Isogeny-based quantum-resistant undeniable signatures

David Jao[1] and Vladimir Soukharev[2]

[1] Department of Combinatorics and Optimization
[2] David R. Cheriton School of Computer Science
University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada
{djao,vsoukhar}@uwaterloo.ca

Abstract. We propose an undeniable signature scheme based on elliptic curve isogenies, and prove its security under certain reasonable number-theoretic computational assumptions for which no efficient quantum algorithms are known. Our proposal represents only the second known quantum-resistant undeniable signature scheme, and the first such scheme secure under a number-theoretic complexity assumption.

Keywords: undeniable signatures, elliptic curves, isogenies

1 Introduction

Many current cryptographic schemes are based on mathematical problems that are considered difficult with classical computers, but can easily be solved using quantum algorithms. To prepare for the emergence of quantum computers, we aim to design cryptographic primitives for common operations such as encryption and authentication which resist quantum attacks. One family of such primitives, proposed by De Feo, Jao, and Plût [13, 20], uses isogenies between supersingular elliptic curves to construct cryptographic protocols for public-key encryption, key exchange, and entity authentication which are believed to be quantum-resistant. To date, however, this protocol family lacks comprehensive techniques for achieving data authentication, although certain limited capabilities, such as isogeny-based strong designated verifier signatures, are available [30].

In this article, we present a new construction of quantum-resistant undeniable signatures based on the difficulty of computing isogenies between supersingular elliptic curves. Few such constructions are known, and indeed the only other proposed quantum-resistant undeniable signature scheme in the literature is the code-based scheme of Aguilar-Melchor et al. [1]. Our scheme uses a completely different approach and is based on completely different assumptions, making it a useful alternative in the event that some breakthrough arises in the cryptanalysis of code-based systems.

1.1 Related work

Mainstream post-quantum cryptosystems can be categorized into several broad families: lattice-based systems [17, 25] and learning with errors [26], code-based

systems [2, 7, 24], hash-based signatures [6, 11], and systems based on multivariate polynomials [3, 34]. Isogeny-based cryptosystems represent an interesting alternative to the above because they are based on a (relatively) naturally occurring number-theoretic computational problem, namely, the problem of computing isogenies between elliptic curves. These systems thus constitute one of the only families of quantum-resistant cryptosystems based on a number-theoretic assumption (depending on whether one counts solutions to multivariate polynomials as a number-theoretic problem).

Generally speaking, lattice-based systems are more naturally suited to encryption, with lattice-based signature schemes being less mature than the corresponding encryption schemes, whereas hash functions and multivariate polynomials more readily yield signature schemes compared to encryption schemes. Isogeny-based cryptosystems to date have dealt primarily with encryption, with the exception of the entity authentication protocol of [13, §3.1]. We remark that, although entity authentication in the classical setting enables data authentication via the Fiat-Shamir transformation [14], the Fiat-Shamir transformation fails against a quantum adversary [10]. This work, together with Sun et al.'s construction of strong designated verifier signatures [30], provides some evidence that isogenies can also be used as the basis for signatures and data authentication in the post-quantum setting.

We emphasize again that quantum-safe undeniable signatures seem to be difficult to construct by any means. The only known prior quantum-resistant undeniable signature scheme is by Aguilar-Melchor et al. [1], using linear codes.

2 Background

Due to space constraints, we cannot provide here a full treatment of the necessary background information. For further details on the mathematical foundations of isogenies, we refer the reader to [13, 20, 28].

Given two elliptic curves E_1 and E_2 over some finite field $\mathbb{F}_q$ of cardinality q, an *isogeny* ϕ is an algebraic morphism from E_1 to E_2 of the form

$$\phi(x, y) = \left(\frac{f_1(x, y)}{g_1(x, y)}, \frac{f_2(x, y)}{g_2(x, y)} \right),$$

such that $\phi(\infty) = \infty$ (here f_1, f_2, g_1, g_2 are polynomials in two variables, and ∞ denotes the identity element on an elliptic curve). Equivalently, an isogeny is an algebraic morphism which is a group homomorphism. The degree of ϕ, denoted $\deg(\phi)$, is its degree as an algebraic morphism. Two elliptic curves are *isogenous* if there exists an isogeny between them.

Given an isogeny $\phi\colon E_1 \to E_2$ of degree n, there exists another isogeny $\hat{\phi}\colon E_2 \to E_1$ of degree n satisfying $\phi \circ \hat{\phi} = \hat{\phi} \circ \phi = [n]$ (where $[n]$ is the multiplication by n map). It follows that the relation of being isogenous is an equivalence relation. The isogeny $\hat{\phi}$ is called the *dual isogeny* of ϕ. Section 6 (Remark 6.1) describes how to compute dual isogenies in our application.

For any natural number n, we define $E[n]$ to be the subgroup

$$E[n] = \{P \in E(\bar{\mathbb{F}}_q) : nP = \infty\}.$$

In other words, $E[n]$ is the kernel of the multiplication by n map over the algebraic closure $\bar{\mathbb{F}}_q$ of $\mathbb{F}_q$. The group $E[n]$ is isomorphic to $(\mathbb{Z}/n\mathbb{Z})^2$ as a group whenever n and q are relatively prime [28]. We define the *endomorphism ring* $\mathrm{End}(E)$ to be the set of all isogenies from E to itself defined over the algebraic closure $\bar{\mathbb{F}}_q$ of $\mathbb{F}_q$. The endomorphism ring is a ring under the operations of pointwise addition and functional composition. If $\dim_{\mathbb{Z}}(\mathrm{End}(E)) = 2$, then we say that E is *ordinary*; otherwise $\dim_{\mathbb{Z}}(\mathrm{End}(E)) = 4$ and we say that E is *supersingular*. Two isogenous curves are either both ordinary or both supersingular. All elliptic curves used in this work are supersingular.

The isogeny $\phi\colon E_1 \to E_2$ is defined to be *separable* if the function field extension $\mathbb{F}_q(E_1)/\phi^*(\mathbb{F}_q(E_2))$ is separable. In this work, we will only consider separable isogenies. An important property of a separable isogeny is that the size of the kernel of that isogeny is equal to the degree of that isogeny (as an algebraic map) [28, III.4.10(c)]. The kernel K of ϕ uniquely defines the isogeny ϕ up to isomorphism [28, III.4.12]; for this reason, we use the notation E_1/K to denote the codomain E_2 of the isogeny ϕ. Methods for computing and evaluating isogenies are given in [5, 13, 20, 21, 32]. All the isogenies that we use have the property that the kernels are cyclic groups, and knowledge of the kernel, or any single generator of the kernel, allows for efficient evaluation of the isogeny (up to isomorphism); conversely, the ability to evaluate the isogeny via a black box allows for efficient determination of the kernel (cf. Remark 3.1). Thus, in our application, the following are equivalent: knowledge of the isogeny, knowledge of the kernel, or knowledge of any generator of the kernel.

3 Quantum-resistant elliptic curve cryptography

The term "elliptic curve cryptography" typically encompasses cryptographic primitives and protocols whose security is based on the hardness of the discrete logarithm problem on elliptic curves. Against quantum computers, this hardness assumption is invalid [27]. Hence, traditional elliptic curve cryptography is not a viable foundation for constructing quantum-resistant cryptosystems. As a result, alternative elliptic curve cryptosystems based on hardness assumptions other than discrete logarithms have been proposed for use in settings where quantum resistance is desired. One early proposal by Stolbunov [29], based on isogenies between ordinary elliptic curves, was subsequently shown by Childs et al. [8] to offer only subexponential difficulty against quantum computers.[3]

Following these developments, De Feo et al. [13, 20] proposed a new collection of quantum-resistant public-key cryptographic protocols for entity authentication, key exchange, and public-key cryptography, based on the difficulty of

[3] An essentially identical scheme had also been proposed earlier by Couveignes in an unpublished manuscript [9], although not with quantum resistance as a motivation.

