SIDH Proof of Knowledge

Luca De Feo[1], Samuel Dobson[2], Steven D. Galbraith[2], and Lukas Zobernig[2]

[1]IBM Research Europe. luca@defeo.lu
[2]Mathematics Department, University of Auckland, New Zealand.
samuel.dobson.nz@gmail.com, s.galbraith@auckland.ac.nz,
lukas.zobernig@auckland.ac.nz

October 19, 2021

Abstract

We demonstrate the soundness proof for the De Feo–Jao–Plût identification scheme (the basis for SIDH signatures) contains an invalid assumption and provide a counterexample for this assumption—thus showing the proof of soundness is invalid. As this proof was repeated in a number of works by various authors, multiple pieces of literature are affected by this result. Due to the importance of being able to prove knowledge of an SIDH key (for example, to prevent adaptive attacks), soundness is a vital property. We propose a modified identification scheme fixing the issue with the De Feo–Jao–Plût scheme, and provide a proof of security of this new scheme. We also prove that a modification of this scheme allows the torsion points in the public key to be verified too. This results in a secure proof of knowledge for SIDH keys. In particular, these schemes provide a non-interactive way of verifying that SIDH public keys are well formed as protection against adaptive attacks, more efficient than generic NIZKs.

1 Introduction

While Supersingular Isogeny Diffie-Hellman (SIDH) [JD11, DJP14] is a fast and efficient post-quantum key exchange candidate, it has been hampered by the existence of practical adaptive attacks on the scheme—the first of these given by Galbraith et al. [GPST16] (the GPST attack). These attacks mean it is not safe to re-use a static key across multiple SIDH exchanges without other forms of protection. As such, various countermeasures have been proposed—though each with their unique drawbacks.

The first of these is to require one participant to use a one-time ephemeral key in the exchange, accompanied by a Fujisaki-Okamoto-type transform [HHK17] revealing the corresponding secret to the other party. This allows the recipient to verify the public key is well formed, ensuring an adaptive attack was not used. This is what was done in SIKE [ACC+17], and converts the scheme to a secure key encapsulation mechanism (KEM). But it is of limited use in cases where both parties wish to use a long-term key.

The second countermeasure is to use many SIDH exchanges in parallel, combining all the resulting secrets into a single value, as proposed by Azarderakhsh, Jao, and Leonardi [AJL17]. This scheme is known as k-SIDH, where k is the number of keys used by each party in the exchange. The authors suggest $k = 92$ is required for a secure key exchange, as Dobson et al. [DGL+20] demonstrate how the GPST adaptive attack can be ported to $k = 2$ and above. Note that the number of SIDH instances grows as k^2, so this scheme is very inefficient. Urbanik and Jao's [UJ20] proposal attempted to improve the efficiency of this protocol by making use of the special automorphisms on curves with j-invariant 0 or 1728, but it was shown by Basso et al. [BKM+20] that Urbanik and Jao's proposal is vulnerable to a more efficient adaptive attack and actually scales worse in efficiency than k–SIDH itself (although the public keys are around 4/5 of the size, it requires around twice as many SIDH instances for the same security).

Finally, adaptive attacks can also be prevented by providing a non-interactive proof that a public key is well-formed or honestly generated. While generic NIZKs would make this possible in a very inefficient manner, Urbanik and Jao [UJ20] claim a method for doing so using a similar idea to their k-SIDH improvement mentioned above. Their scheme is based on the SIDH-based identification scheme by De Feo, Jao, and Plût [DJP14].

Unfortunately, however, we show that the soundness of this original De Feo–Jao–Plût scheme is not rigorously proved—specifically that it does not reduce to the computational assumption they claim—and give a counterexample to this proof. Because this scheme (and proof) has since been used to build an undeniable signature by Jao and Soukharev [JS14], a signature scheme by Yoo, Azarderakhsh, Jalali, Jao, and Soukharev [YAJ+17], and also by Galbraith, Petit, and Silva [GPS20], all of these subsequent papers suffer from the same issue. Our counterexample does not apply to Urbanik and Jao's scheme, but their soundness proof nonetheless does not hold for the same reason.

In this work we examine the issue with the existing soundness proofs and propose a new SIDH-based identification scheme which we prove does satisfy special soundness. We then propose a modification to the scheme which allows the two torsion points in the public key to be proved correct as well, which was not covered by De Feo, Jao, and Plût's scheme. This gives a secure method for proving well-formedness of SIDH public keys—the first sound Proof of Knowledge protocol of a secret isogeny for a given public key—with important applications in all areas where SIDH key exchanges could be used with static keys. What's more, our scheme works with any base elliptic curve, rather than being restricted to the two curves with j-invariant 0 or 1728 as in [UJ20]. While the size of our NIZK proof is larger than a k-SIDH public key of the same security level, it is much more efficient to verify than computing a k-SIDH exchange (due to the quadratic scaling mentioned above).

In concurrent independent work, Ghantous et al. [GPV21] have demonstrated that the soundness property for the De Feo–Jao–Plût scheme (and those based on it) fails for a different reason—namely the existence of multiple isogenies of the same degree between some curves. The new scheme we propose in this paper does not suffer from the issue Ghantous et al. analyze, but this further solidifies the need for a sound replacement to prove honest generation of SIDH public keys—of which ours is the first.

1.1 Outline

This work begins in Section 2 with revision of some preliminary background material. This is followed by a discussion of some relevant isogeny-based hardness assumptions and reductions in Section 3. We then recall the De Feo–Jao–Plût identification scheme in Section 4.1 and outline the issue with its proof of soundness (given in multiple previous works) in Section 4.2. Subsequently, we present a new SIDH identification scheme in Section 5 which modifies the De Feo–Jao–Plût scheme and allows us to prove soundness (and thus security). We then show how the points in the SIDH public key can also be verified under this identification scheme in Section 6, and discuss improvements to the efficiency of this scheme. From this, we construct a secure signature scheme which is a Proof of Knowledge (PoK) of an SIDH secret key, and is the first such scheme which is sound and proves correctness of the points in the public key (a protection mechanism against adaptive attacks [GPST16, DGL+20]) in Section 7.

1.2 Acknowledgements

We thank David Jao, Jason LeGrow, and Yi-Fu Lai for useful discussion about this work. We also thank Paulo Barreto for catching some typos in this paper, and Simon-Philipp Merz for valuable comments. Finally, we would like to thank those involved with the BIRS Supersingular Isogeny Graphs in Cryptography workshop for great discussion on some questions this work raised—especially Lorenz Panny and his work analyzing SIDH squares in small fields.

2 Preliminaries

Notation. As a convention, we will use K_ϕ to denote a point which generates the kernel of an isogeny ϕ. Let $[t]$ denote the set $\{1, \ldots, t\}$.

2.1 SIDH

We now provide a brief refresher on the Supersingular Isogeny Diffie-Hellman (SIDH) key exchange protocol [JD11, DJP14] by De Feo, Jao, and Plût.

As public parameters, we have a prime $p = \ell_1^{e_1} \cdot \ell_2^{e_2} \cdot f \pm 1$, where ℓ_1, ℓ_2 are small primes, f is an integer cofactor, and $\ell_1^{e_1} \approx \ell_2^{e_2}$. We work over the finite field $\mathbb{F}_{p^2}$. Additionally we fix a base supersingular elliptic curve E_0 and bases $\{P_1, Q_1\}$, $\{P_2, Q_2\}$ for both the $\ell_1^{e_1}$ and $\ell_2^{e_2}$-torsion subgroups of $E_0(\mathbb{F}_{p^2})$ respectively (such that $E_0[\ell_i^{e_i}] = \langle P_i, Q_i \rangle$). Typically $\ell_1 = 2$ and $\ell_2 = 3$.

It is well known that knowledge of an isogeny and knowledge of its kernel are equivalent, and we can convert between them at will, via Vélu's formulae [Vél71]. In SIDH, the secret keys of Alice and Bob are isogenies $\phi_A : E(\mathbb{F}_{p^2}) \to E_A(\mathbb{F}_{p^2})$, $\phi_B : E(\mathbb{F}_{p^2}) \to E_B(\mathbb{F}_{p^2})$ of degree $\ell_1^{e_1}$ and $\ell_2^{e_2}$, respectively. These isogenies are generated by randomly choosing secret integers $a_i, b_i \in \mathbb{Z}/\ell_i^{e_i}\mathbb{Z}$ (not both divisible by ℓ_i) and computing the isogeny with kernel $K_i = \langle [a_i]P_i + [b_i]Q_i \rangle$. We thus unambiguously refer to the isogeny, its kernel, and such integers a, b, as "the secret key." Figure 1 depicts the commutative diagram making up the key exchange.

$$\begin{tikzcd}
E \arrow[r, "\phi_A"] \arrow[d, "\phi_B"'] & E_A \arrow[d, "\phi_{AB}"] \\
E_B \arrow[r, "\phi_{BA}"'] & E_{AB}
\end{tikzcd}$$

Figure 1: Commutative diagram of SIDH, where $\ker(\phi_{BA}) = \phi_B(\ker(\phi_A))$ and $\ker(\phi_{AB}) = \phi_A(\ker(\phi_B))$.

In order to make the diagram commute, Alice and Bob are required to not just give their image curves E_A and E_B in their respective public keys, but also the images of the basis points of the other participant's kernel on E. That is, Alice provides E_A, $P_2' = \phi_A(P_2), Q_2' = \phi_A(Q_2)$ as her public key. This allows Bob to "transport" his secret isogeny to E_A and compute ϕ_{AB} whose kernel is $\langle [a_2]P_2' + [b_2]Q_2' \rangle$. Both Alice and Bob will arrive along these transported isogenies at isomorphic image curves E_{AB}, E_{BA} (using Vélu's formulae, they will actually arrive at exactly the same curve). Two elliptic curves are isomorphic over $\overline{\mathbb{F}}_{p^2}$ if and only if their j-invariants $j(E_{AB}) = j(E_{BA})$, hence this j-invariant may be used as the shared secret of the SIDH key exchange.

Some cryptographic hardness assumptions related to isogenies and SIDH are discussed in Section 3.

