf{aux}}(R_{\mathsf{chl}}), \iota_{\mathsf{aux}}(R_{\mathsf{chl}})$, for an isomorphism $\iota_{\mathsf{chl}} : E_{\mathsf{rsp}}^{\mathsf{even}} \to E'_{\mathsf{chl}}$.
$A^\dagger_{\mathsf{aux}} \leftarrow \mathsf{MontgomeryRandomize}(A_{\mathsf{aux}})$ and $E^\dagger_{\mathsf{aux}} \coloneqq E_{A^\dagger_{\mathsf{aux}}}$. // $E_{\mathsf{aux}} = E_{A_{\mathsf{aux}}}$.
$R_{\mathsf{aux}}, S_{\mathsf{aux}} \coloneqq \iota_{\mathsf{aux}}(R_{\mathsf{aux}}), \iota_{\mathsf{aux}}(S_{\mathsf{aux}})$, for an isomorphism $\iota_{\mathsf{aux}} : E_{\mathsf{aux}} \to E^\dagger_{\mathsf{aux}}$.
$P_{\mathsf{aux}}, Q_{\mathsf{aux}} \coloneqq [2^{e-(e_{\mathsf{rsp}}-n_{\mathsf{bt}}+2)}] \cdot \mathsf{DeterministicBasis}(E^\dagger_{\mathsf{aux}})$.
Compute $a, b, c, d \in \mathbb{Z}$ s.t. $P_{\mathsf{aux}} = aR_{\mathsf{aux}} + bS_{\mathsf{aux}}$ and $Q_{\mathsf{aux}} = cR_{\mathsf{aux}} + dS_{\mathsf{aux}}$.
$P_{\mathsf{chl}}, Q_{\mathsf{chl}} \coloneqq (aR_{\mathsf{chl}} + bS_{\mathsf{chl}},\ cR_{\mathsf{chl}} + dS_{\mathsf{chl}})$.
if $-P_{\mathsf{chl}}$ is lexicographically smaller than P_{chl} **then**
    $P_{\mathsf{chl}}, Q_{\mathsf{chl}} \coloneqq -P_{\mathsf{chl}}, -Q_{\mathsf{chl}}$
return $\mathsf{rsp} \coloneqq (n_{\mathsf{bt}}, r', A^\dagger_{\mathsf{aux}}, P_{\mathsf{chl}}, Q_{\mathsf{chl}})$.

Algorithm 3: Verification

Input: The statement $(\mathsf{pp}, A_{\mathsf{pk}})$. // A_{pk} determines the public-key curve E_{pk}
Input: The transcript $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$ with $\mathsf{rsp} = (n_{\mathsf{bt}}, r', A^{\dagger}_{\mathsf{aux}}, P_{\mathsf{chl}}, Q_{\mathsf{chl}})$.
$P_{\mathsf{pk}}, Q_{\mathsf{pk}} \coloneqq \mathsf{DeterministicBasis}(E_{\mathsf{pk}})$.
Compute $E_{\mathsf{pk}}, \varphi'_{\mathsf{chl}} : E_0 \to E'_{\mathsf{chl}}$ with $\ker(\varphi_{\mathsf{chl}}) = \langle [2^{n_{\mathsf{bt}}}](P_{\mathsf{pk}} + [\mathsf{chl}]Q_{\mathsf{pk}})\rangle$.
Check that $E_{\mathsf{pk}}, E^{\dagger}_{\mathsf{aux}} \in \mathsf{Supersingular}_p$, $r' + n_{\mathsf{bt}} \leq e_{\mathsf{rsp}}$, and $P_{\mathsf{chl}}, Q_{\mathsf{chl}} \in E'_{\mathsf{chl}}[2^{e_{\mathsf{rsp}} - n_{\mathsf{bt}} + 2}]$; otherwise, **return** 0.
if $r' > 0$ **then**
  if $[2^{e_{\mathsf{rsp}} - n_{\mathsf{bt}} + 1}]P_{\mathsf{chl}} \neq 0$ **then**
    $R \coloneqq [2^{(e_{\mathsf{rsp}} - n_{\mathsf{bt}} + 2) - r'}]P_{\mathsf{chl}}$.
  else if $[2^{e_{\mathsf{rsp}} - n_{\mathsf{bt}} + 1}]Q_{\mathsf{chl}} \neq 0$ **then**
    $R \coloneqq [2^{(e_{\mathsf{rsp}} - n_{\mathsf{bt}} + 2) - r'}]Q_{\mathsf{chl}}$.
  else
    return 0.
  Compute $\varphi : E'_{\mathsf{chl}} \to E_{\mathsf{rsp}}$ with $\ker(\varphi) = \langle R \rangle$.
  $P_{\mathsf{rsp}}, Q_{\mathsf{rsp}} \coloneqq \varphi(P_{\mathsf{chl}}), \varphi(Q_{\mathsf{chl}})$.
$P_{\mathsf{aux}}, Q_{\mathsf{aux}} \coloneqq \mathsf{DeterministicBasis}(E^{\dagger}_{\mathsf{aux}})$.
$K_1 \coloneqq (P_{\mathsf{rsp}}, [2^{e - (e_{\mathsf{rsp}} - n_{\mathsf{bt}} - r' + 2)}]P_{\mathsf{aux}})$.
$K_2 \coloneqq (Q_{\mathsf{rsp}}, [2^{e - (e_{\mathsf{rsp}} - n_{\mathsf{bt}} - r' + 2)}]Q_{\mathsf{aux}})$.
$F_1, F_2, \_ \leftarrow \mathsf{IsogenyEval2\text{-}2}(E_{\mathsf{rsp}}, E^{\dagger}_{\mathsf{aux}}, K_1, K_2, [])$.
if the computation of $\mathsf{IsogenyEval2\text{-}2}$ fails **or** $j(F_1) \neq \mathsf{com}$ **then**
  return 0.
return 1.

3 EUF-CMA-security from hard relations with hints

To help the weak honest-verifier zero-knowledge (wHVZK) simulator, we will give it a hint sampled from a *hint distribution*.

Definition 3.1 (Hint distribution). *Let $R \subseteq \mathcal{X} \times \mathcal{W}$ be a relation. A hint distribution $\mathcal{H}$ for R is a collection of distributions $\mathcal{H} = \{\mathcal{H}_x\}_{x \in \mathcal{X}}$, where $\mathcal{H}_x : \mathsf{HintSet}_x \to [0, 1]$ and the elements (i.e. the hints) of $\mathsf{HintSet}_x$ are efficiently representable in $|x|$. The distribution $\mathcal{H}_x$ need not be efficiently sampleable.*

We define wHVZK with hints as follows.

Definition 3.2 (Hint-assisted wHVZK). *Let $\Sigma = (\mathcal{P}, \mathcal{V})$ be a Σ-protocol for a relation R and let $\mathcal{H}$ be a hint distribution for R. We say that Σ is $\mathcal{H}$-hint-assisted wHVZK if there exists a PPT algorithm, called the simulator $\mathcal{S}$, such that for all $q = \mathsf{poly}(\lambda)$ and all PPT algorithms $\mathcal{A}$*

$$\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma,\mathcal{H},\mathcal{S}}(\mathcal{A}, q) \coloneqq \mathsf{Adv}^{\mathsf{dist}}\left[\mathsf{Real}_{\Sigma}(1^{\lambda}, q), \mathsf{HintSim}_{\Sigma}(1^{\lambda}, q, \mathcal{S})\right](\mathcal{A}) = \mathsf{negl}(\lambda),$$

where $\mathsf{Real}_{\Sigma}(1^{\lambda}, q)$ and $\mathsf{HintSim}_{\Sigma}(1^{\lambda}, q, \mathcal{S})$ are the distributions defined in Experiment 2.

In Section 2.3 and 2.4, we saw how to reduce the the hardness of the relation R to the unforgeability ($\mathsf{EUF\text{-}CMA}$) of $\mathsf{SIG}[\Sigma]$. Needing hints to simulate complicates the matter. In particular, we will see that the reduction will yield an adversary for R with access to these hints. We therefore need R to be a hard relation, even in the presence of hints.

Definition 3.3 (Hard relation with hints). *Let R be a hard relation and let $\mathcal{H}$ be a hint distribution for R. We call the pair $(R, \mathcal{H})$ a relation with hints. We say that it is a hard relation with hints if, in the following game, it holds for all PPT algorithms $\mathcal{A}$ and all $q = \mathsf{poly}(\lambda)$ that*

$$\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R,\mathcal{H})}(\mathcal{A}, q) \coloneqq \Pr\left[(x, w^*) \in R \;\middle|\; \begin{array}{r}(x, w) \leftarrow \mathsf{Gen}_R(1^{\lambda}),\\ h_1, \ldots, h_q \leftarrow \mathcal{H}_x,\\ w^* \leftarrow \mathcal{A}(x, h_1, \ldots, h_q)\end{array}\right] = \mathsf{negl}(\lambda).$$

These are all the hint-based security properties we will need. In addition, we will also permit Σ to be special sound with respect to a different but related relation.

Experiment 2: Hint-assisted wHVZK

$\mathsf{Real}_{\Sigma}(1^{\lambda}, q)$:

1: $(x, w) \leftarrow \mathsf{Gen}_R(1^{\lambda})$
2: **for** $i = 1$ **to** q **do**
3: $\quad (\mathsf{com}_i, \mathsf{state}_i) \leftarrow \mathcal{P}_1(x, w)$
4: $\quad \mathsf{chl}_i \xleftarrow{\$} \mathcal{C}$
5: $\quad \mathsf{rsp}_i \leftarrow \mathcal{P}_2(\mathsf{state}_i, \mathsf{chl}_i)$
6: $\quad \pi_i := (\mathsf{com}_i, \mathsf{chl}_i, \mathsf{rsp}_i)$
7: **return** $(x, \pi_1, \ldots, \pi_q)$

$\mathsf{HintSim}_{\Sigma}(1^{\lambda}, q, \mathcal{S})$:

1: $(x, w) \leftarrow \mathsf{Gen}_R(1^{\lambda})$
2: **for** $i = 1$ **to** q **do**
3: $\quad h_i \leftarrow \mathcal{H}_x$
4: $\quad \pi_i \leftarrow \mathcal{S}(x, h_i)$
5: **return** $(x, \pi_1, \ldots, \pi_q)$

Definition 3.4 (Soundness relation). *Let Σ be a Σ-protocol for the relation R. Consider some relation $\tilde{R} \subseteq \tilde{\mathcal{X}} \times \tilde{\mathcal{W}}$. We say that $\tilde{R}$ is compatible with R if $\mathcal{X} = \tilde{\mathcal{X}}$ and their instance generators sample statements with the same distribution.*

We say that Σ is special sound with respect to a compatible relation $\tilde{R}$ if given a statement x and two accepting transcripts $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$ and $(\mathsf{com}, \mathsf{chl}', \mathsf{rsp}')$ with $\mathsf{chl} \neq \mathsf{chl}'$, we can compute a witness w^* such that $(x, w^*) \in \tilde{R}$ in polynomial time in $|x|$. In this case, we say that $\tilde{R}$ is a soundness relation for Σ.

Observe that if R and $\tilde{R}$ are compatible relations and $\mathcal{H}$ is a hint distribution for R, then it is also a hint distribution for $\tilde{R}$. Additionally, if Σ is special sound with respect to $\tilde{R}$, then we can always define an instance generator for $\tilde{R}$ so that it is compatible with R. It samples $(x, w) \leftarrow \mathsf{Gen}_R(1^{\lambda})$, and then uses the special soundness of Σ_{SQI} to compute a witness w' such that $(x, w') \in \tilde{R}$.

3.1 Reducing the hard relations with hints to EUF-CMA

Let $(R, \mathcal{H})$ be a relation with hints and let Σ be a Σ-protocol for R. In this section, we identify the following conditions for $\mathsf{SIG}[\Sigma]$ to be EUF-CMA-secure:

1. Σ has high commitment min-entropy.
2. Σ is $\mathcal{H}$-hint-assisted wHVZK.
3. Σ is special sound with respect to a soundness relation $\tilde{R}$.
4. Σ has a challenge space $\mathcal{C}$ of exponential size in λ.
5. $(\tilde{R}, \mathcal{H})$ is a hard relation with hints.

We prove this by a chain of reductions. The first step is standard: given that Σ has high commitment min-entropy, we reduce the security against impersonation attacks (IMP-PA) of Σ to the EUF-CMA security of $\mathsf{SIG}[\Sigma]$ (for more details, see Lemma 2.2 in Section 2.4).

The next step will be to use the hint-assisted wHVZK. We will use it to simulate the transcript oracle OTrans in the IMP-PA game. For this purpose, we consider the intermediate game hint-IMP-PA in Definition 3.5. In this game, the adversary no longer has access to the transcript oracle OTrans. Instead, it is given hints from which it can simulate its own transcripts. Hence, we can view this game as a noninteractive variant of the IMP-PA game.

Definition 3.5 (hint-IMP-PA). *Let $(R, \mathcal{H})$ be a relation with hints and let Σ be a Σ-protocol for R. We say that Σ is secure under $\mathcal{H}$-hint-assisted passive impersonation attacks ($\mathcal{H}$-hint-IMP-PA) if for all $q = \mathsf{poly}(\lambda)$ and all PPT two-stage algorithms $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ it holds that*

$$\mathsf{Adv}^{\mathsf{hint\text{-}IMP\text{-}PA}}_{\Sigma,\mathcal{H}}(\mathcal{A}, q) := \Pr\left[\mathcal{H}\text{-}\mathsf{hint\text{-}IMP\text{-}PA}_{\Sigma}(\mathcal{A}, q) = 1\right] = \mathsf{negl}(\lambda),$$

where $\mathcal{H}$-hint-IMP-PA$_{\Sigma}$ is the game presented in Game 3.

Lemma 3.1. *Let $(R, \mathcal{H})$ be a relation with hints and let Σ be a Σ-protocol for the relation R. Additionally, let $\mathcal{S}$ be a $\mathcal{H}$-assisted simulator for Σ.*

For any two-stage PPT algorithm $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ against the IMP-PA of Σ, there exists a two-stage PPT algorithm $\mathcal{B}$ and a PPT algorithm $\mathcal{D}$ such that

$$\mathsf{Adv}^{\mathsf{IMP\text{-}PA}}_{\Sigma}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{hint\text{-}IMP\text{-}PA}}_{\Sigma,\mathcal{H}}(\mathcal{B}, q) + \mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma,\mathcal{H},\mathcal{S}}(\mathcal{D}, q),$$

where q is an upper-bound on the number of queries that $\mathcal{A}_1$ makes to OTrans.

Game 3: $\mathcal{H}$-hint-IMP-PA$_\Sigma(\mathcal{A}_1, \mathcal{A}_2, q)$

$(x, w) \leftarrow \mathsf{Gen}_R(1^\lambda)$
$h_1, \ldots, h_q \leftarrow \mathcal{H}_x$
$(\mathsf{com}, \mathsf{state}) \leftarrow \mathcal{A}_1(x, h_1, \ldots, h_q)$
$\mathsf{chl} \xleftarrow{\$} \mathcal{C}$
$\mathsf{rsp} \leftarrow \mathcal{A}_2(\mathsf{state}, \mathsf{chl})$
$b \coloneqq \mathsf{Ver}_\Sigma(x, \mathsf{com}, \mathsf{chl}, \mathsf{rsp}) = 1$
return b

Proof. We will use $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ and $\mathcal{S}$ to construct an adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{A}_2)$ for the hint-IMP-PA game. $\mathcal{B}$ has the same second stage algorithm as $\mathcal{A}$.

$\mathcal{B}_1$ takes as input a statement x sampled from $\mathsf{Gen}_R(1^\lambda)$ and q hints $h_1, \ldots, h_q \leftarrow \mathcal{H}_x$. It then runs $\mathcal{A}_1$ on input x. Whenever $\mathcal{A}_1$ makes a query to the transcript oracle OTrans, $\mathcal{B}_1$ responds using the simulator $\mathcal{S}$. For the ith query, $\mathcal{B}_1$ runs $(\mathsf{com}_i, \mathsf{chl}_i, \mathsf{rsp}_i) \leftarrow \mathcal{S}(x, h_i)$ and responds with $(\mathsf{com}_i, \mathsf{chl}_i, \mathsf{rsp}_i)$. When $\mathcal{A}_1$ outputs $(\mathsf{com}^*, \mathsf{state}^*)$, $\mathcal{B}_1$ outputs the same and terminates.

We use the hint-assisted wHVZK of Σ to relate the advantage of $\mathcal{B}$ in the hint-IMP-PA game to the advantage of $\mathcal{A}$ in the IMP-PA game. If there is a difference in success probability, we can construct a distinguisher $\mathcal{D}$. On input $(x, \boldsymbol{\pi})$ with $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_q)$, $\mathcal{D}$ runs the IMP-PA game for $\mathcal{A}$ and answers the ith query to OTrans with π_i. Finally, $\mathcal{D}$ outputs the same bit as the game. We have

$$\mathsf{Adv}^{\mathsf{IMP\text{-}PA}}_{\Sigma}(\mathcal{A}) = \Pr\left[1 \leftarrow \mathcal{D}(x, \boldsymbol{\pi}) \;\middle|\; (x, \boldsymbol{\pi}) \leftarrow \mathsf{Real}_\Sigma(1^\lambda, q)\right] \text{ and}$$
$$\mathsf{Adv}^{\mathsf{hint\text{-}IMP\text{-}PA}}_{\Sigma,\mathcal{H}}(\mathcal{B}, q) = \Pr\left[1 \leftarrow \mathcal{D}(x, \boldsymbol{\pi}) \;\middle|\; (x, \boldsymbol{\pi}) \leftarrow \mathsf{HintSim}_\Sigma(1^\lambda, q)\right].$$

Hence,

$$\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma,\mathcal{H},\mathcal{S}}(\mathcal{A}, q) = |\mathsf{Adv}^{\mathsf{IMP\text{-}PA}}_{\Sigma}(\mathcal{A}) - \mathsf{Adv}^{\mathsf{hint\text{-}IMP\text{-}PA}}_{\Sigma,\mathcal{H}}(\mathcal{B}, q)|. \qquad \square$$

Finally, we use the special soundness of Σ to reduce the hardness of $(\tilde{R}, \mathcal{H})$ to the hint-IMP-PA-security of Σ.

Lemma 3.2. *Let $(R, \mathcal{H})$ be a relation with hints and let Σ be a Σ-protocol for the relation R with challenge space $\mathcal{C}$. Additionally, assume Σ is special sound with respect to the soundness relation $\tilde{R}$.*

For any two-stage PPT algorithm $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ playing the hint-IMP-PA *game for Σ with q hints from $\mathcal{H}$, there exists an expected polynomial time algorithm $\mathcal{B}$ playing the game for $(\tilde{R}, \mathcal{H})$ such that*

$$\mathsf{Adv}^{\mathsf{hint\text{-}IMP\text{-}PA}}_{\Sigma,\mathcal{H}}(\mathcal{A}, q) \leq \mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(\tilde{R},\mathcal{H})}(\mathcal{B}, q) + \frac{1}{|\mathcal{C}|}.$$

Proof. By Lemma 2.1, Σ is knowledge-sound with respect to $\tilde{R}$, with knowledge error $1/|\mathcal{C}|$. Let $\mathcal{E}$ be the knowledge extractor.