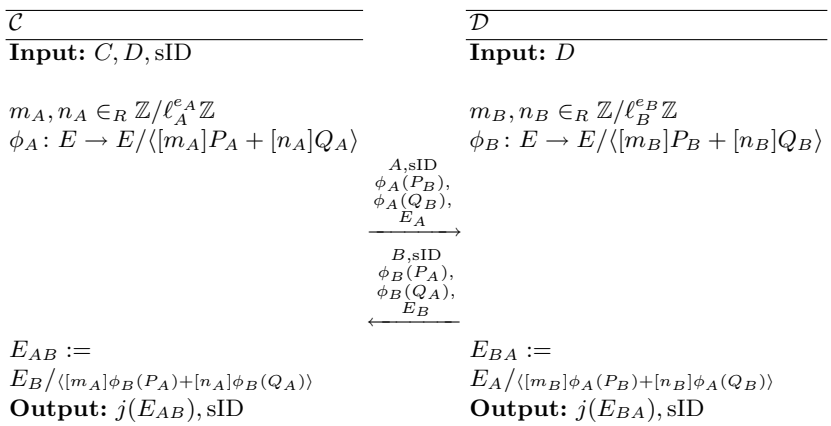

Fig. 1: Key-exchange protocol using isogenies on supersingular curves.

computing isogenies between supersingular elliptic curves. We review here the operation of the most fundamental protocol in the collection, the key exchange protocol, since it contains several critical ideas upon which our undeniable signature scheme is based.

3.1 Parameter generation

Fix a prime p of the form $\ell_A^{e_A}\ell_B^{e_B} \cdot f \pm 1$ where ℓ_A and ℓ_B are small primes, e_A and e_B are positive integers, and f is some (typically very small) cofactor. Also, fix a supersingular curve E defined over $\mathbb{F}_{p^2}$ such that $\#E(\mathbb{F}_{p^2})$ has order divisible by $(\ell_A^{e_A}\ell_B^{e_B})^2$, and bases $\{P_A, Q_A\}$ and $\{P_B, Q_B\}$ which generate $E[\ell_A^{e_A}]$ and $E[\ell_B^{e_B}]$ respectively, so that $\langle P_A, Q_A\rangle = E[\ell_A^{e_A}]$ and $\langle P_B, Q_B\rangle = E[\ell_B^{e_B}]$. Methods for performing these computations are given in [13, Section 4.1].

3.2 Key exchange

Suppose Carol and Dave wish to establish a secret key. Carol chooses two random elements $m_A, n_A \in_R \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$, not both divisible by ℓ_A. The values of m_A and n_A constitute Carol's secret information. (Since Carol and Dave's roles might be reversed in another session, in practice each user requires two sets of values, one for ℓ_A and one for ℓ_B.) On input E and $m_A \cdot P_A + n_A \cdot Q_A$, Carol computes using the method of [13, Section 4.2.2] a curve E_A and an isogeny $\phi_A : E \to E_A$ whose kernel K_A is equal to $\langle [m_A]P_A + [n_A]Q_A\rangle$ (the cyclic subgroup of E generated by $m_A \cdot P_A + n_A \cdot Q_A$). Carol also computes the auxiliary points $\{\phi_A(P_B), \phi_A(Q_B)\} \subset E_A$ obtained by applying her secret isogeny ϕ_A to the basis $\{P_B, Q_B\}$ for $E[\ell_B^{e_B}]$, and sends these points to Dave together with E_A. Similarly, Dave selects random elements $m_B, n_B \in_R \mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$ and computes an isogeny $\phi_B : E \to E_B$ having kernel $K_B := \langle [m_B]P_B + [n_B]Q_B\rangle$, along with the

auxiliary points $\{\phi_B(P_A), \phi_B(Q_A)\}$. Upon receipt of E_B and $\phi_B(P_A), \phi_B(Q_A) \in E_B$ from Dave, Carol computes an isogeny $\phi'_A \colon E_B \to E_{AB}$ having kernel equal to $\langle [m_A]\phi_B(P_A) + [n_A]\phi_B(Q_A)\rangle$; Dave proceeds *mutatis mutandis*. Carol and Dave can then use the common j-invariant of

$$E_{AB} = \phi'_B(\phi_A(E)) = \phi'_A(\phi_B(E)) = E/_{\langle [m_A]P_A + [n_A]Q_A, [m_B]P_B + [n_B]Q_B\rangle}$$

to form a secret shared key.

The full protocol is given in Figure 1. We denote by A and B the identifiers of Carol and Dave, and use sID to denote the unique session identifier.

Remark 3.1. Carol's auxiliary points $\{\phi_A(P_B), \phi_A(Q_B)\}$ allow Dave (or any eavesdropper) to compute Carol's isogeny ϕ_A on any point in $E[\ell_B^{e_B}]$. This ability is necessary in order for the scheme to function, since Dave needs to compute $\phi_A(K_B)$ as part of the scheme. However, Carol must never disclose $\phi_A(P_A)$ or $\phi_A(Q_A)$ (or more generally any information that allows an adversary to evaluate ϕ_A on $E[\ell_A^{e_A}]$), since disclosing this information would allow the adversary to solve a system of discrete logarithms in $E[\ell_A^{e_A}]$ (which are easy since $E[\ell_A^{e_A}]$ has smooth order) to recover K_A.

4 Undeniable signatures from isogenies

In this section, we present a new construction of an undeniable signature scheme from isogenies. An undeniable signature can be verified by any party, but verification requires interaction with the signer. To distinguish between invalid (forged) signatures and valid signatures that the verifier refuses to verify, an undeniable signature scheme also includes a mechanism for the signer to prove (interactively) that an invalid signature is forged. Our construction uses a three-prime variant of the original two-prime protocol given in Section 3.2. As a consequence, the resulting commutative diagrams for zero-knowledge proofs become 3-dimensional rather than 2-dimensional.

4.1 Definition

We were unable to find any prior publications containing a definition and security model for undeniable signatures incorporating quantum computation. For this reason, we make a first attempt at addressing this gap in this section. Our definition of an undeniable signature scheme is the same as that of Kurosawa and Furukawa [22], except for those changes necessary for achieving security in the quantum setting. We caution that our proposed security model is preliminary and may not represent a perfect resolution for this issue.

An undeniable signature scheme [22] consists of a key generation algorithm, a signing algorithm, a validity check, a signature simulator, a confirmation protocol π_{con} and a disavowal protocol π_{dis}. The role of the confirmation protocol π_{con} is for the signer to prove to the verifier that the signature is valid. The role of the disavowal protocol π_{dis} is for a valid signer to be able to prove to the

verifier that the signature that the verifier has received is not valid. Quantum (entangled) information may be transmitted between any two parties which are both capable of quantum computation, or within a single quantum computation, but not between two classical-only parties, or a classical-only party and a quantum-capable party.

In what follows, we make the simplifying assumption that all parties except possibly the adversary are limited to classical computation only; the adversary is permitted to perform quantum computation. This assumption is not part of our security definition; rather, it is merely a simplifying assumption to make our task of analyzing our scheme easier.