Remark 1. Galbraith et al. [GPST16, Lemma 2.1] formally presented the idea of "equivalent keys" (which were implicit in previous works including Costello et al. [CLN16]). Two secret keys (a, b) and (a', b') are equivalent if they generate the same subgroup for any basis of the $\ell_i^{e_i}$-torsion subgroup. This is true when $(a', b') = (\theta a, \theta b)$ for $\theta \in \mathbb{Z}_{\ell_i}^*$. Because we have the condition that at least one of a, b is not divisible by ℓ_i (assume for now this is a), a is invertible modulo $\ell_i^{e_i}$. Thus we can choose $\theta \equiv a^{-1} \pmod{\ell_i^{e_i}}$. This gives an equivalent key $(1, b')$. Similarly, if b was not divisible by ℓ_i, we can invert it and obtain equivalent key $(a', 1)$. Hence we obtain a shorter representation of secret keys without loss of generality, to a single element and one extra bit.

2.2 Sigma protocols

A sigma protocol Π_Σ for a relation $\mathcal{R} = \{(X, W)\}$ is a public-coin three-move interactive proof system consisting of two parties: a verifier V and a prover P.

Definition 1 (Sigma protocol)**.** *A sigma protocol Π_Σ for a family of relations $\{\mathcal{R}\}_\kappa$ parametrized by security parameter κ consists of PPT algorithms $((P_1, P_2), (V_1, V_2))$ where V_2 is deterministic and we assume P_1, P_2 share states. The protocol proceeds as follows:*

1. *Round 1: The prover, on input $(X, W) \in \mathcal{R}$, returns a commitment $\mathsf{com} \leftarrow P_1(X, W)$ and sends com to the verifier.*
2. *Round 2: The verifier, on receipt of com, runs $\mathsf{chall} \leftarrow V_1(1^\kappa)$ to obtain a random challenge, and sends this to the prover.*
3. *Round 3: The prover then runs $\mathsf{resp} \leftarrow P_2(X, W, \mathsf{chall})$ and returns resp to the verifier.*
4. *Verification: The verifier runs $V_2(X, \mathsf{com}, \mathsf{chall}, \mathsf{resp})$ and outputs either $\top$ (accept) or $\perp$ (reject).*

A transcript $(\mathsf{com}, \mathsf{chall}, \mathsf{resp})$ is said to be valid if $V_2(X, \mathsf{com}, \mathsf{chall}, \mathsf{resp})$ outputs $\top$. Let $\langle P, V\rangle$ denote the transcript for interaction between prover P and verifier V. Relevant properties of a sigma protocol are:

Correctness: If the prover P knows $(X, W) \in \mathcal{R}$ and behaves honestly, then the verifier V accepts.

2-special soundness: There exists a polynomial time extraction algorithm $\mathsf{Extract}$, which given a statement X and two valid transcripts $(\mathsf{com}, \mathsf{chall}, \mathsf{resp})$ and $(\mathsf{com}, \mathsf{chall}', \mathsf{resp}')$ where $\mathsf{chall} \neq \mathsf{chall}'$, outputs a witness W such that $(X, W) \in \mathcal{R}$ with probability at least $1 - \varepsilon$ for soundness error ε.

Zero Knowledge (ZK): There exists a polynomial time simulator Sim, which given a statement X for any $(X, W) \in \mathcal{R}$, and for any (cheating) verifier V^*, outputs transcripts $(\mathsf{com}, \mathsf{chall}, \mathsf{resp})$ that are indistinguishable from valid interactions between a prover P and V^*.

Proof of Knowledge (PoK): There exists a polynomial time extraction algorithm $\mathsf{Extract}$, which given an arbitrary statement X and access to any prover P^*, outputs a witness W such that $(X, W) \in \mathcal{R}$ with probability at least $\Pr[\langle P^*, V\rangle = 1] - \varepsilon$ for knowledge error ε.

It is a known result (e.g. by Hazay and Lindell [HL10, Theorem 6.3.2]) that a correct and special-sound sigma protocol with challenge length t is a proof of knowledge with knowledge error 2^{-t}. In this paper, this will generally be a single-bit challenge sigma protocol repeated with t iterations.

2.3 Seed trees

We briefly recall the definition of a seed tree from Beullens et al. [BKP20]. A seed tree is used to generate a number of pseudorandom values and later disclose an arbitrary subset of them, without revealing any information about the other values in the tree that were not disclosed.

A seed tree is formed of λ-bit seed values, where the left (resp. right) child of a node seed_h is the left (resp. right) half of $\mathsf{Expand}(\mathsf{seed}\|h)$, where Expand is a pseudorandom generator (PRG) outputting 2λ bits and h is a unique identifier for the position of seed in the binary tree. An arbitrary subset of the leaf values can be efficiently revealed by disclosing the values of an appropriate set of internal nodes in the tree.

Informally, a seed tree consists of the following four algorithms. In the random oracle model, the PRG Expand would be modelled with a random oracle $\mathcal{O}$.

- $\mathsf{SeedTree}(\mathsf{seed}_{\mathsf{root}}, M) \rightarrow \{\mathsf{leaf}_i\}_{i \in [M]}$: On input a root seed $\mathsf{seed}_{\mathsf{root}} \in \{0, 1\}^\lambda$ and an integer $M \in \mathbb{N}$, it constructs a complete binary tree with M leaves by recursively expanding each seed to obtain its children seeds, as above. The output is the list of the M leaf values in the tree.

- $\mathsf{ReleaseSeeds}(\mathsf{seed}_{\mathsf{root}}, \mathbf{c}) \rightarrow \mathsf{seeds}_{\mathsf{internal}}$: On input a root seed $\mathsf{seed}_{\mathsf{root}} \in \{0,1\}^\lambda$, and a challenge $\mathbf{c} \in \{0,1\}^M$, it outputs the list of seeds $\mathsf{seeds}_{\mathsf{internal}}$ that covers all the leaves with index i such that $c_i = 0$. Here, we say that a set of nodes D *covers* a set of leaves S if the union of the leaves of the subtrees rooted at each node $v \in D$ is exactly the set S.
- $\mathsf{RecoverLeaves}(\mathsf{seeds}_{\mathsf{internal}}, \mathbf{c}) \rightarrow \{\mathsf{leaf}_i\}_{i \text{ s.t. } c_i=0}$: On input a set $\mathsf{seeds}_{\mathsf{internal}}$ and a challenge $\mathbf{c} \in \{0,1\}^M$, it computes and outputs all the leaves of subtrees rooted at seeds in $\mathsf{seeds}_{\mathsf{internal}}$. By construction, this is exactly the set $\{\mathsf{leaf}_i\}_{i \text{ s.t. } c_i=0}$.
- $\mathsf{SimulateSeeds}(\mathbf{c}) \rightarrow \mathsf{seeds}_{\mathsf{internal}}$: On input a challenge $\mathbf{c} \in \{0,1\}^M$, it computes the set of nodes covering the leaves with index i such that $c_i = 0$. It then randomly samples a seed from $\{0,1\}^\lambda$ for each of these nodes, and finally outputs the set of these seeds as $\mathsf{seeds}_{\mathsf{internal}}$.

By construction, the leaves $\{\mathsf{leaf}_i\}_{i \text{ s.t. } c_i=0}$ output by $\mathsf{SeedTree}(\mathsf{seed}_{\mathsf{root}}, M)$ are the same as those output by $\mathsf{RecoverLeaves}(\mathsf{ReleaseSeeds}(\mathsf{seed}_{\mathsf{root}}, \mathbf{c}), \mathbf{c})$ for any $\mathbf{c} \in \{0,1\}^M$. The last algorithm $\mathsf{SimulateSeeds}$ can be used to argue that the seeds associated with all the leaves with index i such that $c_i = 1$ are indistinguishable from uniformly random values for a recipient that is only given $\mathsf{seeds}_{\mathsf{internal}}$ and $\mathbf{c}$.

3 SIDH problems and assumptions

In this section, we recall some standard isogeny-based hardness assumptions of relevance to this work. We then introduce a new decisional and computational assumption pair which will be useful for the proof of zero-knowledge in Section 5, and show a reduction from the computational problem to the decisional one.

The first two are computational isogeny-finding problems.

Definition 2 (General isogeny problem)**.** *Given j-invariants $j, j' \in \mathbb{F}_{p^2}$, find an isogeny $\phi : E \rightarrow E'$ if one exists, where $j(E) = j$ and $j(E') = j'$.*

This is the foundational hardness assumption of isogeny-based cryptography, that it is hard to find an isogeny between two given curves. Note the decisional version, determining whether an isogeny exists, is easy—an isogeny exists if and only if $\#E(\mathbb{F}_{p^2}) = \#E'(\mathbb{F}_{p^2})$.

Definition 3 (Computational Supersingular Isogeny (CSSI) problem)**.** *For fixed SIDH public parameters $(p, E_0, P_1, Q_1, P_2, Q_2)$, let $\phi : E_0 \rightarrow E_A$ be an isogeny of degree $\ell_1^{e_1}$. Given the SIDH public key $(E_A, P = \phi(P_2), Q = \phi(Q_2))$, find an isogeny $\phi' : E_0 \rightarrow E_A$ of degree $\ell_1^{e_1}$ such that $P, Q = \phi'(P_2), \phi'(Q_2)$.*

This is problem 5.2 of [DJP14], and essentially states that it is hard to find the secret key corresponding to a given public key. This problem is also called the SIDH isogeny problem by [GV18, Definition 2].

At the heart of the adaptive attack is the problem that, given a public key (E_1, P, Q), we cannot validate that P, Q are indeed the correct images of basis points P_0, Q_0 under the secret isogeny ϕ. The best we can do is to check they are indeed a basis of the correct order, and use the Weil pairing check from Galbraith et al. [GPST16]:

$$e_N(P, Q) = e_N(P_0, Q_0)^{\deg \phi}.$$

Unfortunately this holds for many different choices of basis points, hence this check is not enough to uniquely determine ϕ (and in particular, is insufficient to protect against the GPST adaptive attack). For example, note that there are $\ell_2^{4e_2-3} \cdot (\ell^2-1)^2/(\ell+1)$ different possible choices for ordered linearly independent basis P, Q of the correct order—this is because there are $\ell_2^{2e_2} - \ell_2^{2(e_2-1)}$ points of the correct order, and the independence between P and Q introduces a factor of $\ell_2/(\ell_2+1)$. Yet, only $(\ell_2+1) \cdot \ell_2^{e_2-1}$ different isogenies of order $\ell_2^{e_2}$ exist. Hence, for any particular choice of coefficients for the basis points, there must be a great deal of overlap in the kernels they generate. If $\ell_2 = 3$, we would have $16 \cdot 3^{3e_2-2}$ different choices of points for each kernel. Obviously, many of these choices will not satisfy the Weil pairing check. However, the codomain of $e_{\ell_2^{e_2}}$ has order $\ell_2^{e_2}$, which is much smaller than the number of choices of points.