Let us construct the adversary $\mathcal{B}$ against the hard relation with hints $(\tilde{R}, \mathcal{H})$. It gets as input a statement x generated by $\mathsf{Gen}_{\tilde{R}}(1^\lambda)$ and hints $h_1, \ldots, h_q \leftarrow \mathcal{H}$. This is exactly the same input distribution that $\mathcal{A}$ gets in its hint-IMP-PA game. $\mathcal{B}(x, h_1, \ldots, h_q)$ performs the following steps.

1. It defines $\mathcal{P}_1^*$ to be the algorithm $\mathcal{A}_1$ with inputs fixed to $x, h_1, \ldots, h_q$ (and security parameter 1^λ).
2. $\mathcal{P}^* \coloneqq (\mathcal{P}_1^*, \mathcal{A}_2)$.
3. $w^* \leftarrow \mathcal{E}^{\mathcal{P}^*}(x)$.
4. Output w^*.

The reason why the first step is needed is because $\mathcal{A}_1$ is not quite compatible with the interface that $\mathcal{E}$ expects. In particular, $\mathcal{A}$ expects a statement and q hints as input.

Let us analyze the success probability of $\mathcal{B}$. Let X and $\boldsymbol{H}$ be random variables distributed as the statement and the q hints in the inputs to $\mathcal{A}$ and $\mathcal{B}$. Let $A(X, \boldsymbol{H})$ be the event that $\mathcal{A}$ succeeds in the hint-IMP-PA game on input an input from X and $\boldsymbol{H}$. Similarly, let $B(X, \boldsymbol{H})$ be the event that $\mathcal{B}$ succeeds in outputting a witness for $\tilde{R}$.

Fix a statement $x \in \mathcal{X}$ and hints $\boldsymbol{h} = (h_1, \ldots, h_q) \in \mathsf{HintSet}^q_x$. By the knowledge soundness,

$$\Pr\left[B(X, H) \;\middle|\; X = x, H = \boldsymbol{h}\right] \geq \Pr\left[A(X, H) \;\middle|\; X = x, H = \boldsymbol{h}\right] - \frac{1}{|\mathcal{C}|}.$$

Then by linearity,

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(\tilde{R},\mathcal{H})}(\mathcal{B},q) &= \Pr\left[B(X,H)\right] \\
&= \sum_{x,\boldsymbol{h}} \Pr\left[X=x,H=\boldsymbol{h}\right]\Pr\left[B(X,H)\;\middle|\;X=x,H=\boldsymbol{h}\right] \\
&\geq \sum_{x,\boldsymbol{h}} \Pr\left[X=x,H=\boldsymbol{h}\right]\left(\Pr\left[A(X,H)\;\middle|\;X=x,H=\boldsymbol{h}\right]-\frac{1}{|\mathcal{C}|}\right) \\
&= \Pr\left[A(X,H)\right]-\frac{1}{|\mathcal{C}|} \\
&= \mathsf{Adv}^{\mathsf{hint\text{-}IMP\text{-}PA}}_{\Sigma,\mathcal{H}}(\mathcal{A},q)-\frac{1}{|\mathcal{C}|}.
\end{aligned}$$

Finally, the expected runtime of $\mathcal{B}$ is about the same as $\mathcal{E}$. □

Note that the above is not quite a reduction from the hard relation with hints $(\tilde{R},\mathcal{H})$ to the hint-IMP-PA-security of Σ. In particular, the definition of a hard relation with hints is only with respect to PPT algorithms. However, the algorithm constructed is just an *expected* polynomial time algorithm. To finish the reduction, we convert it to a PPT algorithm in a standard way.

Corollary 3.1. *Let $\mathcal{A}$ be an algorithm that runs in expected time t with success probability ε. Then there exists an algorithm $\mathcal{B}$ that runs in time at most $2t/\varepsilon$ and succeeds with probability at least $\varepsilon/2$. In particular, if t and ε^{-1} are polynomial in λ, then $\mathcal{B}$ runs in polynomial time in λ and has non-negligible success probability.*

Proof. $\mathcal{B}$ runs $\mathcal{A}$ but times out after $2t/\varepsilon-1$ steps. Let T be the random variable for the runtime of $\mathcal{A}$. By Markov's inequality,

$$\Pr\left[T\geq 2t/\varepsilon\right]\leq\frac{E[T]}{2t/\varepsilon}=\frac{\varepsilon}{2}.$$

Let $S_{\mathcal{A}}$ and $S_{\mathcal{B}}$ be the events that $\mathcal{A}$ and $\mathcal{B}$ succeeds, respectively.

$$\Pr\left[S_{\mathcal{B}}\right]=\Pr\left[S_{\mathcal{A}},T<2t/\varepsilon\right]\geq\Pr\left[S_{\mathcal{A}}\right]-\Pr\left[T\geq 2t/\varepsilon\right]\geq\varepsilon-\varepsilon/2=\varepsilon/2.$$

□

In conclusion, we obtain a reduction from the hard relation with hints $(\tilde{R},\mathcal{H})$ to the EUF-CMA security of $\mathsf{SIG}[\Sigma]$.

Theorem 1 (hint-rel reduces to EUF-CMA). *Let $(R,\mathcal{H})$ be a relation with hints. Let Σ be a Σ-protocol for the relation R that has challenge space $\mathcal{C}$ and is special sound with respect to a soundness relation $\tilde{R}$. Additionally, let $\mathcal{S}$ be a $\mathcal{H}$-hint-assisted simulator for Σ.*

For any PPT algorithm $\mathcal{A}$ against the EUF-CMA of $\mathsf{SIG}[\Sigma]$, there exists a PPT algorithm $\mathcal{B}$ and an expected polynomial time algorithm $\mathcal{E}$ such that

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma]}(\mathcal{A}) \leq (q+1)\cdot\left(\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(\tilde{R},\mathcal{H})}(\mathcal{E},s)+\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma,\mathcal{H},\mathcal{S}}(\mathcal{B},s)+\frac{1}{|\mathcal{C}|}\right) \\
+(q+s+1)s\cdot\mathsf{MinEnt}(\Sigma),
\end{aligned}$$

where q and s are upper-bounds on the number of queries that $\mathcal{A}$ makes to RO and OSign, respectively.

Proof. We simply combine the results in Section 3.

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma]}(\mathcal{A}) &\leq (q+1)\cdot\mathsf{Adv}^{\mathsf{IMP\text{-}PA}}_{\Sigma}(\mathcal{N})+(q+s+1)s\cdot\mathsf{MinEnt}(\Sigma) \\
&\leq (q+1)\cdot\left(\mathsf{Adv}^{\mathsf{hint\text{-}IMP\text{-}PA}}_{\Sigma,\mathcal{H}}(\mathcal{M},s)+\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma,\mathcal{H},\mathcal{S}}(\mathcal{B},s)\right) \\
&\quad+(q+s+1)s\cdot\mathsf{MinEnt}(\Sigma) \\
&\leq (q+1)\cdot\left(\left(\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(\tilde{R},\mathcal{H})}(\mathcal{E},s)+\frac{1}{|\mathcal{C}|}\right)+\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma,\mathcal{H},\mathcal{S}}(\mathcal{B},s)\right) \\
&\quad+(q+s+1)s\cdot\mathsf{MinEnt}(\Sigma),
\end{aligned}$$

where the first inequality uses Lemma 2.2, the second Lemma 3.1 and the third Lemma 3.2. □

4 Analyzing SQIsign in the Fiat–Shamir with hint framework

We now apply the framework introduced in the previous section to study the security of SQIsign.

4.1 Commitment min-entropy

We begin by analyzing the distribution of the commitment D_{com}. For every supersingular curve E, we define D_E^{mix} as the distribution on $j(\mathsf{Supersingular}_p)$ obtained by taking the codomain of a uniformly random cyclic isogeny $\varphi : E \to E'$ of degree N_{mix}. The commitment distribution is $D_{\mathsf{com}} = D_{E_0}^{\mathsf{mix}}$ and $\Delta(D_{\mathsf{com}}, S_j) \leq 1/(2\sqrt{p})$ by Proposition 2.4.

Next, we reduce the problem of distinguishing between D_{com} and S_j to the commitment min-entropy of Σ_{SQI} (as defined in Definition 2.4).

Lemma 4.1. $\mathsf{MinEnt}(\Sigma_{\mathsf{SQI}}) \leq p^{-1/2}$.

Proof. The commitment distribution D_{com} is independent of the statement or witness. By the definition of statistical distance, $\mathsf{MinEnt}(\Sigma_{\mathsf{SQI}}) \leq \Delta(D_{\mathsf{com}}, S_j) + \Pr\left[\mathsf{com}' = \mathsf{com} \;\middle|\; \mathsf{com}' \leftarrow S_j\right]$, for some commitment com. Then,

$$\Pr\left[\mathsf{com}' = \mathsf{com} \;\middle|\; \mathsf{com}' \leftarrow S_j\right] \leq \frac{12}{p-1} \leq \frac{1}{2\sqrt{p}},$$

using first that $|\operatorname{Aut}(E)| \geq 2$ for all $E \in \mathsf{Supersingular}_p$ and second that $p \geq 600$. We conclude using $\Delta(D_{\mathsf{com}}, S_j) \leq 1/(2\sqrt{p})$. □

4.2 Special soundness

While Σ_{SQI} is not special sound with respect to the original relation R_{SQI}, [BDD+24] showed that it is special sound with respect to a relation for the OneEnd_p problem.

Lemma 4.2 ([BDD+24, Theorem 17]). *If $e_{\mathsf{chl}} + e_{\mathsf{rsp}} \leq e$, then Σ_{SQI} is special-sound with respect to the soundness relation*

$$R_{\mathsf{OneEnd}} = \left\{ ((\mathsf{pp}, A_{\mathsf{pk}}), \alpha) \;\middle|\; \begin{array}{c} \mathsf{pp} \text{ public parameters, } A_{\mathsf{pk}} \in \mathbb{F}_{p^2}, \\ E_{\mathsf{pk}} \coloneqq E_{A_{\mathsf{pk}}} \in \mathsf{Supersingular}_p, \\ \alpha \in \operatorname{End}(E) \setminus \mathbb{Z} \text{ in efficient representation}, \\ \deg(\alpha) \leq p^4 \end{array} \right\}.$$

For completeness, we include the proof from [BDD+24]. It begins by proving two propositions.

Proposition 4.1. *From an accepting transcript $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$ for Σ_{SQI}, we can compute an efficient representation of an isogeny $\sigma : E_{\mathsf{com}} \to E_{\mathsf{chl}}$, where E_{com} is a curve such that $j(E_{\mathsf{com}}) = \mathsf{com}$ and $\deg(\sigma) \leq 2^{e_{\mathsf{rsp}}}$.*

Proof. We extract the isogeny by inspecting the verification algorithm (Algorithm 3). Write the challenge isogeny as

$$\varphi_{\mathsf{chl}} : E_{\mathsf{pk}} \xrightarrow{\varphi'_{\mathsf{chl}}} E'_{\mathsf{chl}} \xrightarrow{\tau} E_{\mathsf{chl}}$$

where φ'_{chl} is the non-backtracking challenge isogeny and τ is the last n_{bt} steps. The verification algorithm first computes a $2^{r'}$-isogeny $\varphi : E'_{\mathsf{chl}} \to E_{\mathsf{rsp}}$ and then a $(2^{e_{\mathsf{rsp}} - n_{\mathsf{bt}} - r'}, 2^{e_{\mathsf{rsp}} - n_{\mathsf{bt}} - r'})$-isogeny

$$\Phi : E_{\mathsf{rsp}} \times E_{\mathsf{aux}} \to F_1 \times F_2$$

with $j(F_1) = \mathsf{com}$. The latter is the efficient representation of an isogeny $\psi : E_{\mathsf{rsp}} \to F_1$ of degree at most $2^{e_{\mathsf{rsp}} - n_{\mathsf{bt}} - r'}$. Let $E_{\mathsf{com}} = F_1$. Then, we recover the isogeny $\sigma = \tau \circ \widehat{\varphi} \circ \widehat{\psi} : E_{\mathsf{com}} \to E_{\mathsf{chl}}$ of degree at most

$$\deg(\tau)\deg(\varphi)\deg(\psi) \leq 2^{n_{\mathsf{bt}}} \cdot 2^{r'} \cdot 2^{e_{\mathsf{rsp}} - n_{\mathsf{bt}} - r'}.$$

□

Proposition 4.2. *Let $\varphi_{\mathsf{chl},1} : E_{\mathsf{pk}} \to E_{\mathsf{chl},1}$ and $\varphi_{\mathsf{chl},2} : E_{\mathsf{pk}} \to E_{\mathsf{chl},2}$ be two distinct challenge isogenies from the same public curve E_{pk}. Then the largest integer $m \in \mathbb{Z}$ such that $[m]$ divides $\varphi_{\mathsf{chl},2} \circ \widehat{\varphi_{\mathsf{chl},1}}$ is strictly smaller than $2^{e_{\mathsf{chl}}}$.*

Proof. Let $\mathsf{chl}_1, \mathsf{chl}_2 \in \{0, \ldots, 2^{e_{\mathsf{chl}}} - 1\}$ be the distinct challenges defining $\varphi_{\mathsf{chl},1}$ and $\varphi_{\mathsf{chl},2}$. Let ψ be the backtracking component of $\widehat{\varphi_{\mathsf{chl},1}}$ and $\varphi_{\mathsf{chl},2}$ as in Lemma 2.3. This means that $\varphi_{\mathsf{chl},1} = \varphi'_{\mathsf{chl},1} \circ \psi$, $\varphi_{\mathsf{chl},2} = \varphi'_{\mathsf{chl},2} \circ \psi$ and $\varphi'_{\mathsf{chl},2} \circ \widehat{\varphi'_{\mathsf{chl},1}}$ is cyclic. Since $\varphi_{\mathsf{chl},2} \circ \widehat{\varphi_{\mathsf{chl},1}} = [\deg \psi] \circ \varphi'_{\mathsf{chl},2} \circ \widehat{\varphi'_{\mathsf{chl},1}}$ and $\varphi'_{\mathsf{chl},2} \circ \widehat{\varphi'_{\mathsf{chl},1}}$ is cyclic, $\deg \psi$ is the largest integer dividing $\varphi_{\mathsf{chl},2} \circ \widehat{\varphi_{\mathsf{chl},1}}$.

ψ is cyclic, so its kernel is generated by some point $K \in E_{\mathsf{pk}}$. Likewise, let $K_i = P_{\mathsf{pk}} + [\mathsf{chl}_i]Q_{\mathsf{pk}}$ be the generator of $\ker(\varphi_{\mathsf{chl},i})$ for $i = 1, 2$, where $(P_{\mathsf{pk}}, Q_{\mathsf{pk}})$ is the deterministic basis for $E_{\mathsf{pk}}[2^e]$. $K \in \langle K_1 \rangle \cap \langle K_2 \rangle$ implies that there exists integers $c_1, c_2 \in \{0, \ldots, 2^e - 1\}$ such that $K = c_1 K_1 = c_2 K_2$. Then

$$[c_1 - c_2]P_{\mathsf{pk}} + [c_1 \cdot \mathsf{chl}_1 - c_2 \cdot \mathsf{chl}_2]Q_{\mathsf{pk}} = 0.$$

From the first coefficient, we get $c_1 = c_2 \bmod 2^e$, meaning that $c_1 = c_2$. From the second, we get $c_1(\mathsf{chl}_1 - \mathsf{chl}_2) = 0 \bmod 2^e$, meaning that $2^e \mid c_1 \cdot (\mathsf{chl}_1 - \mathsf{chl}_2)$. Because $\mathsf{chl}_1 - \mathsf{chl}_2 \neq 0$ and is bounded by $-2^{e_{\mathsf{chl}}} < \mathsf{chl}_1 - \mathsf{chl}_2 < 2^{e_{\mathsf{chl}}}$, we must have that $2^{e - e_{\mathsf{chl}} + 1} | c_1$. Then $K \in E_{\mathsf{pk}}[2^{e_{\mathsf{chl}} - 1}]$ and $\deg(\psi) \leq 2^{e_{\mathsf{chl}} - 1}$. □

We are now finally ready to show that Σ_{SQI} is special sound with respect to R_{OneEnd}.

Proof. Let $(\mathsf{com}, \mathsf{chl}_1, \mathsf{rsp}_1)$ and $(\mathsf{com}, \mathsf{chl}_2, \mathsf{rsp}_2)$ be two accepting transcripts with $\mathsf{chl}_1 \neq \mathsf{chl}_2$. Let $\varphi_{\mathsf{chl},1} : E_{\mathsf{pk}} \to E_{\mathsf{chl},1}$ and $\varphi_{\mathsf{chl},2} : E_{\mathsf{pk}} \to E_{\mathsf{chl},2}$ be the challenge isogenies for chl_1 and chl_2. By Proposition 4.1 (and possibly by composing with an isomorphism), we obtain isogenies $\sigma_1 : E_{\mathsf{com}} \to E_{\mathsf{chl},1}$ and $\sigma_2 : E_{\mathsf{com}} \to E_{\mathsf{chl},2}$ such that $j(E_{\mathsf{com}}) = \mathsf{com}$ and each has degree at most $2^{e_{\mathsf{rsp}}}$. We obtain the endomorphism $\alpha = \widehat{\varphi_{\mathsf{chl},2}} \circ \sigma_2 \circ \widehat{\sigma_1} \circ \varphi_{\mathsf{chl},1} \in \mathrm{End}(E_{\mathsf{pk}})$. We have $\deg(\alpha) \leq 2^{2e + 2e_{\mathsf{rsp}}} \leq p^4$.

Assume for the sake of contradiction that $\alpha = [m]$ for some $m \in \mathbb{Z}$. Then

$$\begin{aligned} [m] \circ \varphi_{\mathsf{chl},2} \circ \widehat{\varphi_{\mathsf{chl},1}} &= \varphi_{\mathsf{chl},2} \circ [m] \circ \widehat{\varphi_{\mathsf{chl},1}} \\ &= \varphi_{\mathsf{chl},2} \circ (\widehat{\varphi_{\mathsf{chl},2}} \circ \sigma_2 \circ \widehat{\sigma_1} \circ \varphi_{\mathsf{chl},1}) \circ \widehat{\varphi_{\mathsf{chl},1}} \\ &= [\deg(\varphi_{\mathsf{chl},1}) \deg(\varphi_{\mathsf{chl},2})] \circ \sigma_2 \circ \widehat{\sigma_1}. \end{aligned} \tag{1}$$

Write $\varphi_{\mathsf{chl},2} \circ \widehat{\varphi_{\mathsf{chl},1}} = [2^f]\tau_1$ and $\sigma_2 \circ \widehat{\sigma_1} = [d']\tau_2$, where τ_1 and τ_2 are cyclic isogenies. Then the largest integer dividing (1) is $2^f m = d \deg(\varphi_{\mathsf{chl},1}) \deg(\varphi_{\mathsf{chl},2})$. Dividing, we obtain that $\tau_1 = \tau_2$.