Unforgeability is defined using the following game between a challenger and an adversary A.

1. The challenger generates a key pair $(\mathrm{vk}, \mathrm{sk})$ randomly, and gives the verification key vk to A.
2. For $i = 1, 2, \ldots, q_s$ for some q_s, A queries the signing oracle adaptively with a message m_i and receives a signature σ_i.
3. Eventually, A outputs a forgery (m^*, σ^*).

We allow the adversary A to submit pairs (m_j, σ_j) to the confirmation/disavowal oracle adaptively in step 2, where the confirmation/disavowal oracle responds as follows:

- If (m_j, σ_j) is a valid pair, then the oracle returns a bit $\mu = 1$ and proceeds with the execution of the confirmation protocol π_{con} with A.
- Otherwise, the oracle returns a bit $\mu = 0$ and proceeds with the execution of the disavowal protocol π_{dis} with A.

We say that A succeeds in producing a strong forgery if (m^*, σ^*) is valid and (m^*, σ^*) is not among the pairs (m_i, σ_i) generated during the signing queries. The signature scheme is *strongly unforgeable* if the probability that A succeeds in producing a strong forgery is negligible for any PPT adversary A in the above game.

Invisibility is defined using the following game between a challenger and an adversary A.

1. The challenger generates a key pair $(\mathrm{vk}, \mathrm{sk})$ randomly, and gives the verification key vk to A.
2. A is permitted to issue a series of signing queries m_i to the signing oracle adaptively and receive a signature σ_i.
3. At some point, A chooses a message m^* and sends it to the challenger.
4. The challenger chooses a random bit b. If $b = 1$, then he computes the real signature for m^* using sk and sets it to be σ^*. Otherwise he computes a fake signature m^* using vk and sets it to be σ^*. He sends σ^* to A.
5. A performs some signing queries again.
6. At the end of this game, A outputs a guess b'.

We allow the adversary A to submit pairs (m_j, σ_j) to the confirmation/disavowal oracle adaptively in step 2 and in step 5. However, A is not allowed to submit the challenge (m^*, σ^*) to the confirmation/disavowal oracle in step 5. Also, A is not allowed to submit m^* to the signing oracle. We say that the signature scheme is *invisible* if no PPT adversary A has non-negligible advantage in this game.

For an undeniable signature scheme to be secure, it must satisfy unforgeability and invisibility. In addition, the confirmation π_{con} and disavowal π_{dis} protocols must be complete, sound, and zero-knowledge.

4.2 Protocol

Let p be a prime of the form $\ell_A^{e_A}\ell_M^{e_M}\ell_C^{e_C} \cdot f \pm 1$, and fix a supersingular curve E over $\mathbb{F}_{p^2}$ such that $\#E(\mathbb{F}_{p^2})$ is divisible by $(\ell_A^{e_A}\ell_M^{e_M}\ell_C^{e_C})^2$, together with bases $\{P_A, Q_A\}$, $\{P_M, Q_M\}$ and $\{P_C, Q_C\}$ of $E[\ell_A^{e_A}]$, $E[\ell_M^{e_M}]$ and $E[\ell_C^{e_C}]$ respectively. The design of the protocol is such that, generally speaking, points in $\langle P_A, Q_A\rangle$ are used for key material, points in $\langle P_M, Q_M\rangle$ are used for message data, and points in $\langle P_C, Q_C\rangle$ correspond to commitment data.

To generate such primes p, fix a choice of $\ell_A^{e_A}$, $\ell_M^{e_M}$, and $\ell_C^{e_C}$, and test random values of f until a value is found for which $\ell_A^{e_A}\ell_M^{e_M}\ell_C^{e_C} \cdot f \pm 1$ is prime. The prime number theorem in arithmetic progressions (specifically, the effective version of Lagarias and Odlyzko [23]) guarantees that only $O(\log p)$ trials are needed in expectation before a suitable prime is found. For any prime p, Bröker's algorithm for constructing supersingular curves [4] can efficiently produce a supersingular curve E over $\mathbb{F}_{p^2}$ having any admissible cardinality, namely any cardinality of the form $p^2 + 1 - t$ where t satisfies the Hasse-Weil bound $t \leq 2p$ and the supersingularity condition $t \equiv 0 \pmod{p}$. If we take the admissible value $t = \pm 2p$ in Bröker's algorithm, then we obtain a supersingular elliptic curve of cardinality $(p \mp 1)^2 = (\ell_A^{e_A}\ell_M^{e_M}\ell_C^{e_C} \cdot f)^2$, as desired. We remark that in the event E happens to be defined over $\mathbb{F}_p$, the cardinality of E over $\mathbb{F}_{p^2}$ is necessarily $(p+1)^2$.

The signer generates two secret random integers $m_A, n_A \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$, obtains $K_A = [m_A]P_A + [n_A]Q_A$ and computes $E_A = E/\langle K_A\rangle$. Let ϕ_A be an isogeny from E to E_A.

Public parameters: $p, E, \{P_A, Q_A\}, \{P_M, Q_M\}, \{P_C, Q_C\}$, and a hash function $H\colon \{0,1\}^* \to \mathbb{Z}$.
Public key: E_A, $\phi_A(P_C)$, $\phi_A(Q_C)$.
Private key: m_A, n_A.

To sign a message M, we compute the hash $h = H(M)$. Let $K_M = P_M + [h]Q_M$. Then the signer computes the isogenies

- $\phi_M\colon E \to E_M = E/\langle K_M\rangle$
- $\phi_{M,AM}\colon E_M \to E_{AM} = E_M/\langle\phi_M(K_A)\rangle$
- $\phi_{A,AM}\colon E_A \to E_{AM} = E_A/\langle\phi_A(K_M)\rangle$

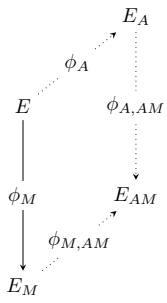

Fig. 2: Signature generation.

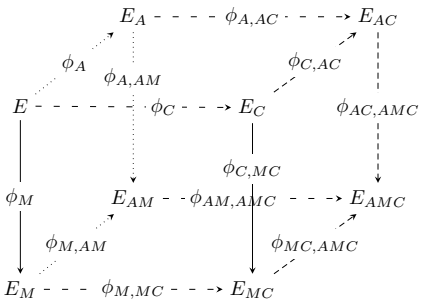

Fig. 3: Confirmation protocol.

along with the auxiliary points $\phi_{M,AM}(\phi_M(P_C))$ and $\phi_{M,AM}(\phi_M(Q_C))$. The signer then presents these two auxiliary points along with E_{AM} as the signature. (See Figure 2.)

The *confirmation protocol* proceeds as follows. We must confirm E_{AM} without revealing the isogenies used to produce it. We do so by "blinding" E_{AM} using ϕ_C and disclosing the blinded isogenies (see Figure 3).