The following decisional problem follows Definition 3 of [GV18], and is also very similar to the key validation problem of Urbanik and Jao [UJ18, Problem 3.4] (the key validation problem asks whether a ϕ of degree *dividing* $\ell_1^{e_1}$ exists). However, the previous definitions did not take the Weil pairing check into account, which would serve as a distinguisher.

Definition 4 (Decisional SIDH isogeny (DSIDH) problem)**.** *The decisional SIDH problem is to distinguish between the following two distributions:*

- $(E_0, P_2, Q_2, E_1, P', Q')$ *such that* E_0 *is a supersingular elliptic curve defined over* $\mathbb{F}_{p^2}$*,* P_2, Q_2 *a basis such that* $E_0[\ell_2^{e_2}] = \langle P_2, Q_2 \rangle$*,* $\phi : E_0 \to E_1$ *is an isogeny of degree* $\ell_1^{e_1}$*, and* $P' = \phi(P_2)$ *and* $Q' = \phi(Q_2)$*.*
- $(E_0, P_2, Q_2, E_1, P', Q')$ *such that* E_0 *is a supersingular elliptic curve defined over* $\mathbb{F}_{p^2}$*,* P_2, Q_2 *a basis such that* $E_0[\ell_2^{e_2}] = \langle P_2, Q_2 \rangle$*,* E_1 *is any supersingular elliptic curve over* $\mathbb{F}_{p^2}$ *with the same cardinality as* E_0*, and* P', Q' *is a basis of* $E_1[\ell_2^{e_2}]$ *satisfying the Weil pairing check* $e_N(P', Q') = e_N(P_2, Q_2)^{\ell_1^{e_1}}$*.*

As shown by Galbraith and Vercauteren [GV18], Thormarker [Tho17], and Urbanik and Jao [UJ18], being able to solve this decisional problem is as hard as solving the computational (CSSI) problem, so key validation is fundamentally difficult. This is done by testing ℓ_1-isogeny neighboring curves of E_1 and learning the correct path one bit at a time.

Definition 5 (Decisional Supersingular Product (DSSP) problem)**.** *Let* E_0, E_1 *be supersingular elliptic curves such that there exists an isogeny* $\phi : E_0 \to E_1$ *of degree* $\ell_1^{e_1}$ *between them. Let* $P_2, Q_2 \in E_0[\ell_2^{e_2}]$ *be a fixed basis of the* $\ell_2^{e_2}$*-torsion subgroup. Suppose we have the following two distributions:*

- (E_2, E_3, ϕ') *such that there exists a cyclic subgroup* $G \subseteq E[\ell_2^{e_2}]$ *of order* $\ell_2^{e_2}$ *and* $E_2 \cong E_0/G$ *and* $E_3 \cong E_1/\phi(G)$*, and* $\phi' : E_2 \to E_3$ *is a degree* $\ell_1^{e_1}$ *isogeny.*
- (E_2, E_3, ϕ') *such that* E_2 *is a random supersingular curve with the same cardinality as* E_0*, and* E_3 *is the codomain of a random isogeny* $\phi' : E_2 \to E_3$ *of degree* $\ell_1^{e_1}$*.*

The Decisional Supersingular Product problem is, given E_0, E_1 *as well as the points* $P_2, Q_2, \phi(P_2), \phi(Q_2)$*, and given a tuple* (E_2, E_3, ϕ') *drawn randomly with probability 1/2 from the above two distributions, to determine which of the two distributions it was drawn from.*

This is problem 5.5 of [DJP14] and intuitively states that it is hard to determine whether there exists valid "vertical sides" to an SIDH square given the corners and the bottom horizontal side.

3.1 A new hardness assumption

We define a new decisional isogeny assumption which will be useful for the proof of zero-knowledge in Section 5. This assumption can intuitively be seen as a "parallel" version of the DSIDH assumption above, and we shall show here that it is an entirely reasonable cryptographic assumption. Let $\mathsf{CanonicalBasis}(E)$ be a deterministic algorithm taking a curve and outputting a basis P, Q for $E[\ell_2^{e_2}]$.

Definition 6 (Decisional Mirror SIDH (DMSIDH) problem)**.** *Let* $\phi : E_0 \to E_1$ *be an isogeny of degree* $\ell_1^{e_1}$*. Let* P_0, Q_0 *be a basis for the* $\ell_2^{e_2}$*-torsion subgroup* $E_0[\ell_2^{e_2}]$*, let* $\psi : E_0 \to E_2$ *be an isogeny of degree* $\ell_2^{e_2}$*, and let* $\psi' : E_1 \to E_3$ *be the isogeny of degree* $\ell_2^{e_2}$ *whose kernel is* $\phi(\ker \psi)$*. Finally, let* P_2, Q_2 *be a canonical basis of* $E_2[\ell_2^{e_2}]$ *generated as* $\mathsf{CanonicalBasis}(E_2)$*.*

The Decisional Mirror SIDH (DMSIDH) problem is to distinguish between the following two distributions:

- $((E_0, P_0, Q_0), (E_1, \phi(P_0), \phi(Q_0)), \psi, \psi', P_2, Q_2, P_3, Q_3)$ *such that* $P_3 = \phi'(P_2)$ *and* $Q_3 = \phi'(Q_2)$ *for the isogeny* $\phi' : E_2 \to E_3$ *of degree* $\ell_1^{e_1}$ *whose kernel is* $\psi(\ker \phi)$*.*

- $((E_0, P_0, Q_0), (E_1, \phi(P_0), \phi(Q_0)), \psi, \psi', P_2, Q_2, P_3, Q_3)$ *such that* P_3, Q_3 *is a random basis of* $E_3[\ell_2^{e_2}]$ *satisfying the Weil pairing condition*

$$e_{\ell_2^{e_2}}(P_2, Q_2) = e_{\ell_2^{e_2}}(P_3, Q_3)^{\ell_1^{e_1}}$$

and such that both the kernels of the dual isogenies $\widehat{\psi}, \widehat{\psi}'$ *can be expressed in terms of* P_2, Q_2, P_3, Q_3 *with the same coefficients* (c, d)*, namely* $K_{\widehat{\psi}} = [c]P_2 + [d]Q_2$ *and* $K_{\widehat{\psi'}} = [c]P_3 + [d]Q_3$*.*

In other words, $(E_1, \phi(P_0), \phi(Q_0))$ is an SIDH public key, and the ψ, ψ' are the vertical sides of an SIDH square. The challenge is to determine whether points P_3, Q_3 are the actual images of P_2, Q_2 under the hidden horizontal isogeny on the fourth (bottom) side of the SIDH square (which is guaranteed to exist).

In a similar manner to how CSSI reduces to DSIDH, we can define a new computational problem which reduces to the DMSIDH problem.

Definition 7 (Computational Mirror SIDH (CMSIDH) problem)**.** *Let* $\phi : E_0 \to E_1$ *be an isogeny of degree* $\ell_1^{e_1}$*. Let* P_0, Q_0 *be a basis for the* $\ell_2^{e_2}$*-torsion subgroup* $E_0[\ell_2^{e_2}]$*, let* $\psi : E_0 \to E_2$ *be an isogeny of degree* $\ell_2^{e_2}$*, and let* $\psi' : E_1 \to E_3$ *be the isogeny of degree* $\ell_2^{e_2}$ *whose kernel is* $\phi(\ker \psi)$*. Finally, let* P_2, Q_2 *be a random basis of* $E_2[\ell_2^{e_2}]$ *and* $P_3 = \phi'(P_2)$*,* $Q_3 = \phi'(Q_2)$ *for the isogeny* $\phi' : E_2 \to E_3$ *of degree* $\ell_1^{e_1}$ *whose kernel is* $\psi(\ker \phi)$*.*

The Computational Mirror SIDH (CMSIDH) problem is to find ϕ*.*

This computational assumption is entirely reasonable, and is in fact very similar to the CSSI problem. Given a challenge instance of the CSSI problem, one can already choose isogenies ψ, ψ' such that $\ker \psi' = \phi(\ker \psi)$. We can also obtain a point P_2 and its image P_3 under ϕ' via these ψ and ψ'. For example, either $\psi(P_0)$ or $\psi(Q_0)$ will have the correct order, and one can verify that using Vélu's formula [Vél71], $\phi'(\psi(P_0)) = \psi'(P_1)$. Note that naively, this equality will only be up to automorphism on E_3, but it can be verified that Vélu does indeed give us equality. Thus, the only additional information provided in the CMSIDH problem is the image Q_3 of one extra point Q_2 on E_2 (independent of P_2).

We now demonstrate the reduction from CMSIDH to DMSIDH. This reduction follows the one from CSSI to DSIDH very closely.

Suppose $\mathcal{B}$ is a (perfect) distinguisher against the DMSIDH problem, which takes an instance $((E_0, P_0, Q_0), (E_1, \phi(P_0), \phi(Q_0)), \psi, \psi', P_2, Q_2, P_3, Q_3)$. Suppose too, that the exponent of ℓ_1 is a parameter to $\mathcal{B}$ (as in the CSSI to DSIDH reduction by Galbraith and Vercauteren [GV18])—it is possible to remove this supposition as we will discuss later, but it simplifies exposition.

