A Digital Signature Scheme Based On Supersingular Isogeny Problem

Kisoon Yoon[1], Jihoon Kwon[2], and Suhri Kim[2]

[1] NSHC Inc., South Korea,
kisoon.yoon@gmail.com,
[2] Korea University, South Korea,
htkwon@korea.ac.kr, suhrikim@gmail.com

Abstract. In this paper we propose a digital signature scheme based on supersingular isogeny problem. We design a signature scheme using the Fiat-Shamir transform. The scheme uses a modified version of zero-knowledge proof proposed by De Feo, Jao, and Plût. Unlike the original version our zero-knowledge proof uses only one curve as a commitment. A digital signature scheme using the similar idea was proposed recently by Galbraith et al., but our proposal uses a different method in computing isogeny. We take advantage of our proposed version of zero-knowledge proof to speed up signature generation process. We also present a method of compressing signature.

Keywords: Post-quantum cryptography, information security, elliptic curve, isogeny

1 Introduction

The security of currently used public key cryptosystems is based on number theoretic problems such as hardness of factoring large numbers or solving discrete logarithms over finite fields. However, due to Shor's algorithm, this problems can be solved in polynomial time by quantum adversary, hence threatening the security of current public key cryptosystems. Hence demands for quantum-secure cryptographic primitives are inevitable.

Post-quantum cryptography is an alternative cryptographic primitives that are safe against quantum adversary. Mutivariate-based, code-based, lattice-based, hash-based digital signature, and isogeny-based cryptography are main categories of post-quantum cryptography. Although isogeny-based cryptography is one of the newest in post-quantum cryptography, it is considered prominent candidate due to short key size and its use of elliptic curve arithmetic.

The security of Isogeny-based cryptography is based on hardness of finding isogeny between two given elliptic curves. The first cryptosystem using isogenies between ordinary elliptic curves proposed by A. Stolbunov [17] was extremely inefficient and even suffers from the quantum sub-exponential algorithm proposed by Childs, Jao, and Soukharev [6]. In 2014 De Feo, Jao, and Plût presented a new cryptosystem based on the difficulty of isogeny construction problem between

supersingular elliptic curves which is still infeasible against the known quantum attacks. In 2016, key compression method for supersingular isogeny key exchange was proposed by Reza et al.. They also implemented key exchange protocol in ARM-NEON, FPGA [12,?]. Costello et al. [7] proposed library for supersingular isogeny key exchange and proposed method for faster computation. As stated above, extensive research has been done in isogeny-based cryptography. However, only key-exchange protocol and lack of digital signature scheme was weakness in isogeny-based cryptography.

Recently, Galbraith et al. proposed the first signature scheme based on supersingular isogeny problem [10]. The scheme uses Fiat-Shamir Transform on the zero-knowledge proof (ZKP) proposed by De Feo, Jao, and Plût [8]. They also proposed modified version that uses quaternion ℓ-isogeny algorithm for computing isogenies, which expand the study for isogeny-based cryptosystem.

In this paper, we propose EUF-CMA secure digital signature scheme for isogeny-based cryptography. Our scheme also uses Fiat-Shamir transform on a modified version of the ZKP of [8]. Our proposal is different from that of Galbraith et al. in isogeny computation method.

This paper is organized as follows: In Section 2, we introduce preliminaries for isogeny-based cryptography and current isogeny construction algorithms in Section 3. In Section 4, we describe supersingular isogeny ZKP. We propose our EUF-CMA secure digital signature scheme and propose its security in Section 5. In Section 6, we introduce modification for shorter signature size and conclude our result in Section 7.

2 Elliptic curves and isogenies

Let K be a field. An *elliptic curve* defined over K is an smooth, projective algebraic curve of genus one defined over K with a distinguished point. We know by the Riemann-Roch theorem that such a curve is isomorphic to a curve defined by a Weierstrass equation

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6. \tag{1}$$

It is well known that the points of an elliptic curve form a group with the distinguished point as the identity under the point addition defined by the chord tangent law.

If the characteristic of K is not 2 or 3, then every elliptic curve can be defined by a short Weierstrass equation

$$y^2 = x^3 + ax + b, \tag{2}$$

with the smoothness condition $4a^3 + 27b^2 \neq 0$. Only finite fields with characteristic not equal to 2 or 3 need to be considered for the purpose of supersingular isogeny based cryptography, we assume this case throughout the paper.