1. The signer secretly selects random integers $m_C, n_C \in \mathbb{Z}/\ell_C^{e_C}\mathbb{Z}$, and computes the point $K_C = [m_C]P_C + [n_C]Q_C$ together with the curves and isogenies in Figure 3. Here $E_C = E/\langle K_C\rangle$, $E_{MC} = E_M/\langle\phi_M(K_C)\rangle = E_C/\langle\phi_C(K_M)\rangle$, $E_{AC} = E_A/\langle\phi_A(K_C)\rangle = E_C/\langle\phi_C(K_A)\rangle$, and $E_{AMC} = E_{MC}/\langle\phi_{C,MC}(K_A)\rangle$.
2. The signer outputs E_C, E_{AC}, E_{MC}, E_{AMC}, and $\ker(\phi_{C,MC})$ as the commitment.
3. The verifier randomly selects $b \in \{0, 1\}$.
4. If $b = 0$, the signer outputs $\ker(\phi_C)$. Using the signer's public key, the verifier computes $\ker(\phi_{A,AC})$. Using knowledge of $\ker(\phi_M)$, the verifier computes $\phi_{M,MC}$. Using the auxiliary points given as part of the signature, the verifier can compute $\phi_{AM,AMC}$. The verifier checks that each isogeny maps between the corresponding two curves specified in the commitment. Using knowledge of $\ker(\phi_C)$, the verifier also independently re-computes $\phi_{C,MC}$ and checks that it matches the commitment.
5. If $b = 1$, the signer outputs $\ker(\phi_{C,AC})$. The verifier computes $\phi_{MC,AMC}$ and $\phi_{AC,AMC}$, and checks that each of $\phi_{C,AC}$, $\phi_{MC,AMC}$, and $\phi_{AC,AMC}$ maps between the corresponding two curves specified in the commitment.

We now describe the *disavowal protocol*. Suppose the signer is presented with a falsified signature (E_F, F_P, F_Q) for a message M, where E_F is the falsified E_{AM}, and $\{F_P, F_Q\}$ are the falsified auxiliary points corresponding to $\phi_{M,AM}(\phi_M(P_C))$ and $\phi_{M,AM}(\phi_M(Q_C))$ respectively. We must disavow E_F without disclosing E_{AM}. To do this, we blind E_{AM} as before to obtain E_{AMC}, and disclose enough information to allow the verifier to compute E_{FC} and check that $E_{FC} \neq E_{AMC}$.

1. The signer secretly selects random integers $m_C, n_C \in \mathbb{Z}/\ell_C^{e_C}\mathbb{Z}$, and computes $K_C = [m_C]P_C + [n_C]Q_C$ along with all the curves and isogenies in Figure 4.

2. The signer outputs E_C, E_{AC}, E_{MC}, E_{AMC}, and $\ker(\phi_{C,MC})$ as the commitment.
3. The verifier randomly selects $b \in \{0, 1\}$.
4. If $b = 0$, the signer outputs $\ker(\phi_C)$. The verifier computes ϕ_C, $\phi_{M,MC}$, $\phi_{A,AC}$, and $\phi_F \colon E_F \to E_{FC} = E_F/\langle [m_C]F_P + [n_C]F_Q \rangle$, and checks that each isogeny maps between the corresponding two curves specified in the commitment. The verifier independently re-computes $\phi_{C,MC}$ and checks that it matches the commitment. The verifier also checks that $E_{FC} \neq E_{AMC}$.
5. If $b = 1$, the signer outputs $\ker(\phi_{C,AC})$. The verifier computes $\phi_{AC,AMC}$ and $\phi_{MC,AMC}$, and checks that these isogenies map to E_{AMC}.

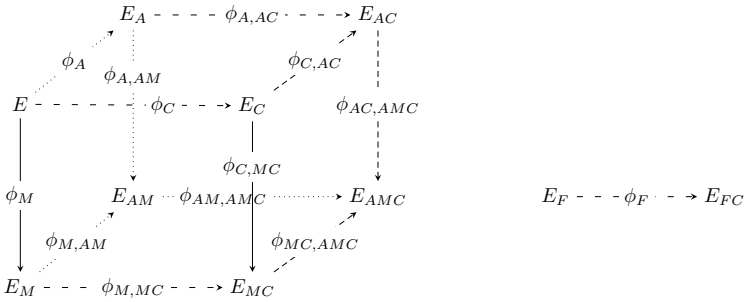

Fig. 4: Disavowal protocol.

5 Complexity assumptions

As before, let p be a prime of the form $\ell_A^{e_A}\ell_B^{e_B}\ell_C^{e_C} \cdot f \pm 1$, and fix a supersingular curve E over $\mathbb{F}_{p^2}$ together with bases $\{P_A, Q_A\}$, $\{P_B, Q_B\}$, and $\{P_C, Q_C\}$ of $E[\ell_A^{e_A}]$, $E[\ell_B^{e_B}]$, and $E[\ell_B^{e_B}]$ respectively. In analogy with [13, 20], we define the following computational problems, which we assume are quantum-infeasible:

Problem 5.1 (Decisional Supersingular Isogeny (DSSI) problem).
Let E_A be another supersingular curve defined over $\mathbb{F}_{p^2}$. Decide whether E_A is $\ell_A^{e_A}$-isogenous to E.

Problem 5.2 (Computational Supersingular Isogeny (CSSI) problem).
Let $\phi_A \colon E \to E_A$ be an isogeny whose kernel is $\langle [m_A]P_A + [n_A]Q_A \rangle$, where m_A and n_A are chosen at random from $\mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ and not both divisible by ℓ_A. Given E_A and the values $\phi_A(P_B)$, $\phi_A(Q_B)$, find a generator R_A of $\langle [m_A]P_A + [n_A]Q_A \rangle$.

We remark that given a generator $R_A = [m_A]P_A + [n_A]Q_A$, it is easy to solve for (m_A, n_A), since E has smooth order and thus extended discrete logarithms are easy in E [31].

Problem 5.3 (Supersingular Computational Diffie-Hellman (SSCDH) problem).
Let $\phi_A \colon E \to E_A$ be an isogeny whose kernel is equal to $\langle [m_A]P_A + [n_A]Q_A \rangle$,

and let $\phi_B \colon E \to E_B$ be an isogeny whose kernel is $\langle [m_B]P_B + [n_B]Q_B \rangle$, where m_A, n_A (respectively m_B, n_B) are chosen at random from $\mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ (respectively $\mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$) and not both divisible by ℓ_A (respectively ℓ_B). Given the curves E_A, E_B and the points $\phi_A(P_B)$, $\phi_A(Q_B)$, $\phi_B(P_A)$, $\phi_B(Q_A)$, find the j-invariant of

$$E/\langle [m_A]P_A + [n_A]Q_A, [m_B]P_B + [n_B]Q_B \rangle.$$

Problem 5.4 (Supersingular Decision Diffie-Hellman (SSDDH) problem). Given a tuple sampled with probability 1/2 from one of the following two distributions:

- $(E_A, E_B, \phi_A(P_B), \phi_A(Q_B), \phi_B(P_A), \phi_B(Q_A), E_{AB})$, where E_A, E_B, $\phi_A(P_B)$, $\phi_A(Q_B)$, $\phi_B(P_A)$, and $\phi_B(Q_A)$ are as in the SSCDH problem and

$$E_{AB} \cong E/\langle [m_A]P_A + [n_A]Q_A, [m_B]P_B + [n_B]Q_B \rangle,$$

- $(E_A, E_B, \phi_A(P_B), \phi_A(Q_B), \phi_B(P_A), \phi_B(Q_A), E_C)$, where E_A, E_B, $\phi_A(P_B)$, $\phi_A(Q_B)$, $\phi_B(P_A)$, and $\phi_B(Q_A)$ are as in the SSCDH problem and

$$E_C \cong E/\langle [m'_A]P_A + [n'_A]Q_A, [m'_B]P_B + [n'_B]Q_B \rangle,$$

  where m'_A, n'_A (respectively m'_B, n'_B) are chosen at random from $\mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$ (respectively $\mathbb{Z}/\ell_B^{e_B}\mathbb{Z}$) and not both divisible by ℓ_A (respectively ℓ_B),

determine from which distribution the tuple is sampled.