Since $[2^f]$ divides $\varphi_{\mathsf{chl},2} \circ \widehat{\varphi_{\mathsf{chl},1}}$, by Proposition 4.2 we have $2^f < 2e_{\mathsf{chl}}$. Using that $e_{\mathsf{rsp}} + e_{\mathsf{chl}} \leq e$,

$$\deg(\tau_1) = \frac{\deg(\varphi_{\mathsf{chl},1}) \deg(\varphi_{\mathsf{chl},2})}{2^{2f}} > 2^{2(e - e_{\mathsf{chl}})} \geq 2^{2^{e_{\mathsf{rsp}}}}$$

At the same time, $\deg(\tau_2) \leq \deg(\sigma_1) \deg(\sigma_2) \leq 2^{2e_{\mathsf{rsp}}}$. This is a contradiction, as $\tau_1 = \tau_2$. Hence, we have $((\mathsf{pp}, A_{\mathsf{pk}}), \alpha) \in R_{\mathsf{OneEnd}}$. □

4.3 Hint-assisted wHVZK

We will prove that Σ_{SQI} is hint-assisted wHVZK with respect to the hint distribution $\mathcal{H}^{\mathsf{sim}}$ defined in Experiment 3. In essence, the hint provides the simulator with a response isogeny and auxiliary isogeny sampled from the correct distribution. Without knowledge of the endomorphism ring of E_{pk}, we only know how to efficiently sample random isogenies of *smooth* degree. Our $\mathcal{H}^{\mathsf{sim}}$-hint-assisted simulator is defined in Algorithm 4.

To show that Σ_{SQI} is $\mathcal{H}^{\mathsf{sim}}$-assisted wHVZK, we begin by comparing the distribution of real and simulated transcripts. The commitment $j(E_{\mathsf{com}})$ has the distribution D_{com} in real transcripts and the stationary distribution S_j in simulated transcripts. By Proposition 2.4, these distributions are statistically close. The challenge is also identically distributed in simulated and real transcripts. What remains is to compare the simulated and the real distribution of the response $(n_{\mathsf{bt}}, r', A^{\dagger}_{\mathsf{aux}}, P_{\mathsf{chl}}, Q_{\mathsf{chl}})$.

Lemma 4.3. *Let* $((\mathsf{pp}, A_{\mathsf{pk}}), I_{\mathsf{sk}}) \in R_{\mathsf{SQI}}$ *and let* $j_{\mathsf{com}} \in j(\mathsf{Supersingular}_p)$. *Consider transcripts for the statement* $(\mathsf{pp}, A_{\mathsf{pk}})$, $(j_{\mathsf{com}}, \mathsf{chl}, \mathsf{rsp})$ *with the commitment fixed to* j_{com}. *Then* $\deg(\varphi_{\mathsf{rsp}})$, n_{bt}, r', E'_{chl} *and* $E^{\dagger}_{\mathsf{aux}}$ *are identically distributed in real and simulated transcripts of this form.*

Proof. In the response algorithm, the ideal J is uniformly distributed among the equivalent ideals to $\overline{I_{\mathsf{com}}} \cdot I_{\mathsf{sk}} \cdot I_{\mathsf{chl}}$ of norm $< 2^{e_{\mathsf{rsp}}}$. Then, the response ideal I_{rsp} is the primitive component of J.

The simulator's response isogeny φ_{rsp} is the cyclic component of a uniformly random isogeny $E_{\mathsf{com}} \to E_{\mathsf{chl}}$ of norm $< 2^{e_{\mathsf{rsp}}}$. The ideal corresponding to φ_{rsp} depends on $j(E_{\mathsf{com}}) = j_{\mathsf{com}}$ but not on the representative

Experiment 3: Hint distribution $\mathcal{H}^{\mathsf{sim}}$ for SQISIGN

$\underline{D_{\mathsf{Chall}}(E)}$:

1: $s \xleftarrow{\$} \{0, \dots, 2^{e_{\mathsf{chl}}} - 1\}$.
2: $(P, Q) \coloneqq \mathsf{DeterministicBasis}(E)$.
3: Compute the isogeny $\varphi : E \to E'$ with $\ker(\varphi) = \langle P + [s]Q \rangle$ exactly as Σ_{SQI} does for the challenge isogeny.
4: **return** s, φ.

$\underline{D_{\mathsf{StatTarget}}(E)}$:

1: Sample an isogeny $\varphi : E \to E'$ such that
   i E' is distributed according to the stationary distribution S on $\mathsf{Supersingular}_p$.
   ii The conditional distribution of φ given E' is uniform among isogenies $E \to E'$ of degree $< 2^{e_{\mathsf{rsp}}}$.
2: Write $\varphi = [m] \circ \varphi'$ with $m \in \mathbb{Z}$ and φ' cyclic.
3: **return** an efficient representation of φ'.

$\underline{D_{\mathsf{Uniflsog}}(E, d)}$:

1: Sample an isogeny $\varphi : E \to E'$ uniformly among the cyclic isogenies from E of degree d to curves in $\mathsf{Supersingular}_p$.
2: **return** an efficient representation of φ.

$\underline{\mathcal{H}^{\mathsf{sim}}_E}$:

1: $s, \varphi_1 \leftarrow D_{\mathsf{Chall}}(E)$, where $\varphi_1 : E \to E_1$.
2: $\varphi_2 \leftarrow D_{\mathsf{StatTarget}}(E_1)$, where $\varphi_2 : E_1 \to E_2$.
3: Write $\deg(\varphi_2) = 2^n d'$ with d' odd.
4: $\varphi_3 \leftarrow D_{\mathsf{Uniflsog}}(E_2, 2^{e_{\mathsf{rsp}} - n} - d')$, where $\varphi_3 : E_2 \to E_3$.
5: **return** $h = (s, \varphi_2, \varphi_3)$.

E_{com}. Hence, the ideal has the same distribution as the prover's I_{rsp}. The values $\deg(\varphi_{\mathsf{rsp}})$ n_{bt} and r' are uniquely determined by the response ideal and the challenge, so their simulated distributions are the same as the real ones.

Fix a choice of $\deg(\varphi_{\mathsf{rsp}})$, n_{bt} and r' and write $\deg(\varphi_{\mathsf{rsp}}) = 2^n d'$ with d' odd. Given these values, the prover and simulator produce the same curve E'_{chl}, by computing $\varphi_{\mathsf{chl}} : E_{\mathsf{pk}} \to E'_{\mathsf{chl}}$ with $\ker(\varphi'_{\mathsf{chl}}) = \langle [2^{n_{\mathsf{bt}}}](P_{\mathsf{pk}} + [\mathsf{chl}]Q_{\mathsf{pk}}) \rangle$.

Let $d_{\mathsf{aux}} = 2^{e_{\mathsf{rsp}} - n} - d'$. In line 7 of the response algorithm, I''_{aux} is uniformly distributed among the primitive left $\mathcal{O}_0$-ideals of norm d_{aux}. Since the norm of I_{com} is coprime to d_{aux}, we can define the pushforward I_{aux} of I''_{aux} through I_{com}. It corresponds to the prover's auxiliary isogeny $\varphi_{\mathsf{aux}} : E_{\mathsf{com}} \to E_{\mathsf{aux}}$. The pushforward by I_{com} induces a bijection between the left $\mathcal{O}_0$ ideals of norm d_{aux} and the left $\mathcal{O}_{\mathsf{com}}$-ideals of norm d_{aux}. Hence, I_{aux} is uniformly distributed among the primitive left-$\mathcal{O}_{\mathsf{com}}$ ideals of norm d_{aux}. Given I_{aux}, the prover picks a uniformly random Montgomery A-invariant $A^{\dagger}_{\mathsf{aux}}$ for the representative of the codomain.

On the other hand, the simulator's auxiliary isogeny $\varphi_{\mathsf{aux}} : E_{\mathsf{com}} \to E^{\dagger}_{\mathsf{aux}}$ is uniformly distributed among the cyclic isogenies from E_{com} of degree d_{aux} to curves in $\mathsf{Supersingular}_p$. Hence, the corresponding ideal has the same distribution as the prover's I_{aux}. Fix the j-invariant j_{aux} of the codomain. The number of cyclic isogenies $E_{\mathsf{com}} \to E_{\mathsf{aux}}$ of degree d_{aux} is the same for any representative E_{aux} with $j(E_{\mathsf{aux}}) = j_{\mathsf{aux}}$. Hence, given j_{aux}, the simulator's $E^{\dagger}_{\mathsf{aux}}$ is uniformly distributed among the representatives for j_{aux}. We conclude that the simulated distribution of $A^{\dagger}_{\mathsf{aux}}$ is the same as the real one. □

Finally, we compare the distribution of the points $P_{\mathsf{chl}}, Q_{\mathsf{chl}}$. It turns out that their distribution depends on the automorphism group of E'_{chl}.

Lemma 4.4. *Let $((\mathsf{pp}, A_{\mathsf{pk}}), I_{\mathsf{sk}}) \in R_{\mathsf{SQI}}$ and let $j_{\mathsf{com}} \in j(\mathsf{Supersingular}_p)$. Consider transcripts for the statement $(\mathsf{pp}, A_{\mathsf{pk}})$ with the commitment fixed to j_{com} and $j(E'_{\mathsf{chl}}) \notin \{0, 1728\}$. Then P_{chl} and Q_{chl} are identically distributed in real and simulated transcripts of this form.*

Algorithm 4: The hint-assisted simulator $\mathcal{S}_{\mathsf{SQI}}(\mathsf{pp}, A_{\mathsf{pk}}, h)$

Input: The statement $(\mathsf{pp}, A_{\mathsf{pk}})$ and a hint $h = (s, \widehat{\varphi_{\mathsf{rsp}}}, \varphi_{\mathsf{aux}}) \in \mathsf{HintSet}^{\mathsf{sim}}_{E_{\mathsf{pk}}}$ with $\widehat{\varphi_{\mathsf{rsp}}} : E_{\mathsf{chl}} \to E_{\mathsf{com}}$ and $\varphi_{\mathsf{aux}} : E_{\mathsf{com}} \to E^{\dagger}_{\mathsf{aux}}$.

Output: Transcript of Σ_{SQI}.

1: $\mathsf{com} \coloneqq j(E_{\mathsf{com}})$ and $\mathsf{chl} \coloneqq s$.
2: $(P_{\mathsf{pk}}, Q_{\mathsf{pk}}) \coloneqq \mathsf{DeterministicBasis}(E_{\mathsf{pk}})$.
3: Compute $\varphi_{\mathsf{chl}} : E_{\mathsf{pk}} \to E_{\mathsf{chl}}$ with $\ker(\varphi_{\mathsf{chl}}) = \langle P_{\mathsf{pk}} + [s]Q_{\mathsf{pk}} \rangle$, exactly as Σ_{SQI} does for the challenge isogeny.
4: Compute the backtracking component of φ_{chl} and $\widehat{\varphi_{\mathsf{rsp}}}$ with Lemma 2.3, obtaining ψ_{bt} and $\widehat{\psi_{\mathsf{rsp}}}$ such that

$$\widehat{\varphi_{\mathsf{rsp}}} : E_{\mathsf{chl}} \xrightarrow{\psi_{\mathsf{bt}}} F \xrightarrow{\widehat{\psi_{\mathsf{rsp}}}} E_{\mathsf{com}} \qquad \text{and} \qquad \ker(\psi_{\mathsf{bt}}) = \ker(\widehat{\varphi_{\mathsf{chl}}}) \cap \ker(\widehat{\varphi_{\mathsf{rsp}}}).$$

5: Let $n_{\mathsf{bt}} \in \mathbb{Z}$ be such that $\deg(\psi_{\mathsf{bt}}) = 2^{n_{\mathsf{bt}}}$.
6: Let r' be the largest integer such that $2^{r'} \mid \deg(\psi_{\mathsf{rsp}})$.
7: Use Lemma 2.3 to compute efficient representations of ψ_{rsp} and $\widehat{\varphi_{\mathsf{aux}}}$.
8: Compute $\varphi'_{\mathsf{chl}} : E_{\mathsf{pk}} \to E'_{\mathsf{chl}}$ with $\ker(\varphi'_{\mathsf{chl}}) = \langle [2^{n_{\mathsf{bt}}}](P_{\mathsf{pk}} + [s]Q_{\mathsf{pk}}) \rangle$, as the verifier would.
9: Compute an isomorphism $\iota_{\mathsf{chl}} : F \to E'_{\mathsf{chl}}$.
10: $\varphi'_{\mathsf{rsp}} \coloneqq \iota_{\mathsf{chl}} \circ \psi_{\mathsf{rsp}} : E_{\mathsf{com}} \to E'_{\mathsf{chl}}$.
11: $(P_{\mathsf{aux}}, Q_{\mathsf{aux}}) \coloneqq [2^{e-(e_{\mathsf{rsp}}-n_{\mathsf{bt}}+2)}] \cdot \mathsf{DeterministicBasis}(E^{\dagger}_{\mathsf{aux}})$.
12: $P_{\mathsf{chl}} \coloneqq [\deg(\varphi_{\mathsf{aux}})^{-1}]\varphi'_{\mathsf{rsp}} \circ \widehat{\varphi_{\mathsf{aux}}}(P_{\mathsf{aux}})$.
13: $Q_{\mathsf{chl}} \coloneqq [\deg(\varphi_{\mathsf{aux}})^{-1}]\varphi'_{\mathsf{rsp}} \circ \widehat{\varphi_{\mathsf{aux}}}(Q_{\mathsf{aux}})$.
14: **if** $-P_{\mathsf{chl}}$ is lexicographically smaller than P_{chl} **then**
15: $\quad P_{\mathsf{chl}}, Q_{\mathsf{chl}} \coloneqq -P_{\mathsf{chl}}, -Q_{\mathsf{chl}}$
16: $\mathsf{rsp} \coloneqq (n_{\mathsf{bt}}, r', E^{\dagger}_{\mathsf{aux}}, P_{\mathsf{chl}}, Q_{\mathsf{chl}})$.
17: **return** $(\mathsf{com}, \mathsf{chl}, \mathsf{rsp})$.

Proof. At line 20 in the response algorithm, we have

$$R_{\mathsf{chl}} = \iota_{\mathsf{chl}} \circ \varphi^{\mathsf{even}}_{\mathsf{rsp}} \circ \varphi^{\mathsf{odd}}_{\mathsf{rsp}}(P_{\mathsf{com}}), \qquad R_{\mathsf{aux}} = \iota_{\mathsf{aux}} \circ \varphi_{\mathsf{aux}}(P_{\mathsf{com}}),$$
$$S_{\mathsf{chl}} = \iota_{\mathsf{chl}} \circ \varphi^{\mathsf{even}}_{\mathsf{rsp}} \circ \varphi^{\mathsf{odd}}_{\mathsf{rsp}}(Q_{\mathsf{com}}), \qquad S_{\mathsf{aux}} = \iota_{\mathsf{aux}} \circ \varphi_{\mathsf{aux}}(Q_{\mathsf{com}}),$$

where $\iota_{\mathsf{chl}} : E^{\mathsf{odd}}_{\mathsf{rsp}} \to E'_{\mathsf{chl}}$ and $\iota_{\mathsf{aux}} : E_{\mathsf{aux}} \to E^{\dagger}_{\mathsf{aux}}$ are the isomorphisms computed. Since $\deg(\varphi_{\mathsf{aux}})$ is odd and $P_{\mathsf{com}}, Q_{\mathsf{com}} \in E_{\mathsf{com}}[2^e]$, $(P_{\mathsf{com}}, Q_{\mathsf{com}})$ is the image of $(R_{\mathsf{aux}}, S_{\mathsf{aux}})$ through $[\deg(\varphi_{\mathsf{aux}})^{-1}] \circ \widehat{\varphi_{\mathsf{aux}}} \circ \widehat{\iota_{\mathsf{aux}}}$. It follows that

$$R_{\mathsf{chl}} = [\deg(\varphi_{\mathsf{aux}})^{-1}] \circ \iota_{\mathsf{chl}} \circ \varphi^{\mathsf{even}}_{\mathsf{rsp}} \circ \varphi^{\mathsf{odd}}_{\mathsf{rsp}} \circ \widehat{\varphi_{\mathsf{aux}}} \circ \widehat{\iota_{\mathsf{aux}}}(R_{\mathsf{aux}}) \text{ and}$$
$$S_{\mathsf{chl}} = [\deg(\varphi_{\mathsf{aux}})^{-1}] \circ \iota_{\mathsf{chl}} \circ \varphi^{\mathsf{even}}_{\mathsf{rsp}} \circ \varphi^{\mathsf{odd}}_{\mathsf{rsp}} \circ \widehat{\varphi_{\mathsf{aux}}} \circ \widehat{\iota_{\mathsf{aux}}}(S_{\mathsf{aux}}).$$

After line 23 the same relationship holds for $(P_{\mathsf{chl}}, Q_{\mathsf{chl}})$ as the evaluation of $(P_{\mathsf{aux}}, Q_{\mathsf{aux}})$. Finally, the lines 24-25 makes a deterministic choice of sign for P_{chl} and Q_{chl}. The simulator computes $P_{\mathsf{chl}}, Q_{\mathsf{chl}}$ in the same manner, by pushing $P_{\mathsf{aux}}, Q_{\mathsf{aux}}$ through $[\deg(\varphi_{\mathsf{aux}})^{-1}] \circ \varphi'_{\mathsf{rsp}} \circ \widehat{\varphi_{\mathsf{aux}}}$ and then choosing the sign in the same way as the prover.

In the proof of Lemma 4.3, we saw that E'_{chl}, $E^{\dagger}_{\mathsf{aux}}$ and the ideals corresponding to $\varphi_{\mathsf{rsp}}, \varphi_{\mathsf{aux}}$ and φ_{chl} all have the same real and simulated distribution. These values determine the isogeny $E^{\dagger}_{\mathsf{aux}} \to E'_{\mathsf{chl}}$, up to post-composition with automorphisms. When $j(E'_{\mathsf{chl}}) \notin \{0, 1728\}$, the only automorphisms of E'_{chl} are $[\pm 1]$ [Sil86, Theorem III.10.1]. Then the automorphisms of E'_{chl} only change the sign of the evaluation of the deterministic basis $(P_{\mathsf{aux}}, Q_{\mathsf{aux}})$. If the prover and simulator use isogenies $E^{\dagger}_{\mathsf{aux}} \to E'_{\mathsf{chl}}$ that agree up to post-composition with automorphisms, they compute the same $P_{\mathsf{chl}}, Q_{\mathsf{chl}}$, because they pick the same sign. So in this case, P_{chl} and Q_{chl} have the same real and simulated distribution. □

The case when $j(E'_{\mathsf{chl}})$ is 0 or 1728 must be handled separately. In this case, the real distribution of $P_{\mathsf{chl}}, Q_{\mathsf{chl}}$ might be distinguishable from the simulated distribution. However, we argue that this case will only happen with negligible probability under the assumption that the $\mathsf{EndRing}_p$ problem is hard.