The *j-invariant* of the elliptic curve $E/K : y^2 = x^3 + ax + b$ is defined as

$$j(E) = 1728\frac{4a^3}{4a^3 + 27b^2} \in K. \tag{3}$$

We easily know that for a given $j_0 \in K$, there exist an elliptic curve over K having the j-invariant equal to j_0. Two elliptic curves are isomorphic to each other if and only if they have a same j-invariant. So the isomorphism classes of elliptic curves defined over K can be represented as the set of their j-invariants. Hence, in this paper we may refer an elliptic curve E as its j-invariant and vice versa.

Let E and E' be elliptic curves defined over K with the distinguished points O and O', respectively. An isogeny from an E to E' defined over K is a surjective morphism from $E(\overline{K})$ to $E'(\overline{K})$ which maps O to O', where $\overline{K}$ is an algebraic closure of K. Then isogeny automatically becomes a group homomorphism.

The *degree* of an isogeny is defined as the extension degree $[K(E) : \phi^* K(E')]$ of function fields where $\phi^* K(E')$ is the field of rational functions of the form $f \circ \phi$ where $f \in K(E')$. We say the isogeny ϕ is *separable*(resp. *inseparable*) if the extension $K(E)/\phi^*(E')$ is separable (resp. inseparable). Every isogeny $\phi : E \to E'$ can be decomposed as $\phi = \phi_s \circ \pi^n$, where $\phi_s : E \to E'$ is a separable isogeny and π is the Frobenius endomorphism on E. Note that if ϕ is a separable isogeny, then we have $\#\ker\phi = \deg\phi$.

An isogeny over K can be formulated as

$$\phi(x, y) = \left(\frac{g(x)}{h(x)}, \left(\frac{g(x)}{h(x)} \right)' y \right), \tag{4}$$

where g and h are poynomials in $K[x]$, which is called the *standard form* of an isogeny. The degree of isogeny can be computed as $\deg\phi = \max\{\deg g, \deg h\}$. The roots of the polynomial $h(x)$ are exactly the abscissae of the points in the kernel except the point at infinity. Thus the kernel of an isogeny is finite. Conversely, if a finite subgroup G of an elliptic curve E is given, then there exist an elliptic curve $E' \cong E/G$ and a separable isogeny $\phi : E \to E'$, i.e. $\ker\phi = G$.

Vélu [18] gave the explicit formulae to construct an isogeny with a given elliptic curve and a given finite subgroup as the kernel. The formulae are based on the transformation

$$(x_P, y_P) \to \left(x_P + \sum_{Q \in G \setminus \{O\}} (x_{P+Q} - x_Q),\ y_P + \sum_{Q \in G \setminus \{O\}} (y_{P+Q} - y_Q) \right) \tag{5}$$

which is invariant under the translation by Q where Q is in the kernel G. The formulae for higher degree isogeny are complicated, but in the case of low-degree isogeny, they are simple enough and can be efficiently computed [19].

Let $\phi : E_1 \to E_2$ be a separable isogeny of degree ℓ. Then there exist unique separable isogeny $\hat{\phi} : E_2 \to E_1$, with equal degree such that $\hat{\phi} \circ \phi$ is multiplication by ℓ map on E_1. We call $\hat{\phi}$ as dual isogeny of ϕ . Note that $\phi = \hat{\hat{\phi}}$.

We denote the ring of endomorphisms of E by $\mathrm{End}(E)$. Then $\mathrm{End}(E)$ is an order of the endomorphism algebra $\mathrm{End}^0(E) := \mathrm{End}(E) \otimes_{\mathbb{Z}} \mathbb{Q}$. $\mathrm{End}^0(E)$ is isomorphic to one of the rational field, an imaginary quadratic field or a quternian algebra. If $j(E)$ is not an algebraic integer, then $\mathrm{End}^0(E) = \mathbb{Q}$. If $\mathrm{End}^0(E) \ncong \mathbb{Q}$, then we say that E has a complex multiplication.

Let $\mathbb{F}_q$ be a finite field with characteristic $p > 0$ and E be an elliptic curve defined over $\mathbb{F}_q$. Then $\mathrm{End}(E)$ contains the q-power Frobenius automorphism $\pi_q : (x, y) \mapsto (x^q, y^q)$. By Hasse's theorem we have $\#E(\mathbb{F}_q) = q + 1 - t_q$, with $|t_q| \leq 2\sqrt{q}$, where t_q is the trace of the Frobenius.