Problem 5.5 (Decisional Supersingular Product (DSSP) problem). Given an isogeny $\phi : E \to E_3$ of degree $\ell_A^{e_A}$ and a tuple sampled with probability 1/2 from one of the following two distributions:

- (E_1, E_2, ϕ'), where the product $E_1 \times E_2$ is chosen at random among those $\ell_B^{e_B}$-isogenous to $E \times E_3$, and where $\phi' : E_1 \to E_2$ is an isogeny of degree $\ell_A^{e_A}$, and
- (E_1, E_2, ϕ'), where E_1 is chosen at random among the curves having the same cardinality as E, and $\phi' : E_1 \to E_2$ is a random isogeny of degree $\ell_A^{e_A}$,

determine from which distribution the tuple is sampled.

Our security proofs also make use of the following additional modified assumptions not stated in [13, 20].

Problem 5.6 (Modified Supersingular Computational Diffie-Hellman (MSSCDH) problem). With notation as in the SSDDH problem, given E_A, E_B, and $\ker(\phi_B)$, determine E_{AB}. Note that no auxiliary points for ϕ_A are given.

An equivalent formulation of the MSSCDH problem is: Given E_A, m_B, and n_B, determine E_{AB}.

Problem 5.7 (Modified Supersingular Decision Diffie-Hellman (MSSDDH) problem). With notation as in the SSDDH problem, given E_A, E_B, E_C, and $\ker(\phi_B)$, determine whether $E_C = E_{AB}$. Note that no auxiliary points for ϕ_A are given.

Problem 5.8 (One-sided Modified Supersingular Computational Diffie-Hellman problem (OMSSCDH)). For fixed E_A and E_B, given an oracle to solve MSSCDH for any E_A, $E_{B'}$, $\ker(\phi_{B'})$ where $E_{B'} \ncong E_B$, solve MSSCDH for E_A, E_B, and $\ker(\phi_B)$.

Problem 5.9 (One-sided Modified Supersingular Decision Diffie-Hellman problem (OMSSDDH)). For fixed E_A, E_B, and E_C, given an oracle to solve MSSCDH for any E_A, $E_{B'}$, $\ker(\phi_{B'})$ where $E_{B'} \ncong E_B$, solve MSSDDH for E_A, E_B, E_C, and $\ker(\phi_B)$.

We conjecture that these problems are computationally infeasible, in the sense that for any polynomial-time solver algorithm, the advantage of the algorithm is a negligible function of the security parameter $\log p$. The resulting security assumptions are referred to as the DSSI assumption, CSSI assumption, etc.

We also need a heuristic assumption concerning the distribution of blinded false signatures:

Assumption 5.10 *Fix a supersingular elliptic curve E, an $\ell_A^{e_A}$-isogeny ϕ_A, an $\ell_B^{e_B}$-isogeny ϕ_B, and a curve E_F, not isomorphic to E_{AB}. For any pair of points $\{F_P, F_Q\}$ in E_F, only a negligibly small fraction of integer pairs m_C, n_C satisfy $E_F/\langle m_C F_P + n_C F_Q\rangle = E_{AB}/\langle \phi_{B,AB}(\phi_B(m_C P_C + n_C Q_C))\rangle$.*

5.1 Hardness of the underlying assumptions

All of our unmodified complexity assumptions (those not containing "Modified" in the name) are identical to the corresponding assumptions from [13, 20], except that our assumptions are formulated using primes of the form $p = \ell_A^{e_A}\ell_B^{e_B}\ell_C^{e_C} \cdot f \pm 1$, rather than primes of the form $p = \ell_A^{e_A}\ell_B^{e_B} \cdot f \pm 1$. We have no reason to believe that this alteration would affect the validity of these assumptions. A close analogy to this situation is the comparison between three-prime RSA and two-prime RSA.

Our modified assumptions are needed in order to prove the security of our undeniable signature scheme. The MSSCDH and MSSDDH assumptions are complementary to the SSCDH and SSDDH assumptions, with the main difference being that the input consists of a kernel but not two pairs of auxiliary points (rather than the other way around). The standard algorithm for computing the commuting isogeny from E_B to E_{AB} requires knowing both the values of the kernel of ϕ_B and the auxiliary points for ϕ_A. Similarly, the standard algorithm for computing the commuting isogeny from E_A to E_{AB} requires knowing both the values of the kernel of ϕ_A and the auxiliary points for ϕ_B. In SSCDH (say), the two sets of auxiliary points are known, but the kernels are not known. In MSSCDH, we break the symmetry, giving the attacker the kernel (and hence also the auxiliary points) for ϕ_B, but no secret information about ϕ_A. This kind of asymmetry is unavoidably necessary for any sort of isogeny-based signature scheme, since one isogeny somewhere will invariably be message-based, and this

isogeny can have no secrets. Nevertheless, it is clear that the standard algorithm is not able to solve the modified problems, and we are not aware of any alternative algorithm which would be able to solve the modified problems using only the information given. Indeed, despite extensive study of these problems, we have not managed to devise any plausible approach to these problems other than the claw-finding attack against CSSI originally proposed in [13, Section 5.1]. This attack does not utilize the auxiliary points, and hence works equally well against our modified assumptions, with a running time of $\sqrt[4]{p}$ (respectively $\sqrt[6]{p}$) on a classical (respectively quantum) computer. Other potential strategies discussed in [13, Section 5.1], such as algebraic approaches based on ideal classes in the endomorphism rings, fail in this setting for the same reasons as in [13]. Based on these considerations, we feel that some confidence can be ascribed to the MSSCDH and MSSDDH assumptions. The OMSSCDH and OMSSDDH assumptions are somewhat more artificial, and more study will be needed to justify confidence in them. They arise naturally in the analysis of our undeniable signature scheme.

Our heuristic assumption (Assumption 5.10) seems quite natural, and we have conducted numerous empirical experiments for random choices of triplets (E_F, F_P, F_Q) without finding any violations at cryptographic parameter sizes. For artificially small parameter sizes, our experiments found that for any fixed choice of $(E, \phi_A, \phi_B, E_F, F_P, F_Q)$, equality occurs with probability around $1/N$ over all pairs of integers (m_C, n_C), where $N = \frac{p+1}{12} + O(1)$ is the number of isomorphism classes of supersingular curves in characteristic p. Based on these experiments, we have no reason to suspect that the assumption would fail to hold. However, we have not yet succeeded in finding a proof of the assumption.

6 Security proofs

To prove the security of our scheme, we must show that the confirmation and disavowal protocols are complete, sound and zero-knowledge, and that the overall scheme satisfies the unforgeability and invisibility properties. In this section we consider a **classical** adversary; the case of quantum adversaries will be considered in Section 7.

The basic principle behind the proofs is that, as was the case in the basic key-exchange protocol (Section 3.2), knowledge of (the kernels of) any two opposite-side isogenies lying in a given cube face reveals no information about the other edges in the cube, by the DSSI and DSSP assumptions. On the other hand, knowledge of any two adjacent isogenies in a given commutative square yields full information about all the isogenies in the square. It does not matter which direction the arrows point, since one can reverse the direction of any arrow using dual isogenies (Section 2).