Lemma 4.5. *Let $((\mathsf{pp}, A_{\mathsf{pk}}), I_{\mathsf{sk}}) \in R_{\mathsf{SQI}}$. Let $J^{\mathsf{real}}_{\mathsf{chl}'}$ and $J^{\mathsf{sim}}_{\mathsf{chl}'}$ be random variables for the distribution of $j(E'_{\mathsf{chl}})$ in real and simulated transcripts, respectively. There exists a PPT adversary $\mathcal{B}$ against the hard relation R_{SQI} such that*

$$\Pr\left[J^{\mathsf{real}}_{\mathsf{chl}'} \in \{0, 1728\}\right] \leq \mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{B}) \text{ and } \Pr\left[J^{\mathsf{sim}}_{\mathsf{chl}'} \in \{0, 1728\}\right] \leq \mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{B}).$$

Proof. The curves with j-invariant 0 and 1728 have known endomorphism ring [McM14]. If the 2-isogeny walk of the challenge isogeny hits one of these curves with good probability, we can construct an efficient adversary $\mathcal{B}$ against R_{SQI}.

On input a statement $(\mathsf{pp}, A_{\mathsf{pk}})$ sampled from $\mathsf{Gen}_{R_{\mathsf{SQI}}}(1^\lambda)$, $\mathcal{B}$ does the following:

1. Sample $\mathsf{chl} \leftarrow \{0, \dots, 2^{e_{\mathsf{chl}}}-1\}$ and set $K = P_{\mathsf{pk}} + [\mathsf{chl}]Q_{\mathsf{pk}}$, where $(P_{\mathsf{pk}}, Q_{\mathsf{pk}}) \coloneqq \mathsf{DeterministicBasis}(E_{\mathsf{pk}})$.
2. Compute the 2-isogeny walk of the challenge isogeny with kernel generated by K,

$$\varphi_{\mathsf{chl}} : E_{\mathsf{pk}} \xrightarrow{\varphi_1} E_1 \xrightarrow{\varphi_2} E_2 \xrightarrow{\varphi_3} \dots \xrightarrow{\varphi_n} E_n = E_{\mathsf{chl}}.$$

3. If none of $E_1, \dots, E_n$ have j-invariant 0 or 1728, abort. Else, let i be such that $j(E_i) \in \{0, 1728\}$.
4. Compute the generator K' for the dual isogeny $\psi = \widehat{\varphi_1} \circ \dots \circ \widehat{\varphi_i} : E_i \to E_{\mathsf{pk}}$. This is done by pushing $[2^{e-i}]P_{\mathsf{pk}}$ and $[2^{e-i}]Q_{\mathsf{pk}}$ through $\varphi_i \circ \dots \circ \varphi_1$, and letting K' be the evaluated point that has order 2^i.
5. With K' and a basis in efficient representation for $\mathrm{End}(E_i) \cong \mathcal{O}_i$, use [DLRW24, Algorithm 9] to compute the ideal I_ψ corresponding to ψ.
6. If $j(E_i) = 1728$, let $I \coloneqq I_\psi$. Else, compute an ideal I_0 connecting $\mathcal{O}_0$ and $\mathcal{O}_i$ with [KV10, Algorithm 3.5], and let $I \coloneqq I_0 \cdot I_\psi$.
7. Output I.

If $\mathcal{B}$ does not abort in step 3, it outputs a left $\mathcal{O}_0$-ideal I with $\mathcal{O}_R(J) \cong \mathrm{End}(E_{\mathsf{pk}})$. Then $((\mathsf{pp}, A_{\mathsf{pk}}), J) \in R_{\mathsf{SQI}}$.

The challenge isogeny φ_{chl} has the same distribution in real and simulated transcripts. The probability that $j(E'_{\mathsf{chl}}) \in \{0, 1728\}$ is at most the probability that some curve on the walk of φ_{chl} has j-invariant 0 or 1728. The latter is exactly the probability that $\mathcal{B}$ succeeds. □

We are now finally ready to prove that Σ_{SQI} is $\mathcal{H}^{\mathsf{sim}}$-hint-assisted wHVZK.

Lemma 4.6 (Computational hint-assisted wHVZK). *Let $\mathcal{S}_{\mathsf{SQI}}$ be the $\mathcal{H}^{\mathsf{sim}}$-hint-assisted simulator defined in Algorithm 4. For any $q = \mathsf{poly}(\lambda)$ and any PPT adversary $\mathcal{A}$ against the hint-assisted wHVZK of Σ_{SQI} with q hints, there exists a PPT algorithm $\mathcal{B}$ such that*

$$\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma_{\mathsf{SQI}}, \mathcal{H}^{\mathsf{sim}}, \mathcal{S}_{\mathsf{SQI}}}(\mathcal{A}, q) \leq \mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{B}) + 2q \cdot \Delta(D_{\mathsf{com}}, S_j).$$

Proof. Recall that $\mathcal{A}$ takes as input a statement $(\mathsf{pp}, A_{\mathsf{pk}})$ sampled from $\mathsf{Gen}_R(1^\lambda)$ and q independently generated transcripts for this statement. The adversary tries to guess whether the transcripts come from the real or simulated distribution. The distinguishing advantage of $\mathcal{A}$ is

$$\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma_{\mathsf{SQI}}, \mathcal{H}^{\mathsf{sim}}, \mathcal{S}_{\mathsf{SQI}}}(\mathcal{A}) = \left| \Pr\left[b^{\mathsf{real}} = 1\right] - \Pr\left[b^{\mathsf{sim}} = 1\right] \right|,$$

where $b^{\mathsf{real}} \coloneqq \mathcal{A}(\mathsf{Real}_{\Sigma_{\mathsf{SQI}}}(1^\lambda, q))$ and $b^{\mathsf{sim}} \coloneqq \mathcal{A}(\mathsf{HintSim}_{\Sigma_{\mathsf{SQI}}}(1^\lambda, q, \mathcal{S}_{\mathsf{SQI}}))$ are random variables. The goal is to upper bound the advantage of $\mathcal{A}$.

We begin by considering the commitment. The commitment is a j-invariant in $V \coloneqq j(\mathsf{Supersingular}_p)$. The commitment is distributed as D_{com} in real transcripts, and as S_j in simulated transcripts. $\mathcal{A}$ is given q independently sampled commitments from one of the distributions. When $\mathcal{A}$ gets real transcripts, let $\boldsymbol{J}^{\mathsf{real}}_{\mathsf{com}} = \left(J^{\mathsf{real}}_{\mathsf{com},1}, \dots, J^{\mathsf{real}}_{\mathsf{com},q}\right)$ be the random variable for the q commitments. Similarly, when $\mathcal{A}$ gets simulated transcripts, let $\boldsymbol{J}^{\mathsf{sim}}_{\mathsf{com}} = \left(J^{\mathsf{sim}}_{\mathsf{com},1}, \dots, J^{\mathsf{sim}}_{\mathsf{com},q}\right)$ be the random variable for the q commitments. For $\boldsymbol{j} \in V^q$, we let $p^{\mathsf{real}}_{\boldsymbol{j}} = \Pr\left[\boldsymbol{J}^{\mathsf{real}}_{\mathsf{com}} = \boldsymbol{j}\right]$ and $p^{\mathsf{sim}}_{\boldsymbol{j}} = \Pr\left[\boldsymbol{J}^{\mathsf{sim}}_{\mathsf{com}} = \boldsymbol{j}\right]$. Using the statistical distance between

D_{com} and S_j,

$$\begin{aligned}
\Pr\left[b^{\mathsf{real}}=1\right] &= \sum_{\boldsymbol{j}\in V^q} p_{\boldsymbol{j}}^{\mathsf{real}} \Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&= \sum_{\boldsymbol{j}\in V^q} \left(p_{\boldsymbol{j}}^{\mathsf{real}} - p_{\boldsymbol{j}}^{\mathsf{sim}} + p_{\boldsymbol{j}}^{\mathsf{sim}}\right) \cdot \Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&\leq \sum_{\boldsymbol{j}\in V^q} \left(\left|p_{\boldsymbol{j}}^{\mathsf{real}} - p_{\boldsymbol{j}}^{\mathsf{sim}}\right| + p_{\boldsymbol{j}}^{\mathsf{sim}}\right) \cdot \Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&\leq \sum_{\boldsymbol{j}\in V^q} \left|p_{\boldsymbol{j}}^{\mathsf{real}} - p_{\boldsymbol{j}}^{\mathsf{sim}}\right| + \sum_{\boldsymbol{j}\in V^q} p_{\boldsymbol{j}}^{\mathsf{sim}} \Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&= 2\Delta(\boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}, \boldsymbol{J}_{\mathsf{com}}^{\mathsf{sim}}) + \sum_{\boldsymbol{j}\in V^q} p_{\boldsymbol{j}}^{\mathsf{sim}} \Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&\leq 2q\Delta(D_{\mathsf{com}}, S_j) + \sum_{\boldsymbol{j}\in V^q} p_{\boldsymbol{j}}^{\mathsf{sim}} \Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right],
\end{aligned} \tag{2}$$

where the last step uses Proposition 2.2.

Next, we focus on $\Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right]$. It is the probability that $\mathcal{A}$ outputs 1 when it is given real transcripts with $\boldsymbol{j}$ as the commitments. When the commitments are fixed, we have seen in Lemma 4.3 and Lemma 4.4 that real and simulated transcripts are identically distributed, unless $j(E'_{\mathsf{chl}}) \in \{0, 1728\}$.

When $\mathcal{A}$ is given real transcripts, let G^{real} be the "good" event that all of the transcripts have $j(E'_{\mathsf{chl}}) \notin \{0, 1728\}$. Let G^{sim} be the corresponding event for the simulated transcripts. By Lemma 4.3, when the commitment is fixed, $j(E'_{\mathsf{chl}})$ has the same distribution in real and simulated transcripts. Hence, for all $\boldsymbol{j} \in V^q$,

$$\Pr\left[G^{\mathsf{real}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] = \Pr\left[G^{\mathsf{sim}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{sim}}=\boldsymbol{j}\right].$$

Furthermore, given that the good event has occurred, real and simulated transcripts are identically distributed, meaning that

$$\Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}, G^{\mathsf{real}}\right] = \Pr\left[b^{\mathsf{sim}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{sim}}=\boldsymbol{j}, G^{\mathsf{sim}}\right].$$

With these observations, we obtain that

$$\begin{aligned}
&\Pr\left[b^{\mathsf{real}}=1 \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&= \Pr\left[b^{\mathsf{real}}=1, G^{\mathsf{real}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] + \Pr\left[b^{\mathsf{real}}=1, \neg G^{\mathsf{real}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&\leq \Pr\left[b^{\mathsf{real}}=1, G^{\mathsf{real}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] + \Pr\left[\neg G^{\mathsf{real}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{real}}=\boldsymbol{j}\right] \\
&= \Pr\left[b^{\mathsf{sim}}=1, G^{\mathsf{sim}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{sim}}=\boldsymbol{j}\right] + \Pr\left[\neg G^{\mathsf{sim}} \;\middle|\; \boldsymbol{J}_{\mathsf{com}}^{\mathsf{sim}}=\boldsymbol{j}\right].
\end{aligned}$$

Combining this with (2), we get that

$$\begin{aligned}
\Pr\left[b^{\mathsf{real}}=1\right] &\leq 2q\Delta(D_{\mathsf{com}}, S_j) + \Pr\left[b^{\mathsf{sim}}=1, G^{\mathsf{sim}}\right] + \Pr\left[\neg G^{\mathsf{sim}}\right] \\
&\leq 2q\Delta(D_{\mathsf{com}}, S_j) + \Pr\left[b^{\mathsf{sim}}=1\right] + \Pr\left[\neg G^{\mathsf{sim}}\right].
\end{aligned}$$

What remains is to upper bound the probability that G^{sim} does not occur, i.e. the event that one of the q simulated transcripts has $j(E'_{\mathsf{chl}}) \in \{0, 1728\}$. For a single transcript, the probability can be upper bounded by the advantage of a PPT adversary $\mathcal{N}$ against the relation R_{SQI}, by Lemma 4.5. $\mathcal{N}$ first perfectly simulates the challenge isogeny, and then tries to use it to compute a witness for the statement. Consider the adversary $\mathcal{B}$ that runs $\mathcal{N}$ q times with input E_{pk}, using a fresh random tape each time. Since $q = \mathsf{poly}(\lambda)$, $\mathcal{B}$ is PPT. Furthermore,

$$\Pr\left[\neg G^{\mathsf{sim}}\right] \leq \mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{B}).$$

We conclude that

$$\Pr\left[b^{\mathsf{real}}=1\right] - \Pr\left[b^{\mathsf{sim}}=1\right] \leq 2q \cdot \Delta(D_{\mathsf{com}}, S_j) + \mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{B}).$$

By a symmetric argument, $\Pr\left[b^{\mathsf{sim}}=1\right] - \Pr\left[b^{\mathsf{real}}=1\right]$ has the same bound. □

4.4 Applying the framework

We have shown that Σ_{SQI} has the properties required to apply the framework of Section 3. It has high commitment min-entropy, is $\mathcal{H}^{\mathsf{sim}}$-hint-assisted wHVZK, is special sound with respect to the soundness relation R_{OneEnd} and has an exponentially large challenge space in λ.[10] By Theorem 1, we obtain a reduction from the relation R_{OneEnd} with $\mathcal{H}^{\mathsf{sim}}$-hints to the EUF-CMA of $\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]$.

Theorem 2. *For any PPT algorithm $\mathcal{A}$ against the* EUF-CMA *of* $\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]$*, there exists an expected polynomial time algorithm $\mathcal{E}$ with*

$$\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]}(\mathcal{A}) \leq (q+1)\cdot\left(2\cdot\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}},\mathcal{H}^{\mathsf{sim}})}(\mathcal{E},s)+2^{-e_{\mathsf{chl}}}\right)+\frac{(2q+s+2)s}{\sqrt{p}},$$

where q and s are upper bounds on the number of queries that $\mathcal{A}$ makes to RO *and* OSign*, respectively.*

Proof. Σ_{SQI} is special sound w.r.t. the soundness relation R_{OneEnd} and has the $\mathcal{H}^{\mathsf{sim}}$-hint-assisted simulator $\mathcal{S}_{\mathsf{SQI}}$. By Theorem 1, there exists a PPT algorithm $\mathcal{N}$ and an expected polynomial time algorithm $\mathcal{E}_1$ such that

$$\begin{aligned}\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]}(\mathcal{A}) \leq (q+1)\cdot&\left(\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}},\mathcal{H}^{\mathsf{sim}})}(\mathcal{E}_1,s)+\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma_{\mathsf{SQI}},\mathcal{H}^{\mathsf{sim}},\mathcal{S}_{\mathsf{SQI}}}(\mathcal{N},s)\right)\\&+\frac{q+1}{|\mathcal{C}|}+(q+s+1)s\cdot\mathsf{MinEnt}(\Sigma_{\mathsf{SQI}}).\end{aligned}\tag{3}$$

We have $|\mathcal{C}|=2^{e_{\mathsf{chl}}}$. By Lemma 4.1, $\mathsf{MinEnt}(\Sigma_{\mathsf{SQI}})\leq 1/\sqrt{p}$. For the hint-assisted wHVZK adversary $\mathcal{N}$, we use Lemma 4.6 to construct a PPT algorithm $\mathcal{M}$ such that by Proposition 2.4,

$$\mathsf{Adv}^{\mathsf{hint\text{-}wHVZK}}_{\Sigma_{\mathsf{SQI}},\mathcal{H}^{\mathsf{sim}},\mathcal{S}_{\mathsf{SQI}}}(\mathcal{N},s)\leq\mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{M})+2s\cdot\Delta(D_{\mathsf{com}},S_j)\leq\mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{M})+\frac{s}{\sqrt{p}}.\tag{4}$$

Next, we use $\mathcal{M}$ to construct an algorithm $\mathcal{E}_2$ for the relation R_{OneEnd} with hints. $\mathcal{E}_2$ does not use the hints, instead it runs $\mathcal{M}$ to compute I_{sk}, and if it succeeds, uses the special soundness to compute a witness for R_{OneEnd}. Let $\mathcal{E}$ be the algorithm with the maximal success probability out of $\mathcal{E}_1$ and $\mathcal{E}_2$, such that

$$\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}},\mathcal{H}^{\mathsf{sim}})}(\mathcal{E}_1,s)+\mathsf{Adv}^{\mathsf{rel}}_{R_{\mathsf{SQI}}}(\mathcal{M})\leq 2\cdot\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}},\mathcal{H}^{\mathsf{sim}})}(\mathcal{E},s).\tag{5}$$

We conclude by plugging (4), (5) and $\mathsf{MinEnt}(\Sigma_{\mathsf{SQI}})\leq 1/\sqrt{p}$ into (3). □

5 Reducing hint-EndRing to the EUF-CMA of SQIsign

We begin in Section 5.1 by introducing a new hint distribution $\mathcal{H}^{\mathsf{unif}}$ and showing it is pushable, i.e. given a 2^n-isogeny $\sigma: E\to E'$ and a hint $h\leftarrow\mathcal{H}^{\mathsf{unif}}_E$ for E, we can push it through σ to get a hint for E' distributed according to $\mathcal{H}^{\mathsf{unif}}_{E'}$. Then, we introduce a new indistinguishability assumption, and we argue why we believe it to be hard. Lastly, in Section 5.3, we define the variant of the EndRing problem with $\mathcal{H}^{\mathsf{unif}}$ hints, which we call the q-hint-EndRing$_p$ problem, and reduce it to the EUF-CMA of SQIsign.

5.1 The pushable hint distribution

We introduce a new hint distribution $\mathcal{H}^{\mathsf{unif}}$: rather than sampling a random curve and generating a connecting isogeny, the new hint distribution $\mathcal{H}^{\mathsf{unif}}_E$ samples random isogenies directly from E, as defined in Algorithm 5. The main property of the new distribution is that, unlike the previous hint distribution $\mathcal{H}^{\mathsf{sim}}$, it is pushable: we present the pushing algorithm in Algorithm 6, and we prove the output of Algorithm 6 is distributed as $\mathcal{H}^{\mathsf{unif}}_{E'}$ in the following lemma.