We denote by $E[n]$ the kernel of the multiplication by n-map. Let K be a field with characteristic p. Then an elliptic curve E over K is supersingular if it satisfies one of the following conditions.

1. $E[p] \cong \{O\}$.
2. The trace of Frobenius of E is divisible by p.
3. $End^0(E)$ is a quaternion algebra.

An elliptic curve which is not supersingular is said to be ordinary.

Since π is a root of the polynomial $X^2 - t_q X + q$, we know that $\pi \in \mathbb{Q}(\sqrt{t^2 - 4q})$. The Hasse condition $|t_q| \leq 2\sqrt{q}$ tells us that every ordinary curve has a complex multiplication. When E is supersingular, the trace of Frobenius is one of $0, \pm\sqrt{q}, \pm 2\sqrt{q}$. The endomorphism ring over $\overline{\mathbb{F}}_q$ of a supersingular curve is the maximal order of a quaternion algebra. In particular, $\mathrm{End}(E) = \mathrm{End}_{\overline{\mathbb{F}}_q}(E)$ if $t_q = \pm 2\sqrt{q}$.

Since every supersingular elliptic curve is isomorphic to an elliptic curve defined over $\mathbb{F}_{p^2}$, we can always take $\mathbb{F}_q = \mathbb{F}_{p^2}$ for a prime number p when E is supersingular. Furthermore, the number of $\overline{\mathbb{F}}_q$-isomorphism classes of supersingular elliptic curve over $\mathbb{F}_{p^2}$ is 1 if $p = 2, 3$ and $\lfloor \frac{p}{12} \rfloor + \epsilon_p$ where $\epsilon_p = 0, 1, 1, 2$ for $p = 1, 5, 7, 11 \pmod{12}$, respectively.

By the Deuring's lifting theorem, every elliptic curve defined over $\mathbb{F}_q$ is a reduction of an elliptic curve defined over a number field L modulo a place in L lying over p. The reduced elliptic curve is supersingular if and only if p does not split in K. Due to Deuring, the CM-method is widely used to find an elliptic curve having a prescribed order for cryptographic use. One can construct supersingular elliptic curves in polynomial time $\tilde{O}(\log q^3)$ using the method proposed by Bröker [5], while the current algorithms for CM-method to find ordinary elliptic curves runs in exponential time in general.

3 Isogeny problems

Galbraith proposed an algorithm for constructing isogenies between ordinary elliptic curves in time $\tilde{O}(p^{\frac{1}{4}})$. The currently fastest known isogeny constructing algorithm between supersingular elliptic curves is in time $\tilde{O}(p^{\frac{1}{2}})$. But Delfs and Galbraith gave a better algorithm in time $\tilde{O}(p^{\frac{1}{4}})$ for the supersingular elliptic curves defined over $\mathbb{F}_p$. In summary, there are currently only exponential time algorithms for isogeny computing between elliptic curves, whether it is supersingular or not.

With the quantum algorithm, Childs, Jao, and Soukharev lowered the complexity of isogeny computation in subexponential time $L_q(1/2, \sqrt{3}/2)$ [6]. The fastest current quantum algorithm for isogeny computation between supersingular elliptic curves has complexity $\tilde{O}(p^{\frac{1}{6}})$ [16,?]. But Biasse, Jao, and Sankar

proposed a quntum subexponential time algorithm in $L_q[1/2, \sqrt{3}/2]$ for constructing isogenies bewteen supersingular elliptic curves defined over $\mathbb{F}_p$.

The supersingular isogeny problem for security of our signature scheme is as follows:

Problem 1. Let be a prime number. Let be E, E' a supersingular elliptic curve over $\mathbb{F}_{p^2}$, chosen uniformly at random. Find an isogeny $\phi : E \to E'$ of given degree.

4 Supersingular isogeny ZKP

An idea of ZKP using supersingular isogeny graph is proposed by De Feo, Jao, and Plût in [8]. In this section we first recall the method of [8], and next introduce our alternative version that enables smaller data representation.

Domain parameters Let ℓ_S, ℓ_R be two small primes and e_S, e_R be positive integers such that $p := \ell_S^{e_S} \ell_R^{e_R} \cdot f \pm 1$ is a prime number, where f is an integral cofactor. Let $K := \mathbb{F}_{p^2}$. Construct a supersingular elliptic curve E defined over K such that $\#E(K) = (\ell_S^{e_S} \ell_R^{e_R} f)^2$. In fact, we can easily generate such a supersingular elliptic curve E over $\mathbb{F}_{p^2}$ using Bröker's method [5].