Remark 6.1. To compute the dual isogeny of an isogeny $\phi\colon E \to E_A = E/\langle A \rangle$ whose kernel is generated by a point A, pick any point $B \in E \setminus \langle A \rangle$, and compute $\phi(B)$. Then $\phi(B)$ generates a kernel subgroup whose corresponding isogeny

$\phi' \colon E_A \to E = E_A/\langle\phi(B)\rangle$ is isomorphic to the dual isogeny $\hat{\phi}$. In general, $E_A/\langle\phi(B)\rangle$ is isomorphic but not equal to E, so we also need to compute the appropriate isomorphism, but computing isomorphisms in general is known to be easy [16].

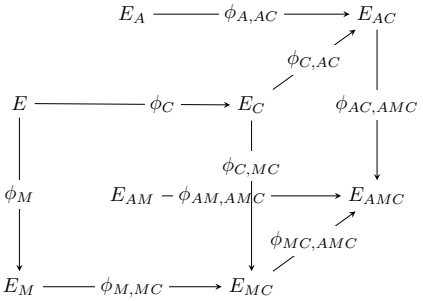

Fig. 5: Proof of soundness (confirmation)

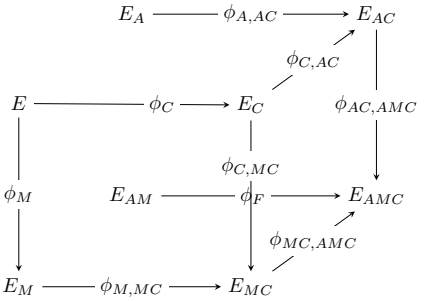

Fig. 6: Proof of soundness (disavowal)

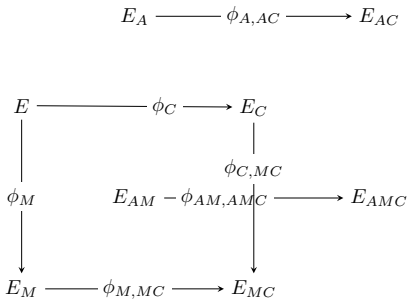

Fig. 7: Confirmation ($b = 0$ case)

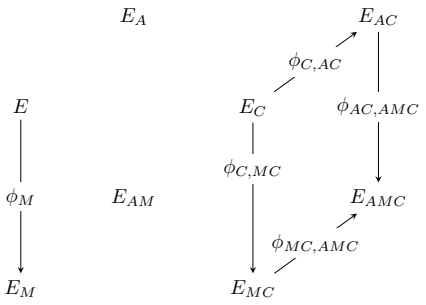

Fig. 8: Confirmation ($b = 1$ case)

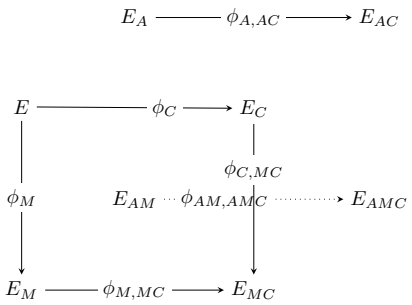

Fig. 9: Disavowal ($b = 0$ case)

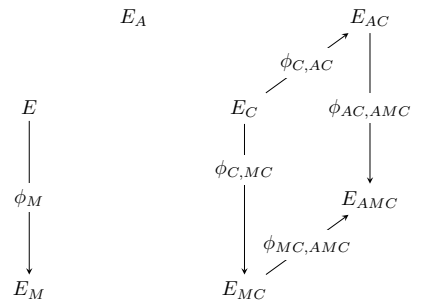

Fig. 10: Disavowal ($b = 1$ case)

6.1 Confirmation protocol

We need to prove three things: *completeness*, *soundness* and *zero-knowledge*. We apply classical techniques from [12, 18].

Proof (Proof of completeness). Completeness for this protocol is obvious. Using the algorithm presented in Section 4.2, the signer can always compute the diagram in Figure 3 and make the verifier accept.

Proof (Proof of soundness). Let Charles be a cheating prover that is able to convince the verifier to accept an invalid signature with non-negligible probability. In order for Charles to be able to provide correct answers to both possible challenges in the confirmation protocol, there must exist a commutative diagram as in Figure 5 with all the edges filled in with actual isogenies. However, the existence of even a single such diagram implies that the signature must actually have been valid, since any three edges of a cube face determine the fourth edge. It follows that isogenies exist between E, E_A, E_M, and E_{AM} to fill in the left face of the cube, rendering the signature valid. Hence soundness holds even against an infinitely powerful malicious prover.

Proof (Proof of zero-knowledge). To prove that this scheme is zero knowledge we construct a simulator. Our simulator S makes uniformly random guesses about what the verifier's challenge will be. Regardless of the guess, S chooses random integers $m_C, n_C \in \mathbb{Z}/\ell_C^{e_C}\mathbb{Z}$ and computes

$$\phi_C \colon E \to E_C = E/\langle m_C P_C + n_C Q_C \rangle.$$

If S guesses $b = 0$, it computes the diagram given in Figure 7. The simulator can now answer any cheating verifier's challenge in the case $b = 0$. The simulator's response is indistinguishable from, and indeed identical to, that of the real prover.

If S guesses $b = 1$, it chooses some random isogeny $\phi_{C,AC} \colon E_C \to E_{AC}$, and computes the diagram given in Figure 8. The simulator uses this diagram to answer the cheating verifier's challenge in the case $b = 1$. In this diagram, the curves E_C and E_{MC} are genuine, and the curves E_{AC} and E_{AMC} are fake. However, the cheating verifier cannot tell that these curves are fake, or else one would be able to solve DSSP for the top face of the cube. Hence the simulator's response is indistinguishable from that of the real prover.

Remark 6.2. The indistinguishability portion of the above proof of the zero-knowledge property holds in the quantum setting as well as in the classical setting. Specifically, if we presume the existence of some quantum cheating verifier (CV) which can perform some quantum computation to distinguish the real transcript from the simulated transcript, then one could use this quantum cheating verifier to obtain a quantum algorithm for solving DSSP simply by alternately supplying the CV with either real curves E_{AC} and E_{AMC} (i.e. the real transcript), or with falsified curves E_{AC} and E_{AMC} (i.e. the simulated transcript), and seeing whether the CV's desired computation performs differently in the two cases.

6.2 Disavowal protocol

As before, we prove *completeness*, *soundness* and *zero-knowledge*.

Proof (Proof of completeness). Suppose first that E_F is not equal to E_{AM}. Using the algorithm presented in Section 4.2, the signer can always compute the diagram in Figure 3 and make the verifier accept. Assumption 5.10 guarantees that the verifier will always accept except with negligible probability. Note that the assumption is formulated without regard to whether the putative auxiliary points F_P and F_Q are compatible with E_F or not.