There are $\ell_1 + 1$ different ℓ_1-isogenies having domain E_1, and the same number having domain E_3, so iterate over each of the $(\ell_1 + 1)^2$ combinations $(\varphi : E_1 \to E_1', \varphi' : E_3 \to E_3')$. Let u be such that $u\ell_1 \equiv 1 \pmod{\ell_2^{e_2}}$, and then let:

$$\begin{aligned} P_3' &= [u]\varphi'(P_3) \\ Q_3' &= [u]\varphi'(Q_3) \\ P_1' &= [u]\varphi(\phi(P_0)) \\ Q_1' &= [u]\varphi(\phi(Q_0)) \end{aligned}$$

Finally, let $\overline{\psi'}$ be the isogeny with kernel $[u]\varphi(\ker \psi')$. The following diagram may be helpful in fixing the notation:

$$\begin{tikzcd}
E_0 \arrow[rr, bend left, "\phi"] \arrow[d, "\psi"'] & E_1' \arrow[l, dashed] \arrow[d, "\varphi(\psi')"'] & E_1 \arrow[l, "\varphi"] \arrow[d, "\psi'"] \\
E_2 \arrow[rr, bend right, "\phi'"'] & E_3' \arrow[l, dashed] & E_3 \arrow[l, "\varphi'"']
\end{tikzcd}$$

If E_1' and E_3' are nodes in the isogeny paths ϕ and ϕ' respectively, then φ, φ' (respectively) will be dual to the final ℓ_1-isogeny "step" in each path. The multiplication by u will cancel out the factor of ℓ_1 introduced by composition with the dual. Hence, $((E_0, P_0, Q_0), (E_1', P_1', Q_1'), \psi, \overline{\psi'}, P_2, Q_2, P_3', Q_3')$ will be a DMSIDH tuple for a secret isogeny ϕ of degree $\ell_1^{e_1-1}$. As mentioned above, if the exponent of ℓ_1 can be passed as a parameter to $\mathcal{B}$, then $\mathcal{B}$ can be used immediately to distinguish between the choices of (φ, φ'). This is true because, for any sane choices of SIDH parameters (where $\ell_1^{e_1} \approx \ell_2^{e_2}$), the secret isogeny ϕ is uniquely determined by the points P_1, Q_1 and so $\mathcal{B}$ will only return true on the "correct" step back along ϕ and ϕ'. A short proof of this fact is given by Martindale and Panny [MP19], showing that two different isogenies of the same degree $\ell_1^{e_1}$ can only have the same action on the $\ell_2^{e_2}$ torsion subgroup if $\ell_2^{2e_2} \leq 4\ell_1^{e_1}$. So, while there may exist collisions in general which would complicate this reduction (we would have to track each of the possible paths back from E_1 which $\mathcal{B}$ accepts), in most cases we only have to consider a single, unique path. This step-back process can be iterated until all of ϕ and ϕ' are learned, one degree-ℓ_1 component at a time (along the dashed arrows in the diagram).

Now to resolve our earlier supposition, and consider a distinguisher $\mathcal{B}$ which only operates with a fixed degree parameter $\ell_1^{e_1}$. In order to keep the degree $\ell_1^{e_1}$ in the DMSIDH instance correct when "taking a step back", we also take an arbitrary ℓ_1-isogeny step away from E_0 and E_2, map points P_0, Q_0, P_2, Q_2 and isogeny ψ along the step, and run $\mathcal{B}((E_0', P_0', Q_0'), (E_1', P_1', Q_1'), \overline{\psi}, \overline{\psi'}, P_2', Q_2', P_3', Q_3')$. The isogeny between E_0' and E_1' then becomes $\ell_1^{e_1}$ again, unless it happens that the step from $E_0 \to E_0'$ was also a component in ϕ (or similarly for E_2'). In that case, $\mathcal{B}$ will not accept any of the choices of φ and we have still successfully learned a degree-ℓ_1 component of ϕ.

Thus, assuming the CMSIDH problem is hard, so too is the DMSIDH problem. As SIDH fundamentally assumes that providing the action of a secret isogeny ϕ on a basis for a co-prime torsion subgroup does not leak the secret isogeny itself, so in our setting, doing the same for ϕ' on a totally independent basis should also not leak either ϕ' or ϕ.

4 Previous SIDH identification scheme and soundness issue

4.1 De Feo–Jao–Plût scheme

Let p be a large prime of the form $\ell_1^{e_1} \cdot \ell_2^{e_2} \cdot f \pm 1$, where ℓ_1, ℓ_2 are small primes. We start with a supersingular elliptic curve E_0 defined over $\mathbb{F}_{p^2}$ with $\#E_0(\mathbb{F}_{p^2}) = (\ell_1^{e_1}\ell_2^{e_2}f)^2$. The private key is a random point $K_\phi \in E_0(\mathbb{F}_{p^2})$ of exact order $\ell_1^{e_1}$. Define $E_1 = E_0/\langle K_\phi \rangle$ and denote the corresponding $\ell_1^{e_1}$-isogeny by $\phi : E_0 \to E_1$.

Let P_0, Q_0 be a basis of the torsion subgroup $E_0[\ell_2^{e_2}] = \langle P_0, Q_0 \rangle$. The fixed public parameters are $pp = (p, E_0, P_0, Q_0)$. The public key is $(E_1, \phi(P_0), \phi(Q_0))$. The private key is the kernel generator K_ϕ (equivalently, the isogeny ϕ). The interaction goes as follows:

1. The prover chooses a random primitive $\ell_2^{e_2}$-torsion point K_ψ as $K_\psi = [a]P_0 + [b]Q_0$ for some integers $0 \le a, b < \ell_2^{e_2}$ not both divisible by ℓ_2. Note that $\phi(K_\psi) = [a]\phi(P_0) + [b]\phi(Q_0)$. The prover defines the curves $E_2 = E_0/\langle K_\psi \rangle$ and $E_3 = E_1/\langle \phi(K_\psi)\rangle = E_0/\langle K_\psi, K_\phi\rangle$, and uses Vélu's formulae to compute the following diagram.

$$\begin{tikzcd} E_0 \arrow[r, "\phi"] \arrow[d, "\psi"'] & E_1 \arrow[d, "\psi'"] \\ E_2 \arrow[r, "\phi'"] & E_3 \end{tikzcd}$$

   The prover sends commitment $\mathsf{com} = (E_2, E_3)$ to the verifier.

2. The verifier challenges the prover with a random bit $\mathsf{chall} \leftarrow \{0, 1\}$.

3. If $\mathsf{chall} = 0$, the prover reveals $\mathsf{resp} = (a, b)$ from which K_ψ and $\phi(K_\psi) = K_{\psi'}$ can be reconstructed.

   If $\mathsf{chall} = 1$, the prover reveals $\mathsf{resp} = (\psi(K_\phi) = K_{\phi'})$.

In both cases, the verifier accepts the proof if the points revealed have the correct order and generate kernels of isogenies between the correct curves. We iterate this process t times to reduce the cheating probability (where t is chosen based on the security parameter κ).

Note that in an honest execution of the proof, we have

$$\widehat{\psi'} \circ \phi' \circ \psi = [\ell_2^{e_2}]\phi.$$

4.2 Issue with soundness proofs for the De Feo–Jao–Plût scheme

A core component of the security proof of the De Feo–Jao–Plût identification scheme is the soundness proof. A proof of soundness was given by multiple previous works [DJP14, YAJ$^+$17, GPS20] based on the CSSI problem in Definition 3. A sketch of this soundness proof is as follows:

Suppose $\mathcal{A}$ is an adversary that takes as input the public key and succeeds in the identification protocol (all t iterations) with noticeable probability ϵ. Given a challenge instance $(E_0, E_1, R_2, S_2, \phi(R_2), \phi(S_2))$ for the CSSI problem, we run $\mathcal{A}$ on the tuple $(E_1, \phi(R_2), \phi(S_2))$ as the public key. In the first round, $\mathcal{A}$ outputs commitments $(E_{i,2}, E_{i,3})$ for $1 \le i \le t$. We then send a challenge $b \in \{0, 1\}^t$ to $\mathcal{A}$ and, with probability ϵ, $\mathcal{A}$ outputs a response that satisfies the verification algorithm. Now, we use the standard replay technique: Rewind $\mathcal{A}$ to the point where it had output its commitments and then respond with a different challenge $b' \in \{0, 1\}^t$. With probability ϵ, $\mathcal{A}$ outputs a valid response. This gives exactly the 2-special soundness requirement of two valid transcripts with the same commitment but different challenges.

Now, choose some index i such that $b_i \neq b'_i$. We now restrict our focus to the components (E_2, E_3) for that index, and the two responses. It means $\mathcal{A}$ sent E_2, E_3 and can answer both challenges $b = 0$ and $b = 1$ successfully. Hence $\mathcal{A}$ has provided the maps ψ, ϕ', ψ' in the following diagram.

$$
\begin{tikzcd}
E_0 \arrow[r, dashed, "\phi"] \arrow[r, bend right, "\tilde{\phi}"'] \arrow[d, "\psi"'] & E_1 \arrow[d, "\psi'"] \\
E_2 \arrow[r, "\phi'"] & E_3
\end{tikzcd}
$$

The argument proceeds as follows: We have an explicit description of an isogeny $\tilde{\phi} = \widehat{\psi'} \circ \phi' \circ \psi$ from E_0 to E_1. The degree of $\tilde{\phi}$ is $\ell_1^{e_1} \ell_2^{2e_2}$. One can determine $\ker(\tilde{\phi}) \cap E_0[\ell_1^{e_1}]$ by iteratively testing points in $E_0[\ell_1^j]$ for $j = 1, 2, \ldots$. Hence, one determines the kernel of ϕ, as desired.

However, the important issue with this argument which has so far gone unnoticed, is that it assumes $\ker(\phi) = \ker(\tilde{\phi}) \cap E_0[\ell_1^{e_1}]$. This assumption has no basis, and we will provide a simple counterexample to this argument in the following section. While we always recover an isogeny, it may not be ϕ at all—it is entirely possible the isogeny we recover does not even have codomain E_1 so this proof of 2-special soundness is not valid.