Since $\ell_S^{e_S}$ divides $\#E(K)$, and also divides $p^2 - 1$, we have $E[\ell_S^{e_S}] \subset E(K)$, so that we can use a non-degenerated bilinear pairing map from $E[\ell_S^{e_S} \ell_R^{e_R}]$ to $\mu_{\ell_S^{e_S} \ell_R^{e_R}} \subset \mathbb{F}_{p^2}^{\times}$ [2]. Note that $E[\ell_\alpha^{e_\alpha}]$ subgroups contatins $\ell_S^{e_S - 1}(\ell_S + 1)$ cyclic subgroups of order $\ell_S^{e_S}$. The analog holds for $\ell_R^{e_R}$. Let P_S, Q_S, P_R and Q_R be points of $E(K)$ such that $\langle P_S, Q_S \rangle = E[\ell_S^{e_S}]$ and $\langle P_R, Q_R \rangle = E[\ell_R^{e_R}]$. Publish $\mathcal{D} := (\ell_S, e_S, \ell_R, e_R, f, p, E, P_S, Q_S, P_R, Q_R)$ as domain parameters.

Private and public parameters Choose integers $m_S, n_S \in \mathbb{Z}/\ell_S^{e_S}\mathbb{Z}$ at random such that $S = m_S P_S + n_S Q_S$ satisfies $\langle S \rangle \cong \mathbb{Z}/\ell_S^{e_S}\mathbb{Z}$. Define a separable isogeny $\phi : E \to E_S \cong E/\langle S \rangle$ with $\ker(\phi) = \langle S \rangle$. Hold the point S as secret information and publish the image curve E_S.

ZKP-I (De Feo, Jao, and Plût) [8] Peggy generates the private parameters as described above. Peggy wants to prove to Victor that she knows ϕ without conveying any information about ϕ. Note that E and E_S are publicly known. Now Peggy and Victor perform the following sigma protocol.

1. Peggy chooses a random point $R \in E[\ell_R^{e_R}]$ such that $\langle R \rangle \cong \mathbb{Z}/\ell_R^{e_R}\mathbb{Z}$ and set $\psi : E \to E_R \cong E/\langle R \rangle$ and $\psi_S : E_S \to E_{SR} \cong E/\langle S, R \rangle$.
2. Peggy sends to Victor (E_R, E_{SR}) as a commitment.
3. Victor sends $b = 0$ or 1 to Peggy as a challenge.
4. If Peggy receives $b = 0$, then Peggy sends $(R, \phi(R))$ to Victor;
   Victor receives (R', R'') from Peggy, computes two isogies ψ', ψ'' with $\ker \psi' = \langle R' \rangle$, $\ker \psi'' = \langle R'' \rangle$ and verifies if $\psi'(E) \cong E_R$, $\psi''(E_S) \cong E_{SR}$.

(4') If Peggy receives $b = 1$, then Peggy sends $\psi(S)$ to Victor;
Victor receives S' from Peggy, computes an isogeny ϕ' with $\ker \phi' = \langle S' \rangle$ and verifies if $\phi'(E_R) \cong E_{SR}$.

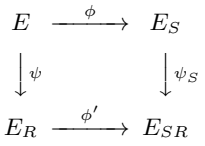

Fig. 1. ZKP-I

The completeness comes from the efficiency of Vélu's formulae and the soundness comes from the hardness of the isogeny problem. The zero-knowledge property is provided since anyone can make a transcript using the following simulation :

1. $b \leftarrow \{0, 1\}$
2. If $b = 0$, then generate two random $\ell_R^{e_R}$-isogenies $\psi' : E \to E'$, $\psi'' : E_S \to E''$ with kernels $\langle R' \rangle \subset E$, $\langle R'' \rangle \subset E_S$ respectively ; output the transcript $((E', E''), 0, (R', R''))$
3. If $b = 1$, then generate a random curve E', E'' such that there exists a random $\ell_S^{e_S}$-isogeny $\phi' : E' \to E''$ with the kernel $\langle S' \rangle$, and output the transcript $((E', E''), 1, S')$.