Now suppose that E_F is equal to E_{AM}. In this case, completeness can only fail if $E_F = E_{AM}$ contains two distinct cyclic subgroups $K_1 = \langle m_C P + n_C Q\rangle$ and $K_2 = \langle \phi_{B,AB}(\phi_B(m_C P_C + n_C Q_C))\rangle$ of cardinality $\ell_C^{e_C}$ in $E_{AB}[\ell_C^{e_C}]$ such that $E_{AM}/K_1 = E_{AM}/K_2$. But then E_{AM} would be a branch point in the covering space of the modular curve $X_0(\ell_C^{e_C})$ over the upper half plane, and the only such non-cusp branch points are the elliptic curves of j-invariant equal to 0 or 1728. The chance of E_{AM} being equal to such a curve is negligibly small. Indeed, there are only two problematic j-invariants, and there are cryptographically many (e.g. 2^{768}) non-problematic j-invariants. A failure probability of 2 in 2^{768} represents no cause for concern, since an adversary could simply guess the private key by brute force with higher success probability. Note that the j-invariant of E_{AM} is determined by a combination of A's public key and the value of the hash $h = H(M)$ of the message M, and this value is never at any point under the control of an adversary. Likewise, the honest user has no control over E_{AM}—its value is completely determined from the user's public key and the message.

Proof (Proof of soundness). Let Charles be a cheating prover that is able to convince the verifier with non-negligible probability that a valid signature is invalid. In order for Charles to be able to provide correct answers to both possible challenges in the confirmation protocol, there must exist a commutative diagram as in Figure 6 with all the edges filled in with actual isogenies. However, in this case, the forged isogeny ϕ_F is computed using exactly the same inputs as the corresponding isogeny $\phi_{AM,AMC}$ for the valid signature in the confirmation protocol, and hence necessarily has codomain E_F equal to E_{AMC}. Equality of E_F and E_{AMC} causes the disavowal protocol to fail. Hence soundness holds even against an infinitely powerful malicious prover.

Proof (Proof of zero-knowledge). To prove that this scheme is zero knowledge we construct a simulator. The simulator S makes uniformly random guesses about what the verifier's challenge will be. The simulator S first chooses random integers $m_C, n_C \in \mathbb{Z}/\ell_C^{e_C}\mathbb{Z}$ and computes

$$\phi_{M,MC} \colon E_M \to E_{MC} = E_M/\langle m_C \phi_M(P_C) + n_C \phi_M(Q_C)\rangle.$$

If S guesses $b = 0$, it computes the diagram given in Figure 9. Here the curves E_C, E_{MC}, and E_{AC} are genuine, and the curves E_{AM} and E_{AMC} are fake. The simulator uses the diagram to answer the cheating verifier's challenge in the case

$b = 0$. The simulator's response is indistinguishable from the real prover, since otherwise one could solve DSSP for the bottom face of the cube.

If S guesses $b = 1$, it chooses some random isogeny $\phi_{C,AC} \colon E_C \to E_{AC}$, and computes the diagram given in Figure 10. The simulator uses this diagram to answer the cheating verifier's challenge in the case $b = 1$. In this diagram, the curves E_C and E_{MC} are genuine, and the curves E_{AC} and E_{AMC} are fake. However, the cheating verifier cannot tell that these curves are fake, or else one would be able to solve DSSP for the top face of the cube. Hence the simulator's response is indistinguishable from that of the real prover.

Remark 6.3. The indistinguishability portion of the above proof of the zero-knowledge property holds in the quantum setting as well as in the classical setting. Specifically, if we presume the existence of some quantum cheating verifier (CV) which can perform some quantum computation to distinguish the real transcript from the simulated transcript, then one could use this quantum cheating verifier to obtain a quantum algorithm for solving DSSP simply by alternately supplying the CV with either real curves E_{AC} and E_{AMC} (i.e. the real transcript), or with falsified curves E_{AC} and E_{AMC} (i.e. the simulated transcript), and seeing whether the CV's desired computation performs differently in the two cases.

6.3 Unforgeability and invisibility

Finally, we prove that the protocol satisfies the unforgeability and invisibility properties from Section 4.1.

Proof (Proof of unforgeability). To prove unforgeability, we must show that after making a polynomial number of queries to a signing oracle, an adversary is still unable to generate a valid signature. Note that we have shown that the confirmation and disavowal protocols are zero-knowledge. Forging signatures is then equivalent to solving OMSSCDH.

Proof (Proof of invisibility). To prove invisibility, we must show that after making a polynomial number of queries to a signing oracle, an adversary will still be unable to decide whether a given signature is valid. This problem is equivalent to OMSSDDH.

7 Quantum-resistant undeniable signatures

Under our simplifying assumption from Section 4.1, all parties except possibly the adversary are restricted to classical computation only. In this setting, all the security proofs in Section 6 other than those for the zero-knowledge proofs hold without modification, since none of these proofs ever at any point involves two quantum parties, and hence we do not need to consider quantum interactions.

By contrast, for zero-knowledge proofs, a classical security proof is not always automatically valid against quantum attacks, since there is the possibility of a

nontrivial quantum interaction: a quantum cheating verifier could conceivably perform some quantum computation on an auxiliary input containing entangled state which is not accessible to the verifier or simulator [33]. Nevertheless, by Hallgren et al. [19], any classical zero-knowledge proof secure against classical honest verifiers can be transformed into a classical zero knowledge proof secure against quantum cheating verifiers at the cost of doubling the number of messages, under the mild condition that the real message transcripts are quantum computationally indistinguishable from the simulated message transcripts. By Remarks 6.2 and 6.3, the real message transcripts are quantum computationally indistinguishable from the simulated message transcripts, for both the confirmation and disavowal protocols, under the assumption that the various computational problems of Section 5 are infeasible on a quantum computer. Therefore the Hallgren et al. transformation can be applied to our confirmation and disavowal protocols to obtain protocols which are zero-knowledge against quantum cheating verifiers. We remark that the prior work of Aguilar-Melchor et al. [1] does not specifically discuss the case of quantum adversaries, and may also require this transformation in order to achieve security against quantum adversaries.

8 Parameter sizes

As stated in [13, 20], the fastest known quantum isogeny finding algorithms in our setting require $O(n^{1/3})$ running time, where n is the size of the kernel. Based on this figure, we obtain the following parameter sizes and signature sizes for various levels of security:

Security level	$\log_2 p$	Signature size
80 bits	720	5760 bits
112 bits	1008	8064 bits
128 bits	1152	9216 bits

These numbers compare favorably with those of the only other prior quantum-resistant undeniable signature scheme, that of Aguilar-Melchor et al. [1]. For example, at the 128-bit security level, the scheme of [1] requires a signature size of 5000 bits for the code-based portion plus an additional "roughly 40k Bytes" [1, p. 116] for the conventional digital signature portion.

Regarding performance, a comparison is difficult because [1] does not provide any performance numbers. For isogeny computations, recent implementation work of De Feo et al. [13, Table 3] and Fishbein [15, Figure 4.1] demonstrates that a single 1024-bit isogeny computation can be performed in 120 ms on a desktop PC, and in under 1 second on an Android device. Our protocol requires three such computations for signing, up to eight for confirmation, and up to nine for disavowal.

9 Conclusion

In this paper we present a quantum-resistant undeniable signature scheme based on the hardness of computing isogenies between supersingular elliptic curves. Our scheme represents the first quantum-resistant undeniable signature scheme based on a number-theoretic computational assumption, and compares well with the only prior undeniable quantum-resistant signature scheme (a code-based scheme) in terms of performance and bandwidth. Future work may entail developing new protocols such as digital signature schemes or more efficient schemes based on weaker assumptions.

10 Acknowledgments

We thank the anonymous referees for providing extensive feedback and assistance in improving our article and our presentation. We also thank Andrew M. Childs, Douglas R. Stinson, Vijay M. Patankar, and Srinath Seshadri for helpful comments and suggestions. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada Collaborative Research and Development Grant CRDPJ 405857-10, and an Ontario Ministry of Research and Innovation Early Researcher Award.