4.3 Counterexample to soundness

Fix a supersingular curve E_0 as above. Generate a random $\ell_2^{e_2}$-torsion point $K_\psi \in E_0(\mathbb{F}_{p^2})$ as $K_\psi = [a]P_2 + [b]Q_2$ for some integers $0 \leq a, b < \ell_2^{e_2}$ not both divisible by ℓ_2. Let $\psi : E_0 \to E_2$ have kernel generated by K_ψ. Then choose a random isogeny $\phi' : E_2 \to E_3$ of degree $\ell_1^{e_1}$ with kernel generated by $K_{\phi'}$. Then choose a random isogeny $\psi' : E_3 \to E_1$ of degree $\ell_2^{e_2}$. Choose points $P_2', Q_2' \in E_1(\mathbb{F}_{p^2})$ such that $\ker \widehat{\psi'} = \langle [a]P_2' + [b]Q_2' \rangle$. Then publish

$$(E_0, E_1, P_2, Q_2, P_2', Q_2')$$

as a public key. In other words, we have

$$E_0 \xrightarrow{\psi} E_2 \xrightarrow{\phi'} E_3 \xrightarrow{\psi'} E_1$$

Now there is no reason to believe that there exists an isogeny from E_0 to E_1 of degree $\ell_1^{e_1}$, yet we can respond to both challenge bits 0 and 1 in a single round of the identification scheme. Pulling back the kernel of ϕ' via ψ to E_0 will result in the kernel of an isogeny which, in general, will not have codomain E_1 (but instead a random other curve). This is because ψ' is entirely unrelated to ψ in this case (they are not "parallel"), so we have no SIDH square.

The key observation is that a verifier could be fooled into accepting this public key by a prover who always uses the same curves (E_2, E_3) instead of randomly chosen ones. When $b = 0$ the prover responds with the pair (a, b) corresponding to the kernel of ψ and $\widehat{\psi'}$, and when $b = 1$ the prover responds with $K_{\phi'}$. The verifier will agree that all responses are correct and will accept the proof.

The reader may immediately have several thoughts:

1. This is not the correct protocol description, since the isogenies ψ and ψ' are supposed to be random. The verifier can check if the same commitments (E_2, E_3) are always being re-used.
2. This scheme would not be zero-knowledge. If the protocol is repeated many times with the same pair (E_2, E_3) then the composition $\psi' \circ \phi' \circ \psi$ will be revealed to the verifier, leaking an isogeny from E_0 to E_1 and therefore allowing the verifier to impersonate the prover in the future.
3. Proving identity (or forging signatures) still requires knowledge of *some* isogeny from E_0 to E_1. So we can rescue the security proof by basing security on the general isogeny problem (Definition 2) instead of the SIDH problem.

4. The SIDH assumption as stated claims that an isogeny from E_0 to E_1 of degree $\ell_1^{e_1}$ exists, and asks to compute it. So surely that prevents the "attack" as well.

In response we say:

1. It is true that the verifier could test if the commitments (E_2, E_3) are being re-used, but this has never been stated as a requirement in any of the protocol descriptions. To tweak the verification protocol we need to know how "random" the pairs (E_2, E_3) (or, more realistically, the pairs (a, b)) need to be.

2. It is true that repeating (E_2, E_3) means the protocol is no longer zero knowledge. But soundness and zero-knowledge are independent security properties that are proved separately (and affect different parties: one gives an assurance to the verifier and the other to the prover). Our counterexample is a counterexample to the soundness proof. The fact that the counterexample is not consistent with the proof that the protocol is zero knowledge is irrelevant.

3-4. It is true that we could instead base security of the protocol on the general isogeny problem. Interestingly, none of the previous authors chose to do it that way. But some applications may require using the identification/signature protocols to prove that an SIDH public key is well-formed. For such applications we need soundness to be rigorously proved. The issue in the security proofs in the literature is not only that it is implicitly assumed that there is an isogeny of degree $\ell_1^{e_1}$ between E_0 and E_1. The key issue is that it is implicitly assumed that the pullback under ψ of $\ker \phi'$ is the kernel of this isogeny. Our counterexample calls these assumptions into question, and shows that the proofs are incorrect as written down.

To make this very clear, consider the soundness proof from De Feo, Jao, and Plût [DJP14]. The following diagram is written within the proof. It implicitly assumes that the horizontal isogeny ϕ' has kernel given by $\psi(S)$, so that the image curve is $E/\langle S, R\rangle$.

$$\begin{tikzcd} E \arrow[d, "\psi"'] & E/\langle S\rangle \arrow[d, "\psi'"] \\ E/\langle R\rangle \arrow[r, "\phi'"'] & E/\langle S, R\rangle \end{tikzcd}$$

This implicit assumption seems to have been repeated in all subsequent works, such as [YAJ$^+$17] and [GPS20].

Note: One may think that the original scheme seems to be secure despite the issue with the proof, as long as the commitment (E_2, E_3) is not reused every time (point 1. above). However, in experiments with small primes, it is entirely possible to construct instances[1] where even with multiple different commitments, a secret isogeny of the correct degree between E_0 and E_1 does not exist. We expect that this extrapolates to large primes too, although one could potentially argue that finding enough such instances is computationally infeasible.

5 New SIDH identification scheme

Let public parameters $pp = (p, E_0, P_0, Q_0)$ such that $E_0(\mathbb{F}_{p^2})[\ell_2^{e_2}] = \langle P_0, Q_0\rangle$. As before, suppose a user has a secret isogeny $\phi : E_0 \to E_1$ with kernel $\ker \phi = K_\phi$. Without loss of generality we assume that the secret isogeny has degree $\ell_1^{e_1}$.

We propose a new sigma protocol to prove knowledge of this isogeny given the public key $(E_1, P_1 = \phi(P_0), Q_1 = \phi(Q_0))$. The protocol is presented in Figure 3. $\mathsf{IsogenyFromKernel}$ is a function taking a

[1]Thank you to Lorenz Panny for demonstrating this.

kernel point and outputting an isogeny and codomain curve with said kernel. $\mathsf{CanonicalBasis}$ is a deterministic function taking a curve and outputting a $\ell_2^{e_2}$-torsion basis on the given curve (as in Section 3). Figure 2 shows the commutative diagram of the sigma protocol.

Intuitively, the identification scheme follows 4.1, with a single bit challenge—if the challenge is 0, we reveal the vertical isogenies ψ, ψ', while if the challenge is 1, we reveal the horizontal ϕ'. The difference is the introduction of additional points on E_3 to the commitment, which force ψ, ψ' to be, in some sense "compatible" or "parallel". This restriction allows the proof of 2-special soundness to work.

We then repeat the identification scheme t times in parallel (where t is chosen based on the security parameter κ) and set com to be the concatenation of all individual $[\mathsf{com}_i]_{i\in[t]}$ for each iteration i, $\mathsf{chall} = [\mathsf{chall}_i]_{i\in[t]}$ and $\mathsf{resp} = [\mathsf{resp}_i]_{i\in[t]}$.

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\phi"] \arrow[d, "\psi"'] & E_1 \arrow[d, "\psi'"] \\
E_2 \arrow[r, "\phi'"'] & E_3
\end{tikzcd}$$

Figure 2: Commutative diagram of SIDH identification scheme

Note: Verification requires checking that there exists integers c, d generating the kernels of dual isogenies $\widehat{\psi}, \widehat{\psi}'$. This computation can be offloaded to the prover by requiring them to send the correct integers. In fact, these integers uniquely determine the vertical isogenies so they could be sent as resp by the prover without needing $K_\psi, K_{\psi'}$, but this would require more computation to verify.

Remark 2. There are certainly improvements that can be made to improve efficiency and compress the size of signatures, but these are standard and we will not explore them here. For example, in practice the commitment information (E_3, P_3, Q_3) would be replaced with a triplet of x-coordinates, as in SIKE [ACC+17].

Theorem 1. *The sigma protocol in Figure 3 for relation*

$$\mathcal{R}_{\mathsf{weakSIDH}} = \{((E_1, P_1, Q_1), \phi) \mid \phi : E_0 \to E_1, \deg\phi = \ell_1^{e_1}\}$$

is correct, 2-special sound, and computationally zero knowledge assuming the DMSIDH and DSSP problems are hard. Repeated with κ iterations, it is thus a Proof of Knowledge for $\mathcal{R}_{\mathsf{weakSIDH}}$ with knowledge error $2^{-\kappa}$.

Proof. We prove the three properties of Theorem 1 separately below.

Correctness: It is clear that following the protocol honestly will result in an accepting transcript.

2-special soundness: Suppose we obtain two accepting transcripts $(\mathsf{com}, \mathsf{chall}, \mathsf{resp})$ and $(\mathsf{com}, \mathsf{chall}', \mathsf{resp}')$ for statement X, with $\mathsf{chall} \neq \mathsf{chall}'$. Consider one of the t rounds i where the challenge bit chall_i differs from chall'_i. The secret isogeny corresponding to the public key (E_1, P_1, Q_1) can be recovered as follows, hence $\mathsf{Extract}$ can extract a valid witness for the statement X such that $(X, W) \in \mathcal{R}_{\mathsf{weakSIDH}}$.

Without loss of generality, suppose $\mathsf{chall}_i = 0$ and $\mathsf{chall}'_i = 1$. Then recover (a, b) and thus $(K_\psi, K_{\psi'})$ from resp_i, and $K_{\phi'}$ from resp'_i. Compute the dual isogeny $\widehat{\psi}$ and use this to pull the kernel $K_{\phi'}$ back to E_0 (this works because the degrees of $K_{\phi'}$ and $\widehat{\psi}$ are co-prime). Let φ be the isogeny with kernel $\langle K_\varphi = \widehat{\psi}(K_{\phi'})\rangle$, so that $\varphi : E_0 \to E_0/\langle K_\varphi\rangle$.