ZKP-I uses the similar idea as that of the ZKP using graph isomorphism problem [15]. One can construct an identification protocol by repeating the above steps many times with different R for sufficiently small soundness. We also know that the protocol has special soundness that if two transcripts $((E_R, E_{SR}), 0, (R, \phi(R)))$ or $((E_R, E_{SR}), 1, \psi(S))$ are known at the same time, then the secret information S is revealed. Indeed, from $\ker \psi = \langle R \rangle$ and $\psi(S)$, one can compute $\ell_R^{e_R} S = \psi \circ \hat{\psi}(S)$, and immediately obtain $\ker \phi = \langle S \rangle = \langle \ell_R^{e_R} S \rangle$ since $\gcd(\ell_R^{e_R}, \deg \phi) = 1$.

ZKP-II, our proposed version In ZKP-I, a couple of elliptic curves E_R and E_{SR} need to be sent as a commitment, and we have to compute two isogenies ψ and ψ_s from the two kernels $\langle R \rangle$ and $\langle \phi(R) \rangle$, respectively. We propose here another version of ZKP using only one curve as a commitment and computing only one isogeny at every choice of challenge b. We propose the following ZKP primitive :

1. Peggy chooses point $R \in E[\ell_R^{e_R}]$ at random such that $\langle R \rangle \cong \mathbb{Z}/\ell_R^{e_R}\mathbb{Z}$ and set the isogenies $\beta : E_S \to E_R \cong E_S/\langle \phi(R) \rangle$ and $\alpha = \beta \circ \phi : E \to E_R$.
2. Peggy sends to Victor E_R as a commitment.

3. Victor sends $b = 0$ or 1 to Peggy as a challenge.
4. If Peggy receives $b = 0$ from Victor, then she sends $G = S + R$ to Victor;
   Victor receives G' from Peggy, computes an isogeny α' with $\ker \alpha' = \langle G' \rangle$ and verifies if $\alpha'(E) \cong E_R$.

(4′) If Peggy receives $b = 1$ from Victor, then she sends $\phi(R)$ to Victor;
   Victor receives G' from Peggy, computes an isogeny β' with $\ker \beta' = \langle G' \rangle$ and verifies if $\beta'(E_S) \cong E_R$.

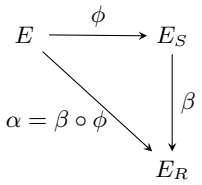

Fig. 2. ZKP-II

The completeness and the soundness of ZKP-II can be showed in the similar manner as ZKP-I. The special soundness is provided as follows.

Lemma 1. *In ZKP-II, if two transcripts $(E_R, 0, \alpha)$ and $(E_R, 1, \beta)$ are obtained at the same time, then one can compute the isogeny $\phi : E \to E_S$ in probabilistic polynomial time.*

Proof. Note that that $E[\ell_S^{e_S} \ell_R^{e_R}] \cong (\mathbb{Z}/\ell_S^{e_S}\mathbb{Z})^2 \oplus (\mathbb{Z}/\ell_R^{e_R}\mathbb{Z})^2$. Let $\overline{S}, \overline{R} \in E(\mathbb{F}_{p^2})$ be points such that $E[\ell_S^{e_S}] = \langle S, \overline{S} \rangle$ and $E[\ell_R^{e_R}] = \langle R, \overline{R} \rangle$. Then every point $T \in E[\ell_S^{e_S} \ell_R^{e_R}]$ can be represented as $T = aS + b\overline{S} + cR + d\overline{R}$ for some integers a, b, c, d. Further, we suppose $\gcd(b, \ell_S) = 1$, which is highly probable when T is chosen at random. We know that two isogenies α and $\beta \circ \phi$ are equivalent from the fact that their kernels are the same subgroup $\langle S + R \rangle \subset E(\mathbb{F}_{p^2})$. The dual isogeny $\hat{\beta}$ is easily constructed from the kernel $\langle \beta(R') \rangle$ where $R' \in E_S[\ell_R^{e_R}] \setminus \langle \phi(R) \rangle$ and $\langle R' \rangle \cong \mathbb{Z}/\ell_R^{e_R}\mathbb{Z}$. So we can compute the image $\hat{\beta} \circ \alpha(T) = \hat{\beta} \circ \alpha(b\overline{S} + d\overline{R}) = \hat{\beta} \circ \beta \circ \phi(b\overline{S} + d\overline{R}) = [\ell_R^{e_R}]\phi(b\overline{S}) \in E_S(\mathbb{F}_{p^2})$. Since we have assumed $\gcd(b, \ell_S) = 1$, the group $\langle [\ell_R^{e_R}]\phi(b\overline{S}) \rangle = \langle \phi(\overline{S}) \rangle$ is the kernel of the dual isogeny $\hat{\phi}$. Now we can compute ϕ easily since $\phi = \hat{\hat{\phi}}$. The proposition holds since every step in the calculation can be done in probabilistic polynomial time. □