References

1. Carlos Aguilar-Melchor, Slim Bettaieb, Philippe Gaborit, and Julien Schrek. A code-based undeniable signature scheme. In Martijn Stam, editor, *Cryptography and Coding*, volume 8308 of *Lecture Notes in Computer Science*, pages 99–119. Springer Berlin Heidelberg, 2013.
2. Marco Baldi, Marco Bianchi, Franco Chiaraluce, Joachim Rosenthal, and Davide Schipani. Using LDGM codes and sparse syndromes to achieve digital signatures. In Philippe Gaborit, editor, *Post-Quantum Cryptography*, volume 7932 of *Lecture Notes in Computer Science*, pages 1–15. Springer Berlin Heidelberg, 2013.
3. Delphine Boucher, Philippe Gaborit, Willi Geiselmann, Olivier Ruatta, and Felix Ulmer. Key exchange and encryption schemes based on non-commutative skew polynomials. In Nicolas Sendrier, editor, *Post-Quantum Cryptography*, volume 6061 of *Lecture Notes in Computer Science*, pages 126–141. Springer Berlin Heidelberg, 2010.
4. Reinier Bröker. Constructing supersingular elliptic curves. *J. Comb. Number Theory*, 1(3):269–273, 2009.
5. Reinier Bröker, Denis Charles, and Kristin Lauter. Evaluating large degree isogenies and applications to pairing based cryptography. In *Pairing '08: Proceedings of the 2nd International Conference on Pairing-Based Cryptography*, pages 100–112, 2008.
6. Johannes Buchmann, Erik Dahmen, and Andreas Hlsing. Xmss - a practical forward secure signature scheme based on minimal security assumptions. In Bo-Yin Yang, editor, *Post-Quantum Cryptography*, volume 7071 of *Lecture Notes in Computer Science*, pages 117–129. Springer Berlin Heidelberg, 2011.

7. Pierre-Louis Cayrel and Mohammed Meziani. Post-quantum cryptography: Code-based signatures. In Tai-hoon Kim and Hojjat Adeli, editors, *Advances in Computer Science and Information Technology*, volume 6059 of *Lecture Notes in Computer Science*, pages 82–99. Springer Berlin Heidelberg, 2010.
8. Andrew Childs, David Jao, and Vladimir Soukharev. Constructing elliptic curve isogenies in quantum subexponential time. *J. Math. Cryptol.*, 8(1):1–29, 2014.
9. Jean-Marc Couveignes. Hard homogeneous spaces, 2006. `http://eprint.iacr.org/2006/291/`.
10. Özgür Dagdelen, Marc Fischlin, and Tommaso Gagliardoni. The Fiat-Shamir Transformation in a Quantum World. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology - ASIACRYPT 2013*, volume 8270 of *Lecture Notes in Computer Science*, pages 62–81. Springer Berlin Heidelberg, 2013.
11. Erik Dahmen, Katsuyuki Okeya, Tsuyoshi Takagi, and Camille Vuillaume. Digital signatures out of second-preimage resistant hash functions. In Johannes Buchmann and Jintai Ding, editors, *Post-Quantum Cryptography*, volume 5299 of *Lecture Notes in Computer Science*, pages 109–123. Springer Berlin Heidelberg, 2008.
12. Uriel Feige, Amos Fiat, and Adi Shamir. Zero-knowledge proofs of identity. *Journal of Cryptology*, 1(2):77–94, June 1988.
13. Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *J. Math. Cryptol.*, (to appear). `http://eprint.iacr.org/2011/506`.
14. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Advances in Cryptology CRYPTO' 86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194, Berlin, Heidelberg, 1987. Springer Berlin / Heidelberg.
15. Dieter Fishbein. Machine-level software optimization of cryptographic protocols. Master's thesis, University of Waterloo, 2014. `http://hdl.handle.net/10012/8400`.
16. Steven D. Galbraith. Constructing isogenies between elliptic curves over finite fields. *LMS J. Comput. Math.*, 2:118–138 (electronic), 1999.
17. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC '08, pages 197–206, New York, NY, USA, 2008. ACM.
18. 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 Association for Computing Machinery*, 38(3):690–728, July 1991.
19. Sean Hallgren, Alexandra Kolla, Pranab Sen, and Shengyu Zhang. Making classical honest verifier zero knowledge protocols secure against quantum attacks. In Luca Aceto, Ivan Damgrd, LeslieAnn Goldberg, MagnsM. Halldrsson, Anna Inglfsdttir, and Igor Walukiewicz, editors, *Automata, Languages and Programming*, volume 5126 of *Lecture Notes in Computer Science*, pages 592–603. Springer Berlin Heidelberg, 2008.
20. David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Bo-Yin Yang, editor, *PQCrypto*, volume 7071 of *Lecture Notes in Computer Science*, pages 19–34. Springer, 2011.
21. David Jao and Vladimir Soukharev. A subexponential algorithm for evaluating large degree isogenies. In *Algorithmic number theory*, volume 6197 of *Lecture Notes in Computer Science*, pages 219–233. Springer, Berlin, 2010.

22. Kaoru Kurosawa and Jun Furukawa. Universally composable undeniable signature. In *Proceedings of the 35th International Colloquium on Automata, Languages and Programming, Part II*, ICALP '08, pages 524–535, Berlin, Heidelberg, 2008. Springer-Verlag.
23. Jeffrey C. Lagarias and Andrew M. Odlyzko. Effective versions of the Chebotarev density theorem. In *Algebraic number fields: L-functions and Galois properties (Proc. Sympos., Univ. Durham, Durham, 1975)*, pages 409–464. Academic Press, London, 1977.
24. R. J. McEliece. A Public-Key Cryptosystem Based On Algebraic Coding Theory. *Deep Space Network Progress Report*, 44:114–116, January 1978.
25. Daniele Micciancio and Oded Regev. Lattice-based cryptography. In DanielJ. Bernstein, Johannes Buchmann, and Erik Dahmen, editors, *Post-Quantum Cryptography*, pages 147–191. Springer Berlin Heidelberg, 2009.
26. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 84–93, New York, NY, USA, 2005. ACM.
27. Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, 1997. Preliminary version in FOCS '94. `arXiv:quant-ph/9508027v2`.
28. Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1992. Corrected reprint of the 1986 original.
29. Anton Stolbunov. Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves. *Adv. Math. Commun.*, 4(2):215–235, 2010.
30. Xi Sun, Haibo Tian, and Yumin Wang. Toward quantum-resistant strong designated verifier signature from isogenies. In *4th International Conference on Intelligent Networking and Collaborative Systems (INCoS)*, pages 292–296, 2012.
31. Edlyn Teske. The Pohlig-Hellman method generalized for group structure computation. *Journal of Symbolic Computation*, 27(6):521–534, 1999.
32. Jacques Vélu. Isogénies entre courbes elliptiques. *C. R. Acad. Sci. Paris Sér. A-B*, 273:A238–A241, 1971.
33. John Watrous. Zero-knowledge against quantum attacks. In Jon M. Kleinberg, editor, *STOC*, pages 296–305. ACM, 2006.
34. Takanori Yasuda, Tsuyoshi Takagi, and Kouichi Sakurai. Multivariate signature scheme using quadratic forms. In Philippe Gaborit, editor, *Post-Quantum Cryptography*, volume 7932 of *Lecture Notes in Computer Science*, pages 243–258. Springer Berlin Heidelberg, 2013.