[10] The challenge space, in the round-2 SQIsign submission to the NIST standardization process, is $2^{e_{\mathsf{chl}}}$-large, which is a few bits short of being 2^λ large. This could be exploited by an attacker: given a valid signature σ for a message msg, one can use a second preimage attack on the hash function used to generate a challenge to obtain a second message msg′. This leads to the exact same challenge in the SQIsign identification protocol, so that σ is also a valid signature for msg′. This would yield a forgery attack with complexity $O(2^{e_{\mathsf{chl}}})$. To make up for the security gap, SQIsign uses a hash function that consists of $2^{\lambda-e_{\mathsf{chl}}}$ iterations of a standard hash function, SHAKE256. This technique, commonly known as "grinding" in the literature, brings the attack cost to the desired $O(2^\lambda)$: finding a collision in the hash function requires $O(2^{e_{\mathsf{chl}}})$ attempts, each of which consists of an evaluation of the hash function, which has a cost of at least $2^{\lambda-e_{\mathsf{chl}}}$. Thus, the total attack cost is $O(2^\lambda)$.

Algorithm 5: The pushable hint distribution

$\mathcal{H}^{\mathsf{unif}}_E$:

1: Sample an integer d from a weighted distribution on the interval $[1, 2^{e_{\mathsf{rsp}}}]$ where each integer n, with prime factorization $n = \prod_{i=1}^{t} p_i^{e_i}$, has weight $\prod_{i=1}^{t}(p_i+1)^{e_i}$.
2: Sample an isogeny $\psi_1' : E \to E_1'$ uniformly among the (possibly non-cyclic) isogenies from E of degree d.
3: Write $\psi_1 : E \to E_1$ for the cyclic component of ψ_1'.
4: Write $\deg \psi_1 = 2^n d'$ with d' odd.
5: Sample an isogeny $\psi_2 : E_1 \to E_2$ uniformly among the cyclic isogenies from E_1 of degree $2^{e_{\mathsf{rsp}}-n} - d'$.
6: **return** $h = (\psi_1, \psi_2)$.

Algorithm 6: $\mathsf{PushHint}(E, h, \sigma)$

Input: $E \in \mathsf{Supersingular}_p$,
$h = (\psi_1, \psi_2) \in \mathsf{HintSet}^{\mathsf{unif}}_E$ with $\psi_1 : E \to E_1$ and $\psi_2 : E_1 \to E_2$,
cyclic 2^k-isogeny $\sigma : E \to E'$ in efficient representation.
Output: $h' \in \mathsf{HintSet}^{\mathsf{unif}}_{E'}$

1: Write ψ for the composition $\psi = \psi_2 \circ \psi_1$.
2: Write $\deg(\psi) = 2^n d_{\mathsf{odd}}$ with d_{odd} odd.
3: Compute the pushforward $\psi'_{\mathsf{odd}} : E' \to E_1'$ of ψ_{odd} by τ.
4: Sample $\psi'_{2^n} : E_1' \to E_2'$ as a random cyclic 2^n-isogeny from E_1'.
5: Let $\psi' = \psi'_{2^n} \circ \psi'_{\mathsf{odd}}$.
6: Write ψ' for the composition $\psi' = \psi_2' \circ \psi_1'$ where $\deg \psi_i' = \deg \psi_i$ for $i \in \{1, 2\}$.
7: **return** $h' = (\psi_1', \psi_2')$.

Lemma 5.1. *Let $E \in \mathsf{Supersingular}_p$, let $k \in \mathbb{N}$ and let $\sigma : E \to E'$ be a cyclic 2^k-isogeny in efficient representation. If $h \leftarrow \mathcal{H}^{\mathsf{unif}}_E$, then $\mathsf{PushHint}(E, h, \sigma)$ is distributed according to $\mathcal{H}^{\mathsf{unif}}_{E'}$.*

Proof. Let $h = (\psi_1, \psi_2) \leftarrow \mathcal{H}^{\mathsf{unif}}_E$ and $h' = (\psi_1', \psi_2') \leftarrow \mathsf{PushHint}(E, h, \sigma)$. Write ψ for the composition $\psi_2 \circ \psi_1$, and similarly $\psi' = \psi_2' \circ \psi_1'$.

We start by noting that Lines 2 and 3 of Algorithm 5 (i.e., sampling of a possibly non-cyclic isogeny and extracting its cyclic component) affects the distribution of the degrees but not the distribution of the isogenies themselves, conditional on the degree. In other words, the isogeny ψ_1 is uniformly distributed among the cyclic isogenies of degree $\deg \psi_1$ from E. Since the $\mathsf{PushHint}$ algorithm preserves degrees, it is sufficient to show that ψ' is uniformly distributed among the cyclic isogenies of degree $\deg \psi'$ from E'.

Then, we first consider the case where the degree $d := \deg(\psi)$ is odd. Then, the isogeny ψ' is precisely the pushforward of ψ by σ; similarly, the isogeny ψ is the pullback of ψ' by σ (or, alternatively, the pushforward of ψ' by $\hat{\sigma}$). Thus, the isogeny σ induces a bijection between the isogenies from E of degree d and the isogenies from E' of degree d for any odd d. Since the supersingular graph is regular, the number of outgoing d-isogenies is the same for any curve in the graph. Hence, the distribution of ψ' is uniform among the cyclic isogenies from E' of degree d; that is, ψ' is perfectly distributed as the composition of the hints produced by $\mathcal{H}^{\mathsf{unif}}_{E'}$.

Lastly, we consider the case where $d = 2^n d_{\mathsf{odd}}$ with d_{odd} odd and $n > 0$. In this case, the isogeny ψ' is the composition of a d_{odd}-isogeny and a random cyclic 2^n-isogeny. The d_{odd} component is uniformly distributed among the d_{odd}-isogenies originating from E', by the previous argument; the 2^n-component is also uniformly distributed, by construction. Hence, the isogeny ψ' is perfectly distributed as the composition of the hints produced by $\mathcal{H}^{\mathsf{unif}}_{E'}$. □

By relying on the pushing algorithm (Algorithm 6), we can use a hint from $\mathcal{H}^{\mathsf{unif}}$ to generate something that looks like a hint from $\mathcal{H}^{\mathsf{sim}}$. Formally, we consider the problem of distinguishing between the two distributions in Experiment 4. For $q = \mathsf{poly}(\lambda)$, we refer to this as the q-hint distinguishing problem (q-$\mathsf{hint\text{-}dist}$). More formally, we have

Problem 3 (q-$\mathsf{hint\text{-}dist}$)**.** *Let $(E, h_1, \ldots, h_q)$ be sampled with probability 1/2 from $\mathsf{RealHints}(1^\lambda, q)$ and with probability 1/2 from $\mathsf{PushedHints}(1^\lambda, q)$, where $\mathsf{RealHints}(1^\lambda, q)$ and $\mathsf{PushedHints}(1^\lambda, q)$ are defined in Experiment 4. Given $(E, h_1, \ldots, h_q)$, distinguish between the two distributions.*

Experiment 4: Hint distinguishing

$\mathsf{RealHints}(1^\lambda, q)$:

1: $(E, I) \leftarrow \mathsf{Gen}_{R_{\mathsf{SQI}}}(1^\lambda)$
2: **for** $i = 1$ **to** q **do**
3: $\quad h_i \leftarrow \mathcal{H}_E^{\mathsf{sim}}$
4: **return** $(E, h_1, \ldots, h_q)$

$\mathsf{PushedHints}(1^\lambda, q)$:

1: $(E, I) \leftarrow \mathsf{Gen}_{R_{\mathsf{SQI}}}(1^\lambda)$
2: **for** $i = 1$ **to** q **do**
3: $\quad (\psi_1, \psi_2) \leftarrow \mathcal{H}_E^{\mathsf{unif}}$
4: $\quad (s, \varphi) \leftarrow D_{\mathsf{Chall}}(E)$ // see Experiment 3
5: $\quad (\psi_1', \psi_2') \leftarrow \mathsf{PushHint}(E, (\psi_1, \psi_2), \varphi)$
6: $\quad h_i := (s, \psi_1', \psi_2')$
7: **return** $(E, h_1, \ldots, h_q)$

For an algorithm $\mathcal{A}$, we let

$$\mathsf{Adv}^{\mathsf{hint\text{-}dist}}(\mathcal{A}, q) := \mathsf{Adv}^{\mathsf{dist}}\left[\mathsf{RealHints}(1^\lambda, q), \mathsf{PushedHints}(1^\lambda, q)\right](\mathcal{A}).$$

Remark 5.1. We expect the problem of distinguishing between the hints in $\mathsf{RealHints}(1^\lambda, q)$ and $\mathsf{PushedHints}(1^\lambda, q)$ to be computationally hard, following a similar argument on the heuristical equivalence of two oracles in SQIsign2D-West [BDD+24, Sec. 5.2]. The two hints distributions provide a representation of three isogenies $\varphi_1 : E \to E_1$, $\varphi_2 : E_1 \to E_2$, and $\varphi_3 : E_2 \to E_3$. The first isogeny φ_1 is sampled according to the same distribution in both games, thus the curves E, E_1 and the isogeny φ_1 cannot provide any distinguishing information. The same is true for the third isogeny φ_3: fixed a starting curve E_2, the isogeny φ_3 is sampled according to the same distribution in both games. Thus, we focus on the second isogeny: in $\mathsf{RealHints}(1^\lambda, q)$, the isogeny φ_2 is uniformly distributed among the isogenies between E_1 and E_2; by rejection sampling, the same is true in $\mathsf{PushedHints}(1^\lambda, q)$. Hence, the only distinguishing factor is the distribution of the curve E_2: in $\mathsf{RealHints}(1^\lambda, q)$ is distributed according to the stationary distribution, while in $\mathsf{PushedHints}(1^\lambda, q)$ is the codomain of a random isogeny φ_2 from E_1 of bounded degree. If the isogeny φ_2 were sufficiently long, the statistical distance between the curves E_1 produced by the two games would become negligible; in our case, the bound on the degree of φ_2 prevents E_2 from being close to stationary, but it is sufficiently large to make the distinguishing problem computationally hard.

5.2 The reduction from hint-OneEnd

Problem 4 (q-**hint-OneEnd**$_p$)**.** *Given a curve E sampled from the stationary distribution S on* $\mathsf{Supersingular}_p$ *and q hints $h_1, \ldots, h_q \leftarrow \mathcal{H}_E^{\mathsf{unif}}$, find an endomorphism in* $\mathrm{End}(E) \setminus \mathbb{Z}$ *in efficient representation.*

For an algorithm $\mathcal{A}$, we write $\mathsf{Adv}^{\mathsf{hint\text{-}OneEnd}_p}(\mathcal{A}, q)$ for the probability that it solves q-$\mathsf{hint\text{-}OneEnd}_p$. We note that the distribution of $j(E_{\mathsf{pk}})$ is $D_{E_0}^{\mathsf{mix}}$ and thus close to the stationary distribution. We obtain a reduction from q-$\mathsf{hint\text{-}OneEnd}_p$ by replacing the hint distribution in Theorem 2 with $\mathcal{H}^{\mathsf{unif}}$.

Theorem 3. *For any PPT algorithm $\mathcal{A}$ against the* EUF-CMA *of* $\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]$*, there exists expected polynomial time algorithms $\mathcal{B}$ and $\mathcal{D}$ with*

$$\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]}(\mathcal{A}) \le (q+1) \cdot \left(2 \cdot \mathsf{Adv}^{\mathsf{hint\text{-}OneEnd}_p}(\mathcal{B}, s) + 2 \cdot \mathsf{Adv}^{\mathsf{hint\text{-}dist}}(\mathcal{D}, s) + 2^{-e_{\mathsf{chl}}}\right) + \frac{2qs + s^2 + 2s + q + 1}{\sqrt{p}},$$

where q and s are upper bounds on the number of queries that $\mathcal{A}$ makes to RO *and* OSign*, respectively.*

Proof. By Theorem 2, there exists an expected polynomial time algorithm $\mathcal{E}$ such that

$$\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]}(\mathcal{A}) \le (q+1) \cdot \left(2 \cdot \mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}}, \mathcal{H}^{\mathsf{sim}})}(\mathcal{E}, s) + 2^{-e_{\mathsf{chl}}}\right) + \frac{(2q + s + 2)s}{\sqrt{p}}. \tag{6}$$

We construct $\mathcal{B}$ from $\mathcal{E}$ by simulating the hints of $\mathcal{H}^{\mathsf{sim}}$ with the hints from $\mathcal{H}^{\mathsf{unif}}$. $\mathcal{B}$ gets as input a curve E_{pk} sampled from $\mathsf{Gen}_{R_{\mathsf{OneEnd}}}(1^\lambda)$ and hints $h_1, \ldots, h_s \leftarrow \mathcal{H}^{\mathsf{unif}}_{E_{\mathsf{pk}}}$. It then uses the $\mathcal{H}^{\mathsf{unif}}$-hints to compute hints $h_1', \ldots, h_s' \in \mathsf{HintSet}^{\mathsf{sim}}_{E_{\mathsf{pk}}}$ as in $\mathsf{PushedHints}(1^\lambda, s)$ of Experiment 4. Finally, it runs $\alpha \leftarrow \mathcal{E}(E_{\mathsf{pk}}, h_1', \ldots, h_s')$ and outputs α.

Any difference in success probability between $\mathcal{E}$ and $\mathcal{B}$ can be used to construct an adversary $\mathcal{D}$ against s-hint-dist. On input E and hints $h_1, \ldots, h_s \in \mathsf{HintSet}_E^{\mathsf{sim}}$, $\mathcal{D}$ runs $\mathcal{E}(E, h_1, \ldots, h_s)$. It outputs 1 if $\mathcal{E}$ outputs a witness α s.t. $(E, \alpha) \in R_{\mathsf{OneEnd}}$, else it outputs 0. By construction,

$$\mathsf{Adv}^{\mathsf{hint\text{-}dist}}(\mathcal{D}, s) = \left| \mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}}, \mathcal{H}^{\mathsf{sim}})}(\mathcal{E}, s) - \mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}}, \mathcal{H}^{\mathsf{unif}})}(\mathcal{B}, s) \right|,$$

so in particular

$$\mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}}, \mathcal{H}^{\mathsf{sim}})}(\mathcal{E}, s) \leq \mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}}, \mathcal{H}^{\mathsf{unif}})}(\mathcal{B}, s) + \mathsf{Adv}^{\mathsf{hint\text{-}dist}}(\mathcal{D}, s). \tag{7}$$

In the q-hint-OneEnd$_p$ problem, the curve is sampled from the stationary distribution S on $\mathsf{Supersingular}_p$. By Proposition 2.3 and Proposition 2.4,

$$\begin{aligned} \mathsf{Adv}^{\mathsf{hint\text{-}rel}}_{(R_{\mathsf{OneEnd}}, \mathcal{H}^{\mathsf{unif}})}(\mathcal{B}, s) &\leq \mathsf{Adv}^{\mathsf{hint\text{-}OneEnd}_p}(\mathcal{B}, s) + \Delta(E_{\mathsf{pk}}, S) \\ &\leq \mathsf{Adv}^{\mathsf{hint\text{-}OneEnd}_p}(\mathcal{B}, s) + \frac{1}{2\sqrt{p}}. \end{aligned} \tag{8}$$

We conclude by plugging (7) and (8) into (6).

□

Furthermore, we show that q-hint-OneEnd$_p$ has a worst-case to average-case reduction.

Lemma 5.2. *Let $\mathcal{A}$ be an algorithm for q-hint-OneEnd$_p$ with advantage ε and outputs of degree at most d. Then we can construct an algorithm $\mathcal{A}'$ with the following properties.*

1. *For any curve $E \in \mathsf{Supersingular}_p$, it solves q-hint-OneEnd$_p$ for E with probability in $[\varepsilon - \frac{\log p}{p}, \varepsilon + \frac{\log p}{p}]$.*
2. *It runs $\mathcal{A}$ once. The rest of the algorithm runs in polynomial time in $\log p$, $\log d$ and q.*
3. *Its output has degree at most $2^{6\lceil \log p \rceil} \cdot d$.*

Proof. $\mathcal{A}'$ gets as input $E \in \mathsf{Supersingular}_p$ and $h_1, \ldots, h_q \leftarrow \mathcal{H}_E^{\mathsf{unif}}$, and does the following.

1. Sample $\sigma' : E \to E_A$ by a random non-backtracking 2-isogeny walk of length $3\lceil \log p \rceil$.
2. $A' \leftarrow \mathsf{MontgomeryRandomize}(A)$ and $E' \coloneqq E_{A'}$.
3. Compute an isomorphism $\iota : E_A \to E'$ and let $\sigma \coloneqq \iota \circ \sigma' : E \to E'$.
4. $h'_i \leftarrow \mathsf{PushHint}(E, h_i, \sigma)$ for $i = 1, \ldots, q$.
5. Run $\mathcal{A}$ on input E' and $h'_1, \ldots, h'_q$.
6. When $\mathcal{A}$ outputs α, abort if $\alpha \notin \mathsf{EndRing}(E') \setminus \mathbb{Z}$.
7. Output $\widehat{\sigma} \circ \alpha \circ \sigma$.

If $\widehat{\varphi} \circ \alpha \circ \varphi = [m]$, then pre-composing with $\widehat{\varphi}$ and post-composing with φ, we get that $[(\deg \varphi)^2] \circ \alpha = [\deg(\varphi) m]$, so that $\alpha = [m/(\deg(\varphi))]$. Hence, if $\alpha \in \mathrm{End}(E') \setminus \mathbb{Z}$, then $\widehat{\sigma} \circ \alpha \circ \sigma \in \mathrm{End}(E) \setminus \mathbb{Z}$.

Let D be the distribution of E' and D_j the distribution of $j(E')$. By Proposition 2.4,

$$\Delta(D, S) = \Delta(D_j, S_j) \leq \frac{3\lceil \log p \rceil + 1}{4p} \leq \frac{\log p}{p},$$

using $\log p \geq 4$ for the last inequality.

Any difference in success probability between $\mathcal{A}$ and $\mathcal{A}'$ can be used to construct a non-efficient distinguisher $\mathcal{D}$ between D and S. It will not be efficient, as it needs to sample hints from $\mathcal{H}^{\mathsf{unif}}$ itself. On input a curve E, $\mathcal{D}$ first computes a basis for $\mathrm{End}(E)$. It then uses the basis to sample q hints $h_1, \ldots, h_q \leftarrow \mathcal{H}_E^{\mathsf{unif}}$. Finally, it runs $\mathcal{A}$ on input E and $h_1, \ldots, h_q$, and outputs 1 if and only if $\mathcal{A}$ succeeds. By Proposition 2.3, $\mathsf{Adv}^{\mathsf{dist}}[D, S](\mathcal{D}) = |\varepsilon'(E) - \varepsilon| \leq \Delta(D, S)$. □

5.3 The classical reduction from hint-EndRing

Problem 5 (q-hint-EndRing$_p$). *Given a curve E sampled from the stationary distribution S on $\mathsf{Supersingular}_p$ and q hints $h_1, \ldots, h_q \leftarrow \mathcal{H}_E^{\mathsf{unif}}$, find four endomorphisms in efficient representation that form a basis of $\mathrm{End}(E)$ as a lattice.*

For an algorithm $\mathcal{A}$, we let $\mathsf{Adv}^{\mathsf{hint\text{-}EndRing}_p}(\mathcal{A}, q)$ denote the probability that $\mathcal{A}$ succeeds in solving q-$\mathsf{hint\text{-}EndRing}_p$.