Computing isogenies Now we have problems of how to compute isogenies practically. Since the degrees of isogenies that appear in our ZKPs are smooth, the computations can be done effectively by repeating the computation of small degree isogenies. One can basically use the following algorithm for computing an ℓ^e-isogeny.

Computing isogeny ψ with degree ℓ^e
INPUT : An elliptic curve $E/\mathbb{F}_q$, a point $G \in E(\mathbb{F}_q)$ of order ℓ^e.
OUTPUT : The image curve $\psi(E)$ where $\ker\psi = \langle G \rangle$.

1. $G_0 \leftarrow G$.
2. For $i = 0, \cdots, e-1$ do the folllowing
   (a) Find cyclic ℓ-isogeny ψ_i with $\ker\psi_i = \langle \ell^{e-i-1} G_i \rangle$.
   (b) Compute $E_{i+1} = \psi_i(E_i)$, $G_{i+1} = \psi_i(G_i)$.
3. OUTPUT E_e.

The algorithm can be improved by using optimal strategy method proposed in [8]. Some results of the efficient implementation of the method was presented in [7]. One can compute isogenies ϕ and β as above. Victor's computation of α needs applying above algorithm two times with the initial point $G = S + R$, i.e. Victor computes at first $\ell_S^{e_S}$-isogeny from E, obtain E_{e_S}, G_{e_S} as the intermediate results and applies $\ell_R^{e_R}$-isogeny from E_{e_S} with the kernel $\langle G_{e_S} \rangle$ to eventually obtain E_R.

Hence, Victor using ZKP-II has no benefit in computational efficiency compared with ZKP-I. However note that the data size of commitment is $2/3$ times smaller than that of ZKP-I at average. This improvement is important since one has to use hundreds of ZKP's when constructing practical cryptographic schemes like an identification or a digital signature.

Remark *Galbraith et al. [10] have proposed a ZKP primitive using isogeny graph of similar structure. However in their main contribution (the second signature scheme), they use the powersmooth version of the quaternion ℓ-isogeny algorithm of Kohel et al. [11] to compute isogenies between elliptic curves E, E_S and E_R where the endomorphism rings* $\mathrm{End}(E)$, $\mathrm{End}(E_S)$ *and* $\mathrm{End}(E_R)$ *are known to signer. Our version uses Vélu's formulae with cyclic kernel generator for isogeny computation.*

5 A supersingular isogeny digital signature algorithm using Fiat-Shamir transform

Fiat-Shamir transform turns a sigma protocol into a digital signature scheme using random oracles [9]. We construct a Fiat-Shamir type digital signature algorithm based on ZKP-II described in Section 4. We continue to use the same domain parameters $\mathcal{D} := (\ell_S, e_S, \ell_R, e_R, f, p, E, P_S, Q_S, P_R, Q_R)$ as defined in Section 4.

Let $\mathcal{M} = \{0,1\}^*$ be the message space and $H : \mathcal{M} \to \{0,1\}^n$ be a cryptographic hash function where n is a security parameter.

Key-pair generation
INPUT : the domain parameter $\mathcal{D}$
OUTPUT : a point $S \in E$ as a private key, an elliptic curve E_S as a public key
We use the same notation for the private and public parameters defined in Section 4. Output the generating point S of the kernel of an isogeny $\phi : E \to E_S$ as a private key and the image curve E_S as the corresponding public key.

Signature generation
INPUT : a domain parameter $\mathcal{D}$, a message $m \in \mathcal{M}$, a private key S.
OUTPUT : the digital signature (e, s) for m
Perform the following steps :

1. Choose random points $R_1, \cdots, R_n \in E[\ell_R^{e_R}]$ such that $\langle R_i \rangle \cong \mathbb{Z}/\ell_R^{e_R}\mathbb{Z}$;
   Put $G_i = S + R_i$; Compute $\phi(R_i)$;
   Compute isogenies $\beta_i : E \to E_i$ with $\ker \beta_i = \langle \phi(R_i) \rangle$.
2. Put $r = (E_1, \cdots, E_n)$.
3. Compute $e = H(r||m)$.
4. Put $(b_1, \cdots, b_n) = e$ with $b_i = 0$ or 1 ;
   Put $s = (K_1, \cdots, K_n)$ where $K_i = \begin{cases} G_i & \text{if } b_i = 0, \\ \phi(R_i) & \text{if } b_i = 1. \end{cases}$
5. Output (e, s).