We first demonstrate that $E_0/\langle K_\varphi\rangle \cong E_1$. This follows by considering the diagram of Figure 2 as an SIDH square starting from base curve E_2. We have that $E_1 \cong E_2/\langle K_{\phi'}, G\rangle$ for subgroup G of order $\ell_2^{e_2}$ such

round 1 (commitment)

$a, b \leftarrow \mathbb{Z}/\ell_2^{e_2}\mathbb{Z}$ $\triangleright$ N.B. we can use equivalent keys (Rem. 1) for compactness
$K_\psi = [a]P_0 + [b]Q_0 \in E_0$
$K_{\psi'} = \phi(K_\psi) = [a]\phi(P_0) + [b]\phi(Q_0) \in E_1$
$\psi, E_2 \leftarrow \mathsf{IsogenyFromKernel}(K_\psi)$
$P_2, Q_2 \leftarrow \mathsf{CanonicalBasis}(E_2)$
$K_{\phi'} \leftarrow \psi(K_\phi) \in E_2$
$\phi', E_3 \leftarrow \mathsf{IsogenyFromKernel}(K_{\phi'})$
$P_3, Q_3 \leftarrow \phi'(P_2), \phi'(Q_2) \in E_3$
Prover sends $\mathsf{com} = (E_2, E_3, P_3, Q_3)$ to Verifier.

round 2 (challenge)

Verifier sends $\mathsf{chall} \leftarrow \{0, 1\}$ to Prover.

round 3 (response)

if $\mathsf{chall} = 1$ **then**
    $\mathsf{resp} \leftarrow K_{\phi'}$
else
    $\mathsf{resp} \leftarrow (a, b)$
Prover sends resp to Verifier.

Verification

$(E_2, E_3, P_3, Q_3) \leftarrow \mathsf{com}$
if $\mathsf{chall} = 1$ **then**
    $K_{\phi'} \leftarrow \mathsf{resp}$
    Check $K_{\phi'}$ has order $\ell_1^{e_1}$ and lies on E_2, otherwise output reject
    $P_2, Q_2 \leftarrow \mathsf{CanonicalBasis}(E_2)$
    $\phi', E_3' \leftarrow \mathsf{IsogenyFromKernel}(K_{\phi'})$
    Verify $E_3 = E_3'$ and $P_3, Q_3 = \phi'(P_2), \phi'(Q_2)$, otherwise output reject
    Output accept
else
    $(a, b) \leftarrow \mathsf{resp}$
    Check that $P_1, Q_1 \in E_1$
    $K_\psi, K_{\psi'} = [a]P_i + [b]Q_i$ for $i = 0, 1$ resp.
    Check K_ψ and $K_{\psi'}$ have order $\ell_2^{e_2}$, otherwise output reject
    $\psi, E_2' \leftarrow \mathsf{IsogenyFromKernel}(K_\psi)$
    $\psi', E_3' \leftarrow \mathsf{IsogenyFromKernel}(K_{\psi'})$
    Check $E_2 = E_2'$ and $E_3 = E_3'$
    $P_2, Q_2 \leftarrow \mathsf{CanonicalBasis}(E_2)$
    Check there exists $c, d \in \mathbb{Z}/\ell_2^{e_2}\mathbb{Z}$ such that, simultaneously,
        i $\ker \widehat{\psi} = [c]P_2 + [d]Q_2$
        ii $\ker \widehat{\psi'} = [c]P_3 + [d]Q_3$
    Output accept if the preceding conditions hold, otherwise reject

Figure 3: One iteration of the sigma protocol for our new SIDH identification scheme. The public parameters are $pp = (p, \ell_1, \ell_2, e_1, e_2, E_0, P_0, Q_0)$. The public key is (E_1, P_1, Q_1), and the corresponding secret isogeny is ϕ.

that $\phi'(G) = \ker \widehat{\psi'}$. However, note that the restriction on the kernels of $\widehat{\psi}, \widehat{\psi'}$ force $\ker \widehat{\psi'} = \phi'(\ker \widehat{\psi})$ so $G = K_{\widehat{\psi}}$. Thus, $E_0 \cong E_2/\langle G = K_{\widehat{\psi}}\rangle$ and commutativity implies φ exists and has the correct degree, and

$E_1 \cong E_0/\langle K_\varphi \rangle$ as required. A perhaps simpler argument is that $\hat{\psi}' \circ \phi' \circ \psi$ is an isogeny from E_0 to E_1 that kills the entire $\ell_2^{e_2}$-torsion $E_0[\ell_2^{e_2}]$ so must factor through $[\ell_2^{e_2}]$. Hence there is a degree $\ell_1^{e_1}$ isogeny from E_0 to E_1.

Thus we recover an isogeny φ of correct degree $\ell_1^{e_1}$ such that the codomain is isomorphic to E_1. This shows the protocol is 2-special sound, and that it is a Proof of Knowledge of an isogeny corresponding to the given public key curve (but says nothing about the points in the public key—hence the weakSIDH relation).

Zero-knowledge: Proof of ZK follows as in [DJP14]. Let V^* be a cheating verifier, which shall be used as a black box by the simulator Sim. We show that Sim can generate a valid transcript for t iterations of the protocol. At each step, Sim makes a guess what the next challenge bit chall will be, and then proceeds as follows.

- If $\mathsf{chall} = 0$, Sim simulates as per the honest protocol by choosing $a, b \leftarrow \mathbb{Z}/\ell_2^{e_2}\mathbb{Z}$ and computing the two vertical isogenies $\psi : E_0 \to E_2, \psi' : E_1 \to E_3$ from kernel generators $K_\psi = [a]P_0 + [b]Q_0$ and $K_{\psi'} = [a]P_1 + [b]Q_1$. The simulator then computes the corresponding dual isogenies and the canonical basis $P_2, Q_2 \leftarrow \mathsf{CanonicalBasis}(E_2)$. It writes $K_{\widehat{\psi}}$ in terms of this basis as $[c]P_2 + [d]Q_2$, then chooses a torsion basis on E_3 as $P_3, Q_3 \in E_3$ such that $\langle K_{\widehat{\psi'}} \rangle = \langle [c]P_3 + [d]Q_3 \rangle$, where $e_{\ell_2^{e_2}}(P_3, Q_3) = e_{\ell_2^{e_2}}(P_0, Q_0)^{\ell_1^{e_1}}$ (see Remark 3). Finally, Sim sets the commitment to $\mathsf{com} = (E_2, E_3, P_3, Q_3)$ and the response to $\mathsf{resp} = (a, b)$ as required.
- If $\mathsf{chall} = 1$, the simulator chooses a random supersingular elliptic curve E_2 (by taking a random $\ell_2^{e_2}$-degree isogeny walk from E_0) and a random point $K \in E_2$ of correct order $\ell_1^{e_1}$. Sim then computes the isogeny $\phi' : E_2 \to E_3$ with kernel K using IsogenyFromKernel. Finally, the simulator generates a canonical basis $P_2, Q_2 \leftarrow \mathsf{CanonicalBasis}(E_2)$, computes $P_3, Q_3 \leftarrow \phi'(P_2), \phi'(Q_2)$, and sets the commitment to (E_2, E_3, P_3, Q_3) and the response to K.

After providing com to V^*, if the challenge V^* outputs is not the same as Sim's guess, Sim simply discards that iteration and runs again. Sim stops whenever V^* rejects or after t successful rounds. Suppose the probability of V^* not choosing the same bit as Sim's guess is noticeably different from 1/2. Then V^* can be used as a distinguisher for the DSSP problem (in fact, an even harder problem than the DSSP, as we point out below). So the probability Sim guesses correctly each round is exponentially close to 1/2 if the DSSP problem is hard. Thus Sim will run in polynomial time.

To prove indistinguishability of simulated transcripts from true interactions of a prover P with V^*, it is enough to show that one round of the sigma protocol is indistinguishable (by the hybrid technique of Goldreich et al. [GMW91]).

When $\mathsf{chall} = 0$, the outputs of the simulator are identical to those generated according to the protocol, except for the points P_3, Q_3. However, because the points P_3, Q_3 are chosen by the simulator to pass the Weil pairing check and satisfy the dual kernel condition, by the DMSIDH (Definition 6) assumption, these points chosen by the simulator are indistinguishable from the honest images of the canonical basis. Hence, the distributions are computationally indistinguishable assuming the DMSIDH problem is hard.

To formalize this, suppose $\mathcal{B}_0$ is a PPT adversary which can distinguish between the simulation and the real transcripts for $\mathsf{chall} = 0$ with advantage Adv_0. Given a DMSIDH instance $((E_0, P_0, Q_0), (E_1, \phi(P_0), \phi(Q_0)), \psi, \psi', P_2, Q_2, P_3, Q_3)$, let E_2 be the codomain of ψ and E_3 be the codomain of ψ', and write the kernel of ψ as $\ker \psi = [a]P_0 + [b]Q_0$ for scalars a, b. We can provide $\mathsf{com} = (E_2, E_3, P_3, Q_3), \mathsf{chall} = 0, \mathsf{resp} = (a, b)$ to $\mathcal{B}_0$, and the response from $\mathcal{B}_0$ will solve the DMSIDH problem with the same advantage Adv_0.

When $\mathsf{chall} = 1$, we consider the distribution of (E_2, E_3, ϕ'). While this distribution is not correct a priori, the DSSP computational assumption in Definition 5 implies it is computationally hard to distinguish the simulation from the real game (as in the proof in [GPS20]). Because the action of ϕ' on canonical basis $P_2, Q_2 \in E_2$ can be computed by any party who knows ϕ', the distribution of (E_2, E_3, P_3, Q_3) must also be indistinguishable between simulation and real transcripts.

Suppose $\mathcal{B}_1$ is a PPT adversary which can distinguish between the simulation and the real transcripts for $\mathsf{chall} = 1$ with advantage Adv_1. Given an instance of the DSSP problem, (E_2, E_3, ϕ'), compute $P_2, P_3 \leftarrow \mathsf{CanonicalBasis}(E_2)$. Then let $P_3 = \phi'(P_2)$ and $Q_3 = \phi'(Q_2)$, and set $\mathsf{com} = (E_2, E_3, P_3, Q_3), \mathsf{chall} = 1, \mathsf{resp} = (\ker \phi')$. $\mathcal{B}_1$, given $(\mathsf{com}, \mathsf{chall}, \mathsf{resp})$, will then solve the DSSP with the same advantage Adv_1.