Without hints, [PW24] proved that an oracle for OneEnd_p can be used to construct an efficient algorithm for $\mathsf{EndRing}_p$. We recap their reduction in Section 6.1. Importantly, we show in Corollary 6.1 that if the OneEnd_p oracle OEnd has outputs of degree at most d, the algorithm for $\mathsf{EndRing}_p$ runs in time $\mathsf{poly}(\log p, \log d)$ times the number of calls to OEnd. In expectation, the number of times it calls OEnd is

$$t_{\mathsf{OneEnd}}(\log p, \log d) < 2^{94} \cdot (\log(p) + \log(d)/30)^{13}.$$

We will show that with the pushable hint distribution $\mathcal{H}^{\mathsf{unif}}$, we can reduce $\mathsf{hint\text{-}EndRing}_p$ to $\mathsf{hint\text{-}OneEnd}_p$ in the same way that [PW24] reduces $\mathsf{EndRing}_p$ to OneEnd_p. We assume throughout that $q, d = \mathsf{poly}(\log p)$. In the intermediate steps of our reduction, we consider several kinds of oracles.

- For each $E \in \mathsf{Supersingular}_p$, OHint_E is an oracle that when queried outputs a fresh hint sampled from $\mathcal{H}^{\mathsf{unif}}_E$.
- For each $E \in \mathsf{Supersingular}_p$, $\mathsf{OEnd1}_{E,d}$ is an oracle which on input a 2^k-isogeny $\sigma : E \to E'$ in efficient representation for some $k \in \mathbb{N}$ and $E' \in \mathsf{Supersingular}_p$, outputs a non-scalar endomorphism of E' of degree at most d.
- $\mathsf{OEnd2}_{q,d,\varepsilon}$ is an oracle which, on input a curve $E \in \mathsf{Supersingular}_p$ and q hints $h_1, \ldots, h_q \leftarrow \mathcal{H}^{\mathsf{unif}}_E$, outputs a non-scalar endomorphism of E of degree at most d, with probability in $[\varepsilon - \frac{\log(p)}{p}, \varepsilon + \frac{\log(p)}{p}]$.

The oracles allow a more modular proof, where the oracles are the parts we have not implemented yet. In the runtime analysis, we count each call to an oracle as a single step.

Lemma 5.3. *There is an algorithm $\mathcal{A}$, which on input a curve $E \in \mathsf{Supersingular}_p$ and given query access to the oracle $\mathsf{OEnd1}_{E,d}$, computes $\mathrm{End}(E)$. Its runtime is $\mathsf{poly}(\log p)$ times the number of calls it makes to $\mathsf{OEnd1}_{E,d}$. In expectation, the number of times it calls $\mathsf{OEnd1}_{E,d}$ is $t_{\mathsf{OneEnd}}(\log p, \log d)$.*

Proof. We follow the reduction from $\mathsf{EndRing}_p$ to OneEnd_p in [PW24]. We only need to implement their "rich oracle" $\mathsf{RICH}^{\mathscr{O}}$ with parameter $k = \mathsf{poly}(\log p)$ [PW24, Algorithm 1]. We implement it as follows.

1. Sample $\sigma : E \to E'$ by a random non-backtracking 2-isogeny walk of length k.
2. $\alpha \leftarrow \mathsf{OEnd1}_{E,d}(\sigma)$.
3. (In the second stage of the reduction, reduce α.)
4. Return $\widehat{\sigma} \circ \alpha \circ \sigma$.

The expected number of calls to $\mathsf{OEnd1}_{E,d}$ is exactly the expected number of calls to the OneEnd_p oracle in the reduction in [PW24]. The runtime follows from Corollary 6.1. □

We can implement $\mathsf{OEnd1}_{E,d}$ using OHint_E and $\mathsf{OEnd2}_{q,d,\varepsilon}$ if ε is sufficiently large.

Lemma 5.4. *Assume $\varepsilon \geq 2\log(p)/p$. There is an algorithm $\mathcal{B}$, which on input a 2^k-isogeny $\sigma : E \to E'$, and given query-access to the oracles $\mathsf{OEnd2}_{q,d,\varepsilon}$ and OHint_E, has the following properties.*

1. *It outputs a non-scalar endomorphism of E' of degree at most d.*
2. *Its runtime is $\mathsf{poly}(\log p)$ times the number of times it calls $\mathsf{OEnd2}_{q,d,\varepsilon}$.*
3. *In expectation, it calls $\mathsf{OEnd2}_{q,d,\varepsilon}$ at most $2/\varepsilon$ times.*
4. *For every call to $\mathsf{OEnd2}_{q,d,\varepsilon}$, it makes q calls to OHint_E.*

Proof. On input $E, E' \in \mathsf{Supersingular}_p$ and a 2^k-isogeny $\sigma : E \to E'$, $\mathcal{B}$ proceeds as follows.

1. Sample $h_i \leftarrow \mathsf{OHint}_E$ for $i = 1, \ldots, q$.
2. $h'_i \leftarrow \mathsf{PushHint}(E, h_i, \sigma)$ for $i = 1, \ldots, q$.
3. $\alpha \leftarrow \mathsf{OEnd2}_{q,d,\varepsilon}(E', h'_1, \ldots, h'_q)$.
4. If $\alpha \notin \mathrm{End}(E) \setminus \mathbb{Z}$, go back to step 1. Else, output α.

Each iteration of the loop is an independent trial that succeeds with probability at least $\varepsilon - \log(p)/p \geq \varepsilon - \varepsilon/2 = \varepsilon/2$. Hence, the expected number of iterations is at most $2/\varepsilon$. □

Combing the two previous results, we obtain the following.

Corollary 5.1. *Assume $\varepsilon \geq 2\log(p)/p$. There exists an algorithm $\mathcal{D}$, which on input a curve $E \in \mathsf{Supersingular}_p$, and given query access to OHint_E and $\mathsf{OEnd2}_{q,d,\varepsilon}$, computes $\mathrm{End}(E)$. It has the following properties.*

1. *Its runtime is $\mathsf{poly}(\log p)$ times the number of times it calls $\mathsf{OEnd2}_{q,d,\varepsilon}$.*
2. *In expectation, it calls $\mathsf{OEnd2}_{q,d,\varepsilon}$ at most $2t_{\mathsf{OneEnd}}(\log p, \log d)/\varepsilon$ times.*
3. *For every call to $\mathsf{OEnd2}_{q,d,\varepsilon}$, it makes q calls to OHint_E.*

When we have an algorithm for q-$\mathsf{hint\text{-}OneEnd}_p$ with non-negligible advantage, we can convert it to an algorithm for s-$\mathsf{hint\text{-}EndRing}_p$, without any oracles.

Lemma 5.5. *Let $\mathcal{A}$ be an expected polynomial time algorithm for q-$\mathsf{hint\text{-}OneEnd}_p$ with advantage $\varepsilon \geq 2\log(p)/p$ and with outputs of degree at most d. Let*

$$s := 8q \cdot t_{\mathsf{OneEnd}}(\log p, \log d + 6\lceil \log p \rceil)/\varepsilon\,.$$

Then there exists an algorithm $\mathcal{B}$ for s-$\mathsf{hint\text{-}EndRing}_p$, running in expected time $\mathsf{poly}(\log p)\cdot s$, with advantage at least $1/2$.

Proof. Let $d' := 2^{6\lceil \log p \rceil} \cdot d$. We let $\mathcal{E}$ be the algorithm from Corollary 5.1, which on input $E \in \mathsf{Supersingular}_p$ and with query access to the oracles OHint_E and $\mathsf{OEnd2}_{q,d',\varepsilon}$, computes $\mathrm{End}(E)$. In expectation, it makes at most $4t_{\mathsf{OneEnd}}(\log p, \log d')/\varepsilon$ calls to $\mathsf{OEnd2}_{q,d',\varepsilon}$.

Define $\mathcal{D}$ as the algorithm that runs $\mathcal{E}$, but times out after s/q calls to $\mathsf{OEnd2}_{q,d',\varepsilon}$. It runs in time at most $s \cdot \mathsf{poly}(\log p)$, by Corollary 5.1. By Corollary 3.1, $\mathcal{D}$ succeeds with probability at least $1/2$.

Next, we implement $\mathsf{OEnd2}_{q,d',\varepsilon}$ using $\mathcal{A}$. Let $\mathcal{A}'$ be the algorithm we obtain from the worst-case to average-case reduction in Lemma 5.2 with $\mathcal{A}$. It runs in expected polynomial time in $\log p$, with advantage in $[\varepsilon - \frac{\log p}{p}, \varepsilon + \frac{\log p}{p}]$. Its output has degree at most d'.

Finally, we construct the algorithm $\mathcal{B}$ for s-$\mathsf{hint\text{-}EndRing}_p$. It runs $\mathcal{D}$ with $\mathcal{A}'$ as $\mathsf{OEnd2}_{q,d',\varepsilon}$. The hint oracle is implemented so that it answers the ith query with the ith hint h_i. $\mathcal{B}$ runs in expected time $s \cdot \mathsf{poly}(\log p)$ and has advantage at least $1/2$. □

We conclude that if q-$\mathsf{hint\text{-}EndRing}_p$ and q-$\mathsf{hint\text{-}dist}$ are computationally hard problems, then $\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]$ is EUF-CMA-secure in the ROM. The reduction is not tight, as the runtime loss from the reduction in [PW24] is polynomial, but concretely huge.

Theorem 4. *For any PPT algorithm $\mathcal{A}$ against the EUF-CMA of $\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]$, there exists expected polynomial time algorithms $\mathcal{B}$ and $\mathcal{D}$ with*

$$\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]}(\mathcal{A}) \leq (q+1)\cdot\left(2\cdot\mathsf{Adv}^{\mathsf{hint\text{-}OneEnd}_p}(\mathcal{B}, s) + 2\cdot\mathsf{Adv}^{\mathsf{hint\text{-}dist}}(\mathcal{D}, s) + 2^{-e_{\mathsf{chl}}}\right) + \frac{2qs + s^2 + 2s + q + 1}{\sqrt{p}},$$

where q and s are upper bounds on the number of queries that $\mathcal{A}$ makes to RO and OSign, respectively. Whenever $\mathcal{B}$ has advantage $\varepsilon_{\mathcal{B}} \geq 2\log(p)/p$, there is an algorithm $\mathcal{E}$ for t-$\mathsf{hint\text{-}EndRing}_p$ with

$$t = 2^{103}\lceil \log p \rceil^{13} \cdot s/\varepsilon_{\mathcal{B}} \quad \text{and} \quad \mathsf{Adv}^{\mathsf{hint\text{-}EndRing}_p}(\mathcal{E}, t) \geq 1/2,$$

running in expected time $\mathsf{poly}(\log p)/\varepsilon_{\mathcal{B}}$.

Proof. The first part is the same as Theorem 3. By the definition of the soundness relation R_{OneEnd}, when $\mathcal{B}$ succeeds, it outputs an endomorphism of degree at most p^4. Assume it has advantage $\varepsilon_{\mathcal{B}} \geq 2\log(p)/p$. By Lemma 5.5, there exists an algorithm $\mathcal{E}$ for t-$\mathsf{hint\text{-}EndRing}_p$ with

$$t = 8 \cdot s \cdot t_{\mathsf{OneEnd}}(\log p, 10\lceil \log p \rceil)/\varepsilon_{\mathcal{B}}$$

hints, having expected time $\mathsf{poly}(\log p) \cdot t = \mathsf{poly}(\log p)/\varepsilon_{\mathcal{B}}$ and advantage at least $1/2$. By Lemma 5.3

$$\begin{aligned} t_{\mathsf{OneEnd}}(\log p, 10\lceil \log p \rceil) &= 2^{94} \cdot (\log(p) + (10\lceil \log p \rceil)/30)^{13} \\ &\leq 2^{100}\lceil \log p \rceil^{13}. \end{aligned}$$

□

5.4 The quantum reduction from hint-EndRing

We can obtain a much tighter *quantum* reduction from $\mathsf{hint\text{-}EndRing}_p$ to $\mathsf{hint\text{-}OneEnd}_p$. A serious source of complexity in the reduction of [PW24] arises from the possibility that a OneEnd_p oracle might produce endomorphisms with hard-to-factor discriminants. Without this obstacle, several steps of the reduction become redundant. We prove the following theorem in Section 6.2.

Theorem 5. *Let* OEnd *be an oracle for the* OneEnd_p *problem with outputs of degree at most* d*, and* $\mathsf{OFactor}$ *an oracle for integer factorization. Then there exists a three-stage algorithm for* $\mathsf{EndRing}_p$ *which runs in expected polynomial time in* $\log p$ *and* $\log d$*. The first and third stage each call* OEnd *in expectation 12 times. The second stage makes a single call to* $\mathsf{OFactor}$ *and runs in polynomial time.*

With Shor's algorithm [Sho94], the second stage can be implemented by a quantum reduction. On the other hand, with a classical algorithm for $\mathsf{hint\text{-}OneEnd}_p$, the first and third stage can remain classical in the reduction to $\mathsf{hint\text{-}EndRing}_p$. We can therefore apply our classical reduction separately for the first and third stage. We summarize the quantum reduction in the following theorem.

Theorem 6. *For any PPT algorithm* $\mathcal{A}$ *against the* $\mathsf{EUF\text{-}CMA}$ *of* $\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]$*, there exists expected polynomial time algorithms* $\mathcal{B}$ *and* $\mathcal{D}$ *with*

$$\mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\mathsf{SIG}[\Sigma_{\mathsf{SQI}}]}(\mathcal{A}) \leq (q+1)\cdot\left(2\cdot\mathsf{Adv}^{\mathsf{hint\text{-}OneEnd}_p}(\mathcal{B},s)+2\cdot\mathsf{Adv}^{\mathsf{hint\text{-}dist}}(\mathcal{D},s)+2^{-e_{\mathsf{chl}}}\right) + \frac{2qs+s^2+2s+q+1}{\sqrt{p}},$$

where q *and* s *are upper bounds on the number of queries that* $\mathcal{A}$ *makes to* RO *and* OSign*, respectively. Whenever* $\mathcal{B}$ *has advantage* $\varepsilon_{\mathcal{B}} \geq 2\log(p)/p$ *there exists a quantum algorithm* $\mathcal{E}$ *for* t-$\mathsf{hint\text{-}EndRing}_p$ *with*

$$t = 192s/\varepsilon_{\mathcal{B}} \quad \text{and} \quad \mathsf{Adv}^{\mathsf{hint\text{-}EndRing}_p}(\mathcal{E},t) \geq 1/4,$$

running in expected time $\log(p)/\varepsilon_{\mathcal{B}}$.

Proof. The first part is the same as Theorem 3. Assume $\mathcal{B}$ has advantage $\varepsilon_{\mathcal{B}} \geq 2\log(p)/p$. Following the proof of Lemma 5.5 with $t_{\mathsf{OneEnd}}(\log p, \log d') = 12$, we obtain a classical algorithm $\mathcal{E}_1$ for the first stage. It runs $\mathcal{B}$ at most $8\cdot 12/\varepsilon_{\mathcal{B}} = 96/\varepsilon_{\mathcal{B}}$ times, and succeeds with probability at least $1/2$. In the same way, we obtain a classical algorithm $\mathcal{E}_3$ for the third stage, with the exact same properties. Let $\mathcal{E}_2$ the quantum algorithm that implements the second stage.

We let $\mathcal{E}$ be the three stage algorithm $(\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3)$. If the first stage succeeds, the second stage always succeeds, and the third stage succeeds with probability at least $1/2$. Hence, $\mathcal{E}$ succeeds with probability at least $1/4$. □

Under the assumption that t-$\mathsf{hint\text{-}EndRing}_p$ is a hard problem for quantum algorithms, this is a meaningful result. By using the classical forger, the quantum reduction can efficiently recover a basis for $\mathrm{End}(E)$.

5.5 Expected hardness of the q-$\mathsf{hint\text{-}EndRing}_p$ problem

In Theorem 4 and Theorem 6, we proved that SQIsign is $\mathsf{EUF\text{-}CMA}$-secure assuming the hardness of q-$\mathsf{hint\text{-}EndRing}_p$ (Problem 5) and q-$\mathsf{hint\text{-}dist}$ (Problem 3). We covered the hardness of q-$\mathsf{hint\text{-}dist}$ in Remark 5.1. Here, we focus on why we expect the q-$\mathsf{hint\text{-}EndRing}_p$ problem to be hard, or —in other words— why we do not expect the hints to make $\mathsf{EndRing}_p$ easier.

Consider what the $\mathcal{H}^{\mathsf{unif}}_E$ hint distribution does: it samples a degree d according to a weighted distribution, and then it samples a random isogeny (if we consider the composition $\psi_2 \circ \psi_1$) of degree depending on d. In Section 5.6, we show that it is possible to sample from a distribution that is negligibly close to the distribution from which the value d is sampled, *in polynomial time*. This means that the only non-trivial information that a hint sampled from $\mathcal{H}^{\mathsf{unif}}_E$ provides to the simulator is the isogeny $\psi_2 \circ \psi_1$ itself.

To sample an isogeny according to $\mathcal{H}^{\mathsf{unif}}_E$, the simulator can do the following:

– Factorize[11] the degree of $\psi_2 \circ \psi_1$ as $\deg \psi_2 \circ \psi_1 = \prod_i^t p_i^{e_i}$, and assume that p_t is the largest prime diving $\deg \psi_2 \circ \psi_1$. The complexity of the factorization is polynomial in p_t.

[11] It is possible to sample the degree d together with its factorization [Bac88], but it is still necessary to factor the degree of ψ_2.

– For every prime p_i:
  - Sample a random point K of $E[p_i]$, which is defined over an extension field of order $O(p_i)$. The complexity is polynomial in p_i.
  - Compute an isogeny with kernel $\langle K \rangle$ with VéluSqrt formulas [BDLS20]. The complexity is similarly polynomial in p_i.
  - This yields an isogeny of degree p_i. Repeating the process e_i times and concatenating the outputs, produces an isogeny of degree $p_i^{e_i}$.
– Finally, iterating over all t possible p_i and concatenating the outputs gives an uniformly random isogeny of the desired degree.

The complexity of the sampling procedure is polynomial in p_t. Hence, if the degree of $\psi_2 \circ \psi_1$ is sufficiently smooth (i.e., the smoothness bound p_t is in $O(\mathsf{poly}(\lambda))$), the hint does not provide any additional information. Unfortunately, that happens with negligible probability.

In conclusion, the only information the hints provide to the adversary is the efficient representation of some isogenies of non-smooth degree. Informally, we do not expect these to provide any help in solving the endomorphism problem: non-smooth degree isogenies do not provide any additional information compared to smooth-degree isogenies.

5.6 Sampling a random degree

For any elliptic curve E and integer N, let $A_N(E)$ be the set of isogenies from E and of degree at most N (up to post-composition with an isomorphism). Let $B_N(E) \subset A_N(E)$ be the subset of isogenies with cyclic kernel. We have

$$A_N(E) = \bigsqcup_{n<N} [n] \circ B_{\lfloor N/n \rfloor}(E).$$

Recall Dedekind's totient is the multiplicative function defined by $\psi(\ell^t) = (\ell+1)\ell^{t-1}$ for all primes ℓ. For any $n \leq N$, the number of isogenies of degree n in $B_N(E)$ is equal to $\psi(n)$. In particular,

$$\#B_N(E) = \sum_{n \leq N} \psi(n).$$

The asymptotic behavoir of this sum is proven in [OEI25, Apo76, HÏ6].

Lemma 5.6. *$\#B_N(E) = \gamma \cdot N^2 + O(N \ln(N))$, where $\gamma = 15/2\pi^2$.*

Lemma 5.7. *There is an algorithm which, on input N, samples a random integer d with the same distribution as $\deg(\varphi)$, where φ is uniform in $B_N(E)$*

Proof. The probability distribution of $\deg(\varphi)$ is $f(n) = \psi(n)/\#B_N(E)$. We have

$$\psi(n) = n \prod_{\substack{\ell \mid n \\ \ell \text{ prime}}} \left(1 + \frac{1}{\ell}\right) = n \prod_{\substack{\ell \mid n \\ \ell \text{ prime}}} \left(1 - \frac{\mu(\ell)}{\ell}\right) = n \sum_{d \mid n} \frac{\mu(d)^2}{d} \leq n(\ln(n) + 2).$$

Let $g(n) = 1/N$ be the uniform distribution on $\{1, \ldots, N\}$. Let $M = \frac{N^2(\ln(N)+2)}{\#B_N(E)}$. We then have

$$\frac{f(n)}{Mg(n)} = \frac{\psi(n)}{\#B_N(E)} \cdot \frac{N\#B_N(E)}{N^2(\ln(N)+2)} = \frac{\psi(n)}{N(\ln(N)+2)} \leq 1.$$

We consider the following *rejection sampling* algorithm:

1. Sample $n \in \{1, \ldots, N\}$ together with its factorization uniformly at random using [Bac88].
2. Compute $\psi(n)$ thanks to the factorization of n, and let $p = \frac{\psi(n)}{N(\ln(N)+2)}$.
3. With probability p, return n. Otherwise, restart.

The algorithm repeats an expected M times, and from Lemma 5.6,

$$M = \frac{N^2(\ln(N)+2)}{\#B_N(E)} = \frac{N^2(\ln(N)+2)}{\gamma \cdot N^2 + O(N\log(N))} = O(\log N).$$

□

Proposition 5.1. *There is an algorithm which, on input N, samples a random integer d at statistical distance $O\left(\frac{(\log N)^2}{N}\right)$ from the distribution of $\deg(\varphi)$, where φ is uniform in $A_N(E)$.*

Proof. Since

$$A_N(E) = \bigsqcup_{n<N} [n] \circ B_{\lfloor N/n \rfloor}(E),$$

and we can sample efficiently in each $B_{\lfloor N/n \rfloor}(E)$ (Lemma 5.7), it only remains to show that we can sample the cyclic part n. The distribution of n for a uniformly random element in $A_N(E)$ is

$$f(n) = \frac{\#B_{\lfloor N/n \rfloor}(E)}{\#A_N(E)}.$$

Let $\alpha = \sum_{i=1}^{N} \frac{1}{i^2}$. Let us prove that the distribution $g(n) = \frac{1}{\alpha n^2}$ on $\{1, \dots, N\}$ is at the claimed statistical distance of f.

We have

$$\begin{aligned}\#A_N(E) &= \sum_{n \le N} \#B_{\lfloor N/n \rfloor}(E) = \sum_{n \le N} \left(\gamma \frac{N^2}{n^2} + O\left(\frac{N}{n} \log N\right)\right) \\ &= \alpha\gamma N^2 + O\left(N(\log N)^2\right)\end{aligned}$$

Now, to bound the statistical distance, we compute

$$\begin{aligned}\#A_N(E)\|f - g\|_1 &= \#A_N(E) \sum_{n \le N} |f(n) - g(n)| \\ &= \sum_{n \le N} \left|\#B_{\lfloor N/n \rfloor}(E) - \frac{1}{\alpha n^2}\#A_N(E)\right| \\ &= \sum_{n \le N} \left|\gamma \frac{N^2}{n^2} + O\left(\frac{N}{n} \log N\right) - \gamma \frac{N^2}{n^2} + O\left(\frac{N}{n^2}(\log N)^2\right)\right| \\ &= O(N(\log N)^2).\end{aligned}$$

Therefore, $\|f - g\|_1 = O\left(\frac{(\log N)^2}{N}\right)$, as claimed.

Finally, to sample from distribution g, one can sample an integer n for the zeta distribution with parameter 2 (e.g., with inverse transform sampling), and reject if $n > N$. □

6 Analyzing the EndRing to OneEnd reduction

This section collects our analysis on the $\mathsf{EndRing}_p$ to OneEnd_p reduction in [PW24]. In Section 6.1, we analyze the expected runtime of their reduction. Then in Section 6.2, we show how to make the reduction a lot tighter, given access to a factoring oracle.

6.1 A detailed analysis of the runtime loss

In [PW24], Page and Wesolowski proved that an efficient algorithm for OneEnd_p can be used to construct an efficient algorithm for $\mathsf{EndRing}_p$.

Lemma 6.1 ([PW24]). *Let $d : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ be a function. Let OEnd be an oracle for OneEnd_p, that when queried on a curve $E \in \mathsf{Supersingular}_p$ outputs an endomorphism $\alpha \in \mathrm{End}(E) \setminus \mathbb{Z}$ of degree at most d. Then there exists an algorithm $\mathcal{A}$ that, on input $E \in \mathsf{Supersingular}_p$ and given query access to the oracle OEnd, outputs a basis for $\mathrm{End}(E)$ in efficient representation. $\mathcal{A}$ runs in expected polynomial time in $\log p$ and $\log d$.*

By inspecting the proof in [PW24], we obtain that the runtime loss is polynomial in $\log p$, but concretely very large.

Corollary 6.1. *Let definitions be as in Lemma 6.1. The runtime of $\mathcal{A}$ is $\mathsf{poly}(\log p, \log d)$ times the number of calls it makes to $\mathcal{A}$. In expectation, the number of times it calls OEnd is*

$$t_{\mathsf{OneEnd}}(\log p, \log d) < 2^{94} \cdot (\log(p) + \log(d)/30)^{13}.$$

Proof. The EndRing algorithm $\mathcal{A}$ is described in [PW24, Algorithm 5]. We derive the bound t by inspecting the proof of [PW24, Theorem 7.2]. We do not include their proof in full, but rather give pointers to the relevant parts.

The OneEnd oracle OEnd is called in the two loops of [PW24, Algorithm 5]. In the first loop (lines 3 to 6), each iteration calls OEnd once. Their proof says that the loop terminates after any 3 consecutive iterations with probability at least $1/16$. Thus, the loop succeeds in expectation after at most $3 \cdot 16 = 48$ iterations.

The second loop is more complicated. Each iteration queries OEnd three times. The loop termination condition is that $[\mathrm{End}(E) : R] = 1$. Note that [PW24] starts by bounding the degree of the endomorphisms produced by their algorithm $\mathsf{Rich}_{k_1}^{\mathsf{OEnd}}(E)$ by $2^{2k_1 \log(d)}$. While correct, the tighter bound $2^{2k_1+\log(d)}$ is satisfied by construction. Using the same techniques as [PW24], but starting from the bound $2^{2k_1+\log(d)}$ instead of $2^{2k_1 \log(d)}$, leads to

$$[\mathrm{End}(E) : R] \leq 2^{3(k_1+\log(d)/2)+2}/p,$$

where

$$k_1 = \left\lceil \frac{\log(12 \cdot 9 \cdot (1+\sqrt{3}) \cdot \sqrt{p+13})}{\log(\frac{3}{2\sqrt{2}})} \right\rceil.$$

When $p \geq 3146$,

$$\begin{aligned}
k_1 &\leq 1 + \frac{\log(12 \cdot 9 \cdot (1+\sqrt{3}) \cdot \sqrt{(1+1/242)} \cdot \sqrt{p})}{\log(\frac{3}{2\sqrt{2}})} \\
&\leq 1 + \frac{\log(296) + \log(\sqrt{p})}{\log(\frac{3}{2\sqrt{2}})} \\
&= 1 + \frac{\log(12 \cdot 9 \cdot (1+\sqrt{3}) \cdot \sqrt{2})}{\log(\frac{3}{2\sqrt{2}})} + \frac{\log(p)}{2\log(\frac{3}{2\sqrt{2}})} \\
&\leq 98 + \frac{\log(p)}{2\log(\frac{3}{2\sqrt{2}})} \\
&\leq 15\log(p),
\end{aligned}$$

and

$$\begin{aligned}
\log[\mathrm{End}(E) : R] &\leq 3(15 \cdot \log(p) + \log(d)/2) + 2 - \log p \\
&\leq 45 \cdot \log(p) + 1.5 \cdot \log(d).
\end{aligned}$$

There are two cases where they say that an iteration of the second loop is a success. In the first case, a new factor of $[\mathrm{End}(E) : R]$ is discovered. This can happen at most $\log([\mathrm{End}(E) : R]) - 1$ times. In the second case, $[\mathrm{End}(E) : R]$ is divided by an integer that is at least 2. Likewise, $[\mathrm{End}(E) : R]$ can be divided by an integer that is at least 2 at most $\log([\mathrm{End}(E) : R])$ times. Hence, the second loop succeeds after at most $2\log([\mathrm{End}(E) : R]) \leq 90 \cdot \log(p) + 3 \cdot \log(d)$ successful iterations.

The probability that an iteration of the second loop succeeds is at least

$$\frac{1}{2 \cdot 10^6 \cdot (\log N)^{12}}$$

by [PW24, Proposition 5.11], where N is a variable in the loop that is always a factor of $[\mathrm{End}(E) : R]$. Hence, the number of iterations needed for a success is in expectation at most

$$\begin{aligned}
&2 \cdot 10^6 \cdot (\log N)^{12} \\
&\leq 2 \cdot 10^6 \cdot (45 \cdot \log(p) + 1.5 \cdot \log(d))^{12} \\
&\leq 2^{87} \cdot (\log(p) + \log(d)/30)^{12}.
\end{aligned}$$

In total, the number of calls to OEnd is in expectation at most

$$\begin{aligned}
&48 + (90 \cdot \log(p) + 3 \cdot \log(d)) \cdot 2^{87} \cdot (\log(p) + \log(d)/30)^{12} \\
&= 48 + 90\,(\log(p) + \log(d)/30) \cdot 2^{87} \cdot (\log(p) + \log(d)/30)^{12} \\
&< 2^{94} \cdot (\log(p) + \log(d)/30)^{13}.
\end{aligned}$$

□

6.2 A tighter reduction by factoring

In this section, we prove Theorem 5. Namely, let OEnd be an oracle for the OneEnd_p problem with outputs of degree at most d, and $\mathsf{OFactor}$ an oracle for integer factorization. Then we will show that Algorithm 7 solves the $\mathsf{EndRing}_p$ problem in expected polynomial time in $\log p$ and $\log d$, with a single call to $\mathsf{OFactor}$ and in expectation at most 24 calls to OEnd.

To prove the theorem, let us start by reworking some of the preliminary results of [PW24]. In the following, for any ring R, we write $M_2(R)$ for the ring of 2×2 matrices with coefficients in R, and $\mathrm{SL}_2(R)$ for the multiplicative subgroup of matrices with determinant 1.

Definition 6.1 ([PW24, Definition 5.9]). *Let N be an integer, and $\varepsilon \geq 0$. Let M be a ring with an isomorphism $\iota : M_2(\mathbb{Z}/N\mathbb{Z}) \to M$. The distribution of a random $\alpha \in M/\iota(\mathbb{Z}/N\mathbb{Z})$ is* ε-close to $\mathrm{SL}_2(\mathbb{Z}/N\mathbb{Z})$-invariant *if, for every $g \in \mathrm{SL}_2(\mathbb{Z}/N\mathbb{Z})$, the statistical distance between the distributions of α and $g^{-1}\alpha g$ is at most ε. When the distributions are the same (i.e., $\varepsilon = 0$), we say that the distribution is $\mathrm{SL}_2(\mathbb{Z}/N\mathbb{Z})$-invariant.*

We first refine a result of [PW24] for the case where $N = \ell$ is a prime.

Lemma 6.2. *Let ℓ be an odd prime. Let $\alpha_1, \alpha_2, \alpha_3 \in M_2(\mathbb{F}_\ell)/\mathbb{F}_\ell$ be independent, non-zero, random elements from an $\mathrm{SL}_2(\mathbb{F}_\ell)$-invariant distribution. Then, $(\alpha_1, \alpha_2, \alpha_3)$ is a basis of $M_2(\mathbb{F}_\ell)/\mathbb{F}_\ell$ with probability at least $\left(1 - \frac{4\ell}{\ell^2-1}\right)^3$.*

Proof. Note that [PW24, Lemma 5.3] is almost the same statement, but with probability 1/8. This 1/8 is obtained as the cube of the probability $1 - 1/2$, where the 1/2 is computed in [PW24, Lemma 5.2]. However, this 1/2 in [PW24, Lemma 5.2] is obtained by proving the bound $\frac{4\ell}{\ell^2-1}$ (better than 1/2 when $\ell \geq 11$). Using $\frac{4\ell}{\ell^2-1}$ in place of 1/2 in the proof of [PW24, Lemma 5.3] yields the desired result. □

Definition 6.2 (ℓ-reduced). *For any prime ℓ and ring R, we say that $\alpha \in R$ is ℓ-reduced if $\alpha \notin \mathbb{Z} + \ell R$.*

Proposition 6.1. *Let $N > 910$ be a squarefree odd integer such that its prime factors are all larger that some bound $B \geq 10 \cdot \log(N)$. Let $R = \mathbb{Z}/N\mathbb{Z}$, $M = \mathrm{End}(E)/N\,\mathrm{End}(E) \cong M_2(R)$ and $\overline{M} = M/R$. Consider a distribution ν on $\overline{M}$ that is ε-close to $\mathrm{SL}_2(R)$-invariant, and supported on ℓ-reduced elements for all prime factors ℓ of N.*

1. *Let $\alpha_1, \alpha_2, \alpha_3 \in \overline{M}$ independent random samples with distribution ν. The triple $(\alpha_1, \alpha_2, \alpha_3)$ generates $\overline{M}$ with probability at least $1/2 - 3\varepsilon$.*
2. *Let $\alpha_1, \alpha_2, \alpha_3 \in \mathrm{End}(E)$ independent random elements such that $\alpha_i \bmod (\mathbb{Z} + N\,\mathrm{End}(E))$ follows distribution ν. Let Λ be the lattice generated by $(1, \alpha_1, \alpha_2, \alpha_3)$. Then, $\gcd(N, [\mathrm{End}(E) : \Lambda]) = 1$ with probability at least $1/2 - 3\varepsilon$.*

Proof. **Item 1, with $\varepsilon = 0$.** Let ℓ be a prime factor of N. By the Chinese Remainder Theorem, conditional on $\alpha \bmod (N/\ell)$, the variable $\alpha \bmod \ell$ is $\mathrm{SL}_2(\mathbb{F}_\ell)$-invariant. So $(\alpha_1, \alpha_2, \alpha_3)$ generates $\overline{M}/\ell\overline{M}$ with probability at least $\left(1 - \frac{4\ell}{\ell^2-1}\right)^3$ (Lemma 6.2), and this bound applies independently on each prime factor of N, so $(\alpha_1, \alpha_2, \alpha_3)$ generates $\overline{M}$ with probability at least

$$\prod_{\ell \mid N} \left(1 - \frac{4\ell}{\ell^2 - 1}\right)^3 \geq \left(1 - \frac{4B}{B^2 - 1}\right)^{3\omega(N)}.$$

By [Rob83], we have $\omega(N) \leq 1.3841 \ln(N)/\ln\ln(N) = (1.3841/\log\ln(N))\log(N)$. We get that if $N \geq e^{2^{1.3841 \cdot x}}$, we have $\omega(N) \leq \log(N)/x$. If furthermore $B \geq y\log(N) = z$, we get

$$\prod_{\ell \mid N} \left(1 - \frac{4\ell}{\ell^2 - 1}\right)^3 \geq \left(1 - \frac{4y\log(N)}{y^2\log(N)^2 - 1}\right)^{3\log(N)/x} = f\left(y\log(N)\right)^{\frac{3}{yx}},$$

where $f : z \mapsto \left(1 - \frac{4z}{z^2-1}\right)^z$. This function f is increasing for $z \geq 2 + \sqrt{5}$, so for $z \geq 20$, we have $f(z) \geq f(20) \geq 0.011$.

So if $N \geq e^{2^{1.3841 \cdot x}}$, and $B \geq y\log(N) \geq 20$, we obtain

$$\prod_{\ell \mid N} \left(1 - \frac{4\ell}{\ell^2 - 1}\right)^3 \geq 0.011^{\frac{3}{yx}}.$$

When $xy \geq 20$, we have $0.011^{\frac{3}{yx}} > 0.5$. We obtain the result by setting $x = 2$ and $y = 10$ (in particular, $e^{2^{1.3841 \cdot x}} \approx 909.2$).

Item 1, with $\varepsilon > 0$. Applying the triangular inequality, the random triple $(\alpha_1, \alpha_2, \alpha_3)$ is at total variation distance at most 3ε from a triple of $\mathrm{SL}_2(\mathbb{Z}/N\mathbb{Z})$-invariant random variables. We conclude from the case $\varepsilon = 0$ and the defining property of the total variation distance.

Item 2. This is a consequence of Item 1, [PW24, Proposition 5.6]. □

Given an oracle OEnd for the OneEnd_p problem, the article [PW24, Algorithm 1] describes a procedure $\mathsf{Rich}_k^{\mathsf{OEnd}}$ which also solves OneEnd_p, but with the additional guarantee that its output is a random endomorphism, for a distribution which is ε-close to SL_2-invariant. Let us recall the formal result below.