Hence the scheme has computational zero knowledge assuming the DSSP and DMSIDH problems are hard. $\square$

Remark 3 (**Computing a compatible basis**). In the proof of ZK, we require choosing a basis P_3, Q_3 for the torsion subgroup $E_3[\ell_2^{e_2}]$ such that for a fixed kernel $\langle K \rangle < E_3[\ell_2^{e_2}]$ and fixed integers c, d, $\langle K \rangle = \langle [c]P_3 + [d]Q_3 \rangle$. If d is invertible, choose a random point P_3 of order $\ell_2^{e_2}$, and a random invertible scalar r. Then let $Q_3 = [d^{-1}]([r]K - [c]P_3)$ and check that P_3, Q_3 form a basis for $E_3[\ell_2^{e_2}]$ as desired (i.e. they are linearly independent). If not, repeat the random choice of P_3 and r until they do. On the other hand, if d is not invertible, then necessarily c is, so instead choose a random Q_3 and solve for P_3 using the same process. To ensure the Weil pairing check passes, an extra scalar θ can be multiplied by P_3 and Q_3, because as long as θ is co-prime to $\ell_2^{e_2}$, $\langle K \rangle = \langle [\theta]K \rangle = \langle [c]\theta P_3 + [d]\theta Q_3 \rangle$. If $e_{\ell_2^{e_2}}(P_3, Q_3) = e_{\ell_2^{e_2}}(P, Q)^{\mu}$, then θ should be chosen as $\theta^2 = \mu \ell_1^{e_1}$. If $\mu \ell_1^{e_1}$ is not a quadratic residue modulo $\ell_2^{e_2}$, the simulator re-generates the random basis P_3, Q_3 using the above process.

6 Correctness of the points in an SIDH public key

We have shown in Section 5 that successful completion of the new sigma protocol indeed proves knowledge of a degree $\ell_1^{e_1}$ isogeny from E_0 to E_1 (as per the relation $\mathcal{R}_{\mathsf{weakSIDH}}$ in Theorem 1). However, an SIDH public key also consists of the two torsion points, and these points are the cause of issues such as the adaptive attack [GPST16], as discussed in Section 3. In this section, we show that the choice of points P_1, Q_1 by the adversary is severely restricted if they must keep them consistent with "random enough" values of a, b (i.e., random choices of ψ)—preventing adaptive attacks entirely. This gives the following stronger SIDH relation:

$$\mathcal{R}_{\mathsf{SIDH}} = \left\{ ((E_1, P_1, Q_1), \phi) \;\middle|\; \begin{array}{c} \phi : E_0 \to E_1, \deg \phi = \ell_1^{e_1} \wedge \\ P_1 = [\lambda]\phi(P_0) \wedge \\ Q_1 = [\lambda]\phi(Q_0) \\ \lambda \in \pm 1 \end{array} \right\}$$

We have that $\ker \psi = \langle K_\psi \rangle = \langle [a]P_0 + [b]Q_0 \rangle$ for the fixed points P_0, Q_0. This choice of ψ also fixes $\widehat{\psi}$. Now, $\ker \widehat{\psi}' = \phi'(\ker \widehat{\psi})$ as before, so $\widehat{\psi}'$ is also fixed, and by extension ψ'. Finally then, to ensure verification succeeds, the adversary must choose $P', Q' \in E_1$ such that $\langle [a]P' + [b]Q' \rangle = \langle K_{\psi'} \rangle$ for the same a, b as before. For a single choice of a, b, there are many ways to decompose $\ker \psi'$ in terms of two basis points. The key observation though, is that once these points have been fixed in the first iteration of the sigma protocol, all future iterations must use the same two points, but answer with different (a, b) values. If the verifier checks that these (a, b) values are "random enough" whenever they are revealed (challenge bit 0), the prover is restricted in their choice of points as we will see below.

So, as stated above, the prover is in a position where they have a fixed kernel $\langle K_{\psi'} \rangle$. The "honest" behavior will give kernel generator $K_{\psi'} = [a]\phi(P_0) + [b]\phi(Q_0)$. Two generators generate the same kernel if and only if they are (invertible) scalar multiples of each other. Hence, we consider the case where the adversary wishes to decompose any arbitrary kernel generator K' such that $[\lambda]K' = K_{\psi'}$ in terms of a, b, that is, $[a]\phi(P_0) + [b]\phi(Q_0) = [a][\lambda]P' + [b][\lambda]Q'$. For ease of notation, let $P = \phi(P_0), Q = \phi(Q_0)$.

Because both P, Q and P', Q' are bases of the same torsion subgroup, we can represent P', Q' in terms of P, Q with a change-of-basis matrix. This matrix must be invertible, so $cf - de$ must be invertible modulo $\ell_2^{e_2}$.

$$\begin{pmatrix} P' \\ Q' \end{pmatrix} = \begin{pmatrix} c & d \\ e & f \end{pmatrix} \cdot \begin{pmatrix} P \\ Q \end{pmatrix} \tag{1}$$

Now because P and Q are linearly independent, we can match coefficients (modulo the order of the generators) and obtain the following two congruences:

$$\begin{aligned} a &\equiv a\lambda c + b\lambda e \pmod{\ell_2^{e_2}} \\ b &\equiv a\lambda d + b\lambda f \pmod{\ell_2^{e_2}} \end{aligned}$$

Giving:

$$0 \equiv a(\lambda c - 1) + b\lambda e \pmod{\ell_2^{e_2}} \tag{2}$$

$$0 \equiv a\lambda d + b(\lambda f - 1) \pmod{\ell_2^{e_2}} \tag{3}$$

Because P', Q' are published by the prover before beginning the protocol, c, d, e, f are all fixed. We now add the restriction that the verifier confirms the a, b's cover the following three congruency classes modulo ℓ_2 (note that at least one of a, b must not be divisible by ℓ_2 for the kernel to have the correct order):

$$\begin{aligned} a \equiv 0, b \not\equiv 0 &\pmod{\ell_2} \\ a \not\equiv 0, b \equiv 0 &\pmod{\ell_2} \\ a, b \not\equiv 0 &\pmod{\ell_2} \end{aligned}$$

For ease of notation, we will denote these three cases as $(0, \star), (\star, 0)$, and $(\star, \star)$ respectively. It is clearly implied that $\ell_2 \nmid \star$. We will also treat all values modulo $\ell_2^{e_2}$ as integers in the range $0, \ldots, \ell_2^{e_2} - 1$.

If the prover convinces the verifier that with overwhelming probability (in the security parameter κ) they can answer queries using all three classes of a, b above, then it must be the case that $e = d = 0$ and $c = f$ invertible. This indeed proves that the points P', Q' are simply an invertible scalar multiple of the original points $P' = [\lambda]P, Q' = [\lambda]Q$, and is sufficient to prevent adaptive attacks from being performed. In fact, using the Weil pairing check from Galbraith et al. [GPST16] as well, we can force the only choices for this scalar to be $\lambda = \pm 1$ (but we do not need this extra restriction so we will not discuss this further).

To set some notation, we use $\ell^n \parallel x$ to denote that ℓ^n divides x, but ℓ^{n+1} does not divide x. That is, ℓ^n is the highest power of ℓ dividing x. In this case, we say ℓ^n *exactly divides* x.

Theorem 2. *For a fixed security parameter κ and SIDH public key (E, P, Q), if the prover is able to successfully complete 3κ iterations of the identification scheme sigma protocol in Figure 3 as follows:*

(a) κ iterations where the prover uses non-repeating challenges (a, b) for $a, b \not\equiv 0 \pmod{\ell_2}$—case $(\star, \star)$,

(b) κ iterations where the prover uses non-repeating challenges (a, b) for $a \not\equiv 0, b \equiv 0 \pmod{\ell_2}$—case $(\star, 0)$, and

(c) κ iterations where the prover uses non-repeating challenges (a, b) for $a \equiv 0, b \not\equiv 0 \pmod{\ell_2}$—case $(0, \star)$

then with probability $1 - 2^{-\kappa}$ the points P, Q are of the form $[\lambda]\phi(P_0), [\lambda]\phi(Q_0)$ for some invertible scalar λ (where ϕ is a secret $\ell_1^{e_1}$-isogeny $E_0 \to E$).

Proof. We fix c, d, e, f and suppose the prover is able to commit to and successfully answer challenges for (a, b) tuples in all three of the classes above.

If $a \equiv 0, b \not\equiv 0 \pmod{\ell_2}$, then Equation 2 implies that $e \equiv 0 \pmod{\ell_2}$, while Equation 3 requires $f \not\equiv 0 \pmod{\ell_2}$. Similarly, if Equations 2 and 3 are able to be satisfied by a, b where $a \not\equiv 0, b \equiv 0 \pmod{\ell_2}$, we get that $d \equiv 0 \pmod{\ell_2}$ and $c \not\equiv 0 \pmod{\ell_2}$.

In the simplest case, $e = d = 0$. Requiring Equations 2 and 3 to have solutions of the form $(\star, \star)$ (i.e. $a, b \not\equiv 0 \pmod{\ell_2}$) immediately implies that $\lambda c - 1 \equiv \lambda f - 1 \equiv 0 \pmod{\ell_2^{e_2}}$. Hence, $c = f$. This case is the "honest prover" scenario where the points P', Q' the prover provides in the public key are the same as the correct image points $\phi(P_0), \phi(Q_0)$ under the prover's secret isogeny, up to (co-prime) scalar multiple.

It remains to show, then, that being able to satisfy Equations 2 and 3 with (a, b) pairs across all three of the equivalence classes above force $e = d = 0$—that they cannot be non-zero multiples of ℓ_2. We therefore proceed with a proof by contradiction. Let

$$
\begin{aligned}
d &= d'\ell_2^g \\
e &= e'\ell_2^h
\end{aligned}
$$

where g, h are the greatest powers of ℓ_2 dividing d, e (respectively infinite if d or e is 0). Without loss of generality, we can assume that $h \geq g$, because otherwise we can swap the variables $(a, c, e) \leftrightarrow (b, f, d)$. Because we assume that at least one of e, d are non-zero, then this convention implies d (and so too d') is non-zero, while e (and e') may or may not be zero. Note that by definition, $\ell_2 \nmid d'$ and if $e' \neq 0$, then $\ell_2 \nmid e'$.

If (a, b) tuples of the form $(a, b) \equiv (\star, \star) \pmod{\ell_2}$ are able to satisfy Equation 3, then

$$\ell_2^g \parallel 1 - \lambda f$$

By considering Equation 2, we also get that

$$\ell_2^h \mid 1 - \lambda c$$

(if $e \neq 0$ this divisibility is exact, while if $e = 0$, $1 - \lambda c$ must also be 0). Because $g \leq h$, clearly $\ell_2^g \mid 1 - \lambda c$. Then,

$$
\begin{aligned}
1 - \lambda f &\equiv 0 \pmod{\ell_2^g} \\
1 - \lambda c &\equiv 0 \pmod{\ell_2^g} \\
(1 - \lambda f) - (1 - \lambda c) &\equiv 0 \pmod{\ell_2^g} \\
\lambda f &\equiv \lambda c \pmod{\ell_2^g}
\end{aligned}
$$

so we have that $c \equiv f \pmod{\ell_2^g}$.

Now suppose Equations 2 and 3 can be satisfied by $(a, b) \equiv (\star, 0) \pmod{\ell_2}$ as well. Because $\ell_2 \nmid a\lambda d'$, Equation 3 gives:

$$\ell_2^g \parallel b(1 - \lambda' f) \tag{4}$$

We also obtain from Equation 2 that:

$$\ell_2^h \mid 1 - \lambda' c$$

From this, using the fact that $c \equiv f \pmod{\ell_2^g}$ from the $(\star, \star)$ case, and that $g \leq h$, we get

$$
\begin{aligned}
\ell_2^g &\mid 1 - \lambda' c \\
\ell_2^g &\mid 1 - \lambda' c - \lambda(f - c) \\
\ell_2^g &\mid 1 - \lambda' f
\end{aligned}
$$

However, if $\ell_2^g \mid 1 - \lambda' f$ and $\ell_2 \mid b$, then

$$\ell_2^{g+1} \mid b(1 - \lambda' f)$$

Which contradicts Equation 4 by definition of exact divisibility.

Thus, if (a, b) tuples of both forms $(\star, \star)$ and $(\star, 0)$ modulo ℓ_2 are able to satisfy Equations 2 and 3, then necessarily $d = 0$ (and by extension of our assumption $h \geq g$, $e = 0$). To remove the assumption that $h \geq g$, we simply require that tuples of the form $(0, \star)$ are also satisfiable (due to the $(a, c, e) \leftrightarrow (b, f, d)$ variable swap). This concludes the proof. The probability given in the theorem follows trivially from the fact that, as in the original SIDH identification scheme, κ iterations convinces the verifier that the prover can answer each type of case except with probability $2^{-\kappa}$ each time. Hence, we treat each of the three cases as independent proofs and require 3κ iterations overall.

□

6.1 Efficiency

While 3κ is the trivial requirement to ensure the prover can indeed answer all three forms of (a, b) with overwhelming probability, we believe κ-bit security can be achieved with a more efficient choice. However, more thorough analysis is needed. For example, using a biased challenge bit space where $\mathsf{chall} = 0$ with 0.75 probability, we believe 2.4κ iterations would provide soundness error less than $2^{-\kappa}$.

Because 3κ (or perhaps 2.4κ) iterations of the sigma protocol are used rather than κ, this protocol will result in transcripts 3 (or 2.4) times larger than those from Figure 3, when proving the correctness of the points is important.

In terms of the protocol in Figure 3, verification only requires one extra check: In the case that $\mathsf{chall} = 0$ (the **else** clause of the verification algorithm), after extracting (a, b) from resp, the verifier simply keeps track of how many of each case $(\star, \star)$, $(\star, 0)$, and $(0, \star)$ are seen, and accepts overall only if the number of each case is roughly equal. The prover is able to check this requirement is met and repeat the proof generation if it is not, so exactly what "roughly equal" means can be made precise with a tradeoff between assurance and prover efficiency.

Alternatively, it could be enforced that the first 1/3 of iterations must match $(\star, \star)$, the second $(\star, 0)$, and the third $(0, \star)$—at the cost of the verifier knowing the parity of a and b even when $\mathsf{chall} = 1$. Note that in many isogeny schemes / implementations, keys of the form $(1, \alpha)$ are already used exclusively (giving a slightly smaller keyspace, as discussed in the preliminaries on equivalent keys in Section 2.1), so we believe that leaking one further bit of parity of these ephemeral keys would not have a significant impact on the security of the scheme.

The size of our proofs can be further improved using a seed tree, as described in Section 2.3. All commitment values (a, b) could be generated from the leaves of such a seed tree, which would compress the size of responses (as all responses where challenge $\mathsf{chall} = 0$ could be released in a compressed form). If we assume the biased challenge space where $\mathsf{chall} = 0$ with 0.75 probability, this will allow three-quarters of the responses to be compressed. Concretely, for each leaf of the seed tree, a PRG with outputs that are $(2e_2 \log_2 \ell_2)$-bits long can be used to obtain (a, b) directly, and if both a and b are divisible by ℓ_2, the leaf value can be incremented by one repeatedly until they are not both divisible by ℓ_2. This should result in outputs roughly equal in the number of each case $(\star, 0), (0, \star)$, and $(\star, \star)$.

We expect that further improvements to the efficiency and size of the scheme are possible with more analysis, but leave this for future work.

7 SIDH signatures and Non-Interactive Proof of Knowledge

We conclude with some brief, standard remarks about the use of the new protocol proposed above.

It is standard to construct a non-interactive signature scheme from an interactive protocol using the Fiat-Shamir transformation (secure in the (quantum) random oracle model [LZ19]). This works by making the challenge chall for the t rounds of the ID scheme a random-oracle output from input the commitment com and a message M. That is, for message M,

$$V_1^{\mathcal{O}}(\mathsf{com}) = \mathcal{O}(\mathsf{com} \parallel M)$$

Thus the prover does not need to interact with a verifier and can compute a non-interactive transcript. Because the sigma protocol described in the preceding sections not only proves knowledge of the secret isogeny between two curves, but also correctness of the torsion points in the public key, we obtain a signature scheme that is also a proof of knowledge of the secret key corresponding to a given SIDH public key, and proves that the SIDH public key is well-formed. For example, simply signing the public key with its own secret key using the new scheme gives a simple NIZK proof of well-formedness for the public key, which provides protection against adaptive attacks. The unforgeability of such a scheme is additionally based on the CSSI assumption.

References

[ACC+17] Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, David Jao, Brian Koziel, Brian LaMacchia, Patrick Longa, et al. Supersingular isogeny key encapsulation. *Submission to the NIST Post-Quantum Standardization project*, 2017.

[AJL17] Reza Azarderakhsh, David Jao, and Christopher Leonardi. Post-quantum static-static key agreement using multiple protocol instances. In *International Conference on Selected Areas in Cryptography*, pages 45–63. Springer, 2017.

[BKM+20] Andrea Basso, Péter Kutas, Simon-Philipp Merz, Christophe Petit, and Charlotte Weitkämper. On adaptive attacks against jao-urbanik's isogeny-based protocol. In Abderrahmane Nitaj and Amr Youssef, editors, *Progress in Cryptology - AFRICACRYPT 2020*, pages 195–213, Cham, 2020. Springer International Publishing.

[BKP20] Ward Beullens, Shuichi Katsumata, and Federico Pintore. Calamari and falafl: Logarithmic (linkable) ring signatures from isogenies and lattices. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2020*, pages 464–492, Cham, 2020. Springer International Publishing.

[CLN16] Craig Costello, Patrick Longa, and Michael Naehrig. Efficient algorithms for supersingular isogeny diffie-hellman. In *Annual International Cryptology Conference*, pages 572–601. Springer, 2016.

[DGL+20] Samuel Dobson, Steven D. Galbraith, Jason LeGrow, Yan Bo Ti, and Lukas Zobernig. An adaptive attack on 2-SIDH. *International Journal of Computer Mathematics: Computer Systems Theory*, 5(4):282–299, 2020.

[DJP14] Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.

[GMW91] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. *Journal of the ACM (JACM)*, 38(3):690–728, 1991.

[GPS20] Steven D. Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. *Journal of Cryptology*, 33(1):130–175, 2020.

[GPST16] Steven D. Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. In *Advances in Cryptology – ASIACRYPT 2016*, pages 63–91. Springer Berlin Heidelberg, 2016.

[GPV21] Wissam Ghantous, Federico Pintore, and Mattia Veroni. Collisions in supersingular isogeny graphs and the SIDH-based identification protocol. Cryptology ePrint Archive, Report 2021/1051, 2021. `https://eprint.iacr.org/2021/1051`.

[GV18] Steven D. Galbraith and Frederik Vercauteren. Computational problems in supersingular elliptic curve isogenies. *Quantum Information Processing*, 17(10):1–22, 2018.

[HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In *Theory of Cryptography Conference*, pages 341–371. Springer, 2017.

[HL10] Carmit Hazay and Yehuda Lindell. *Sigma Protocols and Efficient Zero-Knowledge*, pages 147–175. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[JD11] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *Post-Quantum Cryptography*, pages 19–34, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[JS14] David Jao and Vladimir Soukharev. Isogeny-based quantum-resistant undeniable signatures. In *PQCrypto 2014*, volume 8772 of *Lecture Notes in Computer Science*, pages 160–179. Springer, 2014.

[LZ19] Qipeng Liu and Mark Zhandry. Revisiting post-quantum Fiat-Shamir. In *Annual International Cryptology Conference*, pages 326–355. Springer, 2019.

[MP19] Chloe Martindale and Lorenz Panny. How to not break SIDH. *CFAIL*, 2019.

[Tho17] Erik Thormarker. *Post-Quantum Cryptography: Supersingular Isogeny Diffie-Hellman Key Exchange*. Thesis, Stockholm University, 2017.

[UJ18] David Urbanik and David Jao. SoK: The problem landscape of SIDH. In *Proceedings of the 5th ACM on ASIA Public-Key Cryptography Workshop*, pages 53–60, 2018.

[UJ20] David Urbanik and David Jao. New techniques for SIDH-based NIKE. *Journal of Mathematical Cryptology*, 14(1):120–128, 2020.

[Vél71] Jacques Vélu. Isogénies entre courbes elliptiques. *C. R. Acad. Sci. Paris Sér. A-B*, 273:A238–A241, 1971.

[YAJ+17] Youngho Yoo, Reza Azarderakhsh, Amir Jalali, David Jao, and Vladimir Soukharev. A post-quantum digital signature scheme based on supersingular isogenies. In *International Conference on Financial Cryptography and Data Security*, pages 163–181. Springer, 2017.