Proposition 6.2. *Let $p > 3$ be a prime, N an odd integer, and $\varepsilon > 0$. Let*

$$k = \left\lceil \frac{\log\left(\frac{1}{\varepsilon} \cdot \frac{1+\sqrt{3}}{4} \cdot N^2\sqrt{p+13}\right)}{\log\left(\frac{3}{2\sqrt{2}}\right)} \right\rceil$$

Let OEnd be an oracle for the OneEnd_p problem. Let $E/\mathbb{F}_{p^2}$ be a supersingular elliptic curve, and α be a random endomorphism produced by $\mathsf{Rich}_k^{\mathsf{OEnd}}(E)$. The distribution of $\alpha \bmod (\mathbb{Z} + N\,\mathrm{End}(E))$ is ε-close to $\mathrm{SL}_2(\mathbb{Z}/N\mathbb{Z})$-invariant.

Proof. This is a reformulation of [PW24, Theorem 4.2]. □

With a procedure to generate endomorphisms which are ε-close to SL_2-invariant (Proposition 6.2), and a proof that such endomorphisms are likely to form (local) bases of the endomorphism ring (Proposition 6.1), we can now prove Theorem 5.

Proof of Theorem 5. Let us analyse Algorithm 7. The first loop terminates as soon as $\mathrm{rank}_{\mathbb{Z}}(R) = 4$. By construction of the length of the walk k_1, Proposition 6.2 ensures that the conditions of Proposition 6.1 are satisfied for $\varepsilon = 1/12$ and $N = 911$ (the smallest prime larger than 910). In particular, at each iteration of the loop, the elements $(1, \alpha_1, \alpha_2, \alpha_3)$ are linearly independent with probability as least $1/2 - 3\varepsilon = 1/4$. Since each iteration calls OEnd three times, this loop incurs an expected 12 calls to OEnd.

The loop of Line 13 ensures that, after that point, the squarefree part N_R of $[\mathrm{End}(E) : R]$ satisfies the condition for N is Proposition 6.1. Now, the choice of k_2 following Proposition 6.2 ensures that the random endomorphisms α_i on Line 20 are ε-close to $\mathrm{SL}_2(\mathbb{Z}/N\mathbb{Z})$-invariant (for $\varepsilon = 1/12$), and after the loop of Line 21, they are also reduced at every prime factor of N_R. In particular, all the conditions for Proposition 6.1, Item 2, are satisfied. Therefore, $\gcd(N_R, [\mathrm{End}(E) : \Lambda]) = 1$ with probability at least $1/2 - 3\varepsilon = 1/4$, where Λ is the lattice generated by $(1, \alpha_1, \alpha_2, \alpha_3)$. When that happens, $\gcd([\mathrm{End}(E) : R], [\mathrm{End}(E) : \Lambda]) = 1$, so $[\mathrm{End}(E) : R + \Lambda] = 1$, and the loop terminates. This proves that the expected number if iterations of the loop of Line 16 is at most 4. Since each iteration calls OEnd three times, this loop incurs an expected 12 additional calls to OEnd. □

7 Discussion and future work

We provided a full proof of security of SQIsign: we gave a reduction of the EUF-CMA security of SQIsign in the ROM to the hardness of two non-interactive problems, the endomorphism ring problem with hints and the hint indistinguishability problem. Along the way, we developed several technical tools: the framework of Fiat–Shamir with hints, which captures all HD variants of SQIsign and possibly more; a significantly tighter quantum reduction between the Endomorphism Ring problem and the One Endomorphism problem, both with and without hints; several intermediate results, such as a proof EUF-CMA security that does not rely on the hint indistinguishability problem, or a polynomial-time algorithm to sample degrees of uniformly random isogenies of bounded degree. Overall, this work contributes to identify the precise assumptions needed for the security of SQIsign and provide further evidence of its security.

Algorithm 7: Reducing $\mathsf{EndRing}_p$ to OneEnd_p with a factoring oracle

Input: A supersingular elliptic curve $E \in \mathsf{Supersingular}_p$, an oracle OEnd for OneEnd_p, and an oracle $\mathsf{OFactor}$ for factoring integers.
Output: The endomorphism ring $\mathrm{End}(E)$.

$R \coloneqq \mathbb{Z}$.
$N_1 \coloneqq 911$.
$\varepsilon \coloneqq 1/12$.
$k_1 \coloneqq \left\lceil \frac{\log\left(\frac{1}{\varepsilon}\cdot\frac{1+\sqrt{3}}{4}\cdot N_1^2\sqrt{p+13}\right)}{\log\left(\frac{3}{2\sqrt{2}}\right)} \right\rceil$.
while $\mathrm{rank}_{\mathbb{Z}}(R) \neq 4$ **do**
    $\alpha_j \leftarrow \mathsf{Rich}_{k_1}^{\mathsf{OEnd}}(E)$, for $j \in \{1,2,3\}$, three random endomorphisms of E. // [PW24, Algorithm 1]
    $\alpha_j \coloneqq \mathsf{Reduce}_{N_1}(\alpha_j)$, for $j \in \{1,2,3\}$. // [PW24, Algorithm 4]
    $R \coloneqq$ the ring generated by $R, \alpha_1, \alpha_2, \alpha_3$.
$R \coloneqq \mathsf{Saturate}_2(R)$. // [PW24, Algorithm 2]
$R \coloneqq \mathsf{SaturateRam}(R)$. // [PW24, Algorithm 3]
$[\mathrm{End}(E) : R] \coloneqq \sqrt{\mathrm{disc}(R)}/p$.
$\prod_{i=1}^{t} \ell_i^{e_i} \coloneqq \mathsf{OFactor}([\mathrm{End}(E) : R])$.
for $i \in \{i \mid \ell_i \leq \max(910, 10 \cdot \log([\mathrm{End}(E) : R]))\}$ **do**
    $R \coloneqq \mathsf{Saturate}_{\ell_i}(R)$. // [PW24, Algorithm 2]
Update $[\mathrm{End}(E) : R] \coloneqq \sqrt{\mathrm{disc}(R)}/p$ and its factorisation $[\mathrm{End}(E) : R] = \prod_{i=1}^{t} \ell_i^{e_i}$.
while $[\mathrm{End}(E) : R] \neq 1$ **do**
    $N_R \coloneqq$ squarefree part of $[\mathrm{End}(E) : R]$.
    $\varepsilon \coloneqq 1/12$.
    $k_2 \coloneqq \left\lceil \frac{\log\left(\frac{1}{\varepsilon}\cdot\frac{1+\sqrt{3}}{4}\cdot N_R^2\sqrt{p+13}\right)}{\log\left(\frac{3}{2\sqrt{2}}\right)} \right\rceil$.
    $\alpha_j \leftarrow \mathsf{Rich}_{k_2}^{\mathsf{OEnd}}(E)$, for $j \in \{1,2,3\}$, three random endomorphisms of E. // [PW24, Algorithm 1]
    for $i \in \{1, 2, \ldots, t\}$, $j \in \{1,2,3\}$ **do**
        $\alpha_j \coloneqq \mathsf{Reduce}_{\ell_i}(\alpha_j)$. // [PW24, Algorithm 4]
    $R \coloneqq$ the ring generated by $R, \alpha_1, \alpha_2, \alpha_3$.
    Update $[\mathrm{End}(E) : R] \coloneqq \sqrt{\mathrm{disc}(R)}/p$ and its factorisation $[\mathrm{End}(E) : R] = \prod_{i=1}^{t} \ell_i^{e_i}$.
return R.

Remaining gaps and limitations. The implementation of SQIsign, in its round-2 submission to the NIST standardization process, differs slightly from the protocol analyzed so far: for efficiency reasons, the implementation relies on some algorithms that may possibly fail instead of producing the desired output. These failure cases can be split into two categories: those that fail with negligible probability and therefore do not affect security, and those that fail with small but not negligible probability (approximately 2^{-64}, according to [AAA+25]). This second type of failures has a more significant impact on security: while they are unlikely to be practically exploitable, they introduce a bias in the public keys and signatures that is not captured by our security analysis.

Furthermore, our result in Theorem 6 shows a loss factor that is quadratic in the number of signing queries, which is then divided by $\sqrt{p}$. This implies that, for adversaries that make exponentially many signing queries (say, 2^{64} queries against a prime $p \approx 2^{256}$, as in NIST security level I), the reduction becomes vacuous.

Takeaways and recommendations. In light of the previous discussion, we invite the research community to further investigate the algorithmic building blocks of SQIsign that currently have non-negligible failure probability. We expect that developing better algorithms that maintain the same efficiency while obtaining negligible failure probability is within reach. This would close the remaining gap between the theoretical analysis and the implemented version of SQIsign.

Similarly, we suggest that future revisions of SQIsign bring the commitment min-entropy to within $1/p$, rather than $1/\sqrt{p}$. This could be easily achieved at almost no cost by increasing N_{mix} to $\approx 2^{8\lambda}$, but possibly even more efficient solutions exist. A smaller min-entropy would increase the denominator in the loss factor to p, which would then make the statement of Theorem 6 meaningful even in the presence of attackers with 2^{64} queries or more.

Future work. Lastly, we leave the analysis of additional security properties for future work. This includes studying strong unforgeability (which, in its current formulation, SQIsign is unlikely to achieve), the three BUFF properties [CDF+21], and a security proof in the quantum random oracle model. The current techniques for a QROM reduction do not seem to apply to SQIsign, so further research is needed.

Acknowledgments. We thank the whole SQIsign team for their help and support throughout this project.

This work was funded by ERC grant No. 101116169 (AGATHA CRYPTY); France 2030 program, grants No. ANR-22-PETQ-0008 (PQ-TLS) Danish Independent Research Council, grant No. 1026-00350B (RENAIS); Swiss SNF Consolidator Grant No. 213766 (CryptonIs).

References

AAA+25. Marius A. Aardal, Gora Adj, Diego F. Aranha, Andrea Basso, Isaac Andrés Canales Martínez, Jorge Chávez-Saab, Maria Corte-Real Santos, Pierrick Dartois, Luca De Feo, Max Duparc, Jonathan Komada Eriksen, Tako Boris Fouotsa, Décio Luiz Gazzoni Filho, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Luciano Maino, Michael Meyer, Kohei Nakagawa, Hiroshi Onuki, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Giacomo Pope, Krijn Reijnders, Damien Robert, Francisco Rodríguez Henríquez, Sina Schaeffler, and Benjamin Wesolowski. SQIsign. Technical report, National Institute of Standards and Technology, 2025. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures.

AABN02. Michel Abdalla, Jee Hea An, Mihir Bellare, and Chanathip Namprempre. From identification to signatures via the Fiat-Shamir transform: Minimizing assumptions for security and forward-security. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 418–433. Springer, Berlin, Heidelberg, April / May 2002. doi:10.1007/3-540-46035-7_28.

ACK21. Thomas Attema, Ronald Cramer, and Lisa Kohl. A compressed Σ-protocol theory for lattices. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part II*, volume 12826 of *LNCS*, pages 549–579, Virtual Event, August 2021. Springer, Cham. doi:10.1007/978-3-030-84245-1_19.

Apo76. Tom M. Apostol. *Introduction to Analytic Number Theory*. Springer New York, 1976. doi:10.1007/978-1-4757-5579-4.

Bac88. Eric Bach. How to generate factored random numbers. *SIAM Journal on Computing*, 17(2):179–193, 1988. doi:10.1137/0217012.

BBSS18. Matilda Backendal, Mihir Bellare, Jessica Sorrell, and Jiahao Sun. The Fiat-Shamir zoo: Relating the security of different signature variants. In *NordSec*, volume 11252 of *Lecture Notes in Computer Science*, pages 154–170. Springer, 2018.

BCC+23. Andrea Basso, Giulio Codogni, Deirdre Connolly, Luca De Feo, Tako Boris Fouotsa, Guido Maria Lido, Travis Morrison, Lorenz Panny, Sikhar Patranabis, and Benjamin Wesolowski. Supersingular curves you can trust. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part II*, volume 14005 of *LNCS*, pages 405–437. Springer, Cham, April 2023. doi:10.1007/978-3-031-30617-4_14.

BDD+24. Andrea Basso, Pierrick Dartois, Luca De Feo, Antonin Leroux, Luciano Maino, Giacomo Pope, Damien Robert, and Benjamin Wesolowski. SQIsign2D-west - the fast, the small, and the safer. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 339–370. Springer, Singapore, December 2024. doi:10.1007/978-981-96-0891-1_11.

BDLS20. Daniel J. Bernstein, Luca De Feo, Antonin Leroux, and Benjamin Smith. Faster computation of isogenies of large prime degree. Cryptology ePrint Archive, Report 2020/341, 2020. URL: https://eprint.iacr.org/2020/341.

BLL24. Giacomo Borin, Yi-Fu Lai, and Antonin Leroux. Erebor and durian: Full anonymous ring signatures from quaternions and isogenies. Cryptology ePrint Archive, Report 2024/1185, 2024. URL: https://eprint.iacr.org/2024/1185.

CDF+21. Cas Cremers, Samed Düzlü, Rune Fiedler, Marc Fischlin, and Christian Janson. BUFFing signature schemes beyond unforgeability and the case of post-quantum signatures. In *2021 IEEE Symposium on Security and Privacy*, pages 1696–1714. IEEE Computer Society Press, May 2021. doi:10.1109/SP40001.2021.00093.

CSD+23. Jorge Chavez-Saab, Maria Corte-Real Santos, Luca De Feo, Jonathan Komada Eriksen, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Michael Meyer, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Francisco Rodríguez Henríquez, Sina Schaeffler, and Benjamin Wesolowski. SQIsign. Technical report, National Institute of Standards and Technology, 2023. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures.

CSD+24. Jorge Chavez-Saab, Maria Corte-Real Santos, Luca De Feo, Jonathan Komada Eriksen, Basil Hess, David Kohel, Antonin Leroux, Patrick Longa, Michael Meyer, Lorenz Panny, Sikhar Patranabis, Christophe Petit, Francisco Rodríguez Henríquez, Sina Schaeffler, and Benjamin Wesolowski. SQIsign. Technical report, National Institute of Standards and Technology, 2024. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures.

DF24. Max Duparc and Tako Boris Fouotsa. SQIPrime: A dimension 2 variant of SQISignHD with non-smooth challenge isogenies. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 396–429. Springer, Singapore, December 2024. doi:10.1007/978-981-96-0891-1_13.

DKL+20. Luca De Feo, David Kohel, Antonin Leroux, Christophe Petit, and Benjamin Wesolowski. SQISign: Compact post-quantum signatures from quaternions and isogenies. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part I*, volume 12491 of *LNCS*, pages 64–93. Springer, Cham, December 2020. doi:10.1007/978-3-030-64837-4_3.

DLLW23. Luca De Feo, Antonin Leroux, Patrick Longa, and Benjamin Wesolowski. New algorithms for the deuring correspondence - towards practical and secure SQISign signatures. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part V*, volume 14008 of *LNCS*, pages 659–690. Springer, Cham, April 2023. doi:10.1007/978-3-031-30589-4_23.

DLRW24. Pierrick Dartois, Antonin Leroux, Damien Robert, and Benjamin Wesolowski. SQIsignHD: New dimensions in cryptography. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part I*, volume 14651 of *LNCS*, pages 3–32. Springer, Cham, May 2024. doi:10.1007/978-3-031-58716-0_1.

EHL+18. Kirsten Eisenträger, Sean Hallgren, Kristin E. Lauter, Travis Morrison, and Christophe Petit. Supersingular isogeny graphs and endomorphism rings: Reductions and solutions. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part III*, volume 10822 of *LNCS*, pages 329–368. Springer, Cham, April / May 2018. doi:10.1007/978-3-319-78372-7_11.

EHL+20. Kirsten Eisenträger, Sean Hallgren, Chris Leonardi, Travis Morrison, and Jennifer Park. Computing endomorphism rings of supersingular elliptic curves and connections to path-finding in isogeny graphs. *Open Book Series*, 4(1):215–232, 2020. doi:10.2140/obs.2020.4.215.

HÏ6. Werner Hürlimann. *Journal of Algebra, Number Theory: Advances and Applications*, 14(2):73–88, February 2016. doi:10.18642/jantaa_7100121599.

Kan97. Ernst Kani. The number of curves of genus two with elliptic differentials. *Journal für die reine und angewandte Mathematik*, 1997(485):93–122, 1997.

KV10. Markus Kirschmer and John Voight. Algorithmic enumeration of ideal classes for quaternion orders. *SIAM Journal on Computing*, 39(5):1714–1747, 2010.

Ler22. Antonin Leroux. *Quaternion Algebra and isogeny-based cryptography. (Algèbres de quaternions et cryptographie à base d'isogénies)*. PhD thesis, Polytechnic Institute of Paris, France, 2022.

McM14. Ken McMurdy. Explicit representation of the endomorphism rings of supersingular elliptic curves. https://pages.ramapo.edu/~kmcmurdy/research/McMurdy-ssEndoRings.pdf, 2014. Preprint.

NOC+24. Kohei Nakagawa, Hiroshi Onuki, Wouter Castryck, Mingjie Chen, Riccardo Invernizzi, Gioella Lorenzon, and Frederik Vercauteren. SQIsign2D-east: A new signature scheme using 2-dimensional isogenies. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 272–303. Springer, Singapore, December 2024. doi:10.1007/978-981-96-0891-1_9.

OEI25. OEIS Foundation Inc. Entry A173290 in The On-Line Encyclopedia of Integer Sequences, 2025. URL: https://oeis.org/A173290.

PS96. David Pointcheval and Jacques Stern. Security proofs for signature schemes. In Ueli M. Maurer, editor, *EUROCRYPT'96*, volume 1070 of *LNCS*, pages 387–398. Springer, Berlin, Heidelberg, May 1996. doi:10.1007/3-540-68339-9_33.

PW24. Aurel Page and Benjamin Wesolowski. The supersingular endomorphism ring and one endomorphism problems are equivalent. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part VI*, volume 14656 of *LNCS*, pages 388–417. Springer, Cham, May 2024. doi:10.1007/978-3-031-58751-1_14.

Rob83. Guy Robin. Estimation de la fonction de Tchebychef θ sur le k-ième nombre premier et grandes valeurs de la fonction $\omega(n)$ nombre de diviseurs premiers de n. *Acta Arithmetica*, 42(4):367–389, 1983.

Rob22. Damien Robert. Evaluating isogenies in polylogarithmic time. Cryptology ePrint Archive, Report 2022/1068, 2022. URL: https://eprint.iacr.org/2022/1068.

Rob24. Damien Robert. On the efficient representation of isogenies (a survey). Cryptology ePrint Archive, Report 2024/1071, 2024. URL: https://eprint.iacr.org/2024/1071.

Sho94. P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994. doi:10.1109/SFCS.1994.365700.

Sil86. Joseph H. Silverman. *The Arithmetic of Elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer, 1986.

Wes22. Benjamin Wesolowski. The supersingular isogeny path and endomorphism ring problems are equivalent. In *62nd FOCS*, pages 1100–1111. IEEE Computer Society Press, February 2022. doi:10.1109/FOCS52979.2021.00109.