Signature verification
INPUT : a domain parameter $\mathcal{D}$, a message $m \in \mathcal{M}$, a public key E_s, a digital signature (e', s') where $s' = (K'_1, \cdots, K'_n)$.
OUTPUT : status (TRUE or FALSE)
Perform the following steps :

1. Put $(b'_1, \cdots, b'_n) = e'$ with $b'_i = 0$ or 1.
   For $i = 1, \cdots, n$, do the following steps :
   If $b'_i = 0$, then compute isogeny $\alpha_i : E \to E'_i$ with $\ker \alpha_i = \langle K'_i \rangle$.
   If $b'_i = 1$, then compute isogeny $\beta_i : E_S \to E'_i$ with $\ker \beta_i = \langle K'_i \rangle$.
2. Put $r = (E'_1, \cdots, E'_n)$.
3. If $e' = H(r||m)$, then output TRUE
   else output FALSE.

6 Security

We now prove that the proposed signature scheme is existentially unforgeable under adaptive chosen-message attacks (EUF-CMA-secure) in the random oracle model using forking lemma. We briefly recall the forking lemma.

Lemma 2 (Forking Lemma [3]). *Fix an integer $q \geq 1$ and a set $\mathcal{H}$ of size $h \geq 2$. Let A be a randomized algorithm that on input $x, h_1, \cdots, h_q$ returns a pair, the first element of which is an integer in the range $0, \cdots, q$ and the second element of which we refer to as a side output. Let IG be a randomized algorithm that we call the input generator. The accepting probability of A, denoted acc, is defined as the probability that $I \geq 1$ in the experiment*

$$x \xleftarrow{\$} IG \ ; \ h_1, \cdots, h_q \xleftarrow{\$} \mathcal{H} \ ; \ (I, \sigma) \xleftarrow{\$} A(x, h_1, \cdots, h_q).$$

The forking algorithm F_A associated to A is the randomized algorithm that takes input x proceeds as follows:

Algorithm $F_A(x)$

Pick coins ρ for A at random
$h_1, \cdots, h_q \xleftarrow{\$} \mathcal{H}$
$(I, \sigma) \leftarrow A(x, h_1, \cdots, h_q; \rho)$
If $I = 0$ *then return* $(0, null, null)$
$h'_I, \cdots, h'_q \xleftarrow{\$} \mathcal{H}$
$(I', \sigma') \leftarrow A(x, h_1, \cdots, h_{I-1}, h'_I, \cdots, h'_q; \rho)$
If $(I = I'$ *and* $h_I \neq h'_I)$ *then return* $(1, \sigma, \sigma')$
Else return $(0, null, null)$.

$$\textit{Then } Pr\left[b = 1 : x \xleftarrow{\$} IG; (b, \sigma, \sigma') \xleftarrow{\$} F_A(x)\right] \geq acc \cdot \left(\frac{acc}{q} - \frac{1}{h}\right).$$

The forking lemma means that if an algorithm A can produce a value σ at an index I of the sequence $h_1, \cdots, h_q$ with a non-negligible probability, then the algorithm F_A, which uses A, can obtain another value σ' different from σ at the same index I of a modified sequence $h_1, \cdots, h_{I-1}, h'_I, \cdots, h'_q$ with a non-negligible property. Note that the sequences are different after the index $I - 1$ in the second run. Refer to [3] for details.

We apply Lemma 2 letting $I = r$ and $\sigma = s$ in our scheme. That is to say, if an adversary A can produce an existential forgery (r, e, s) on a message m with a non-negligible probability, then the adversary can obtain another forgery (r, e', s') on the same message with a non-negligible probability by performing the process again with a different sequence of hash values.

Theorem 1. *In the random oracle model let A denote a EUF-CMA adversary against proposed signature scheme with adventage ϵ, making q queries to its hash function h. Then there is an adversary T_A against the isogeny problem with adventage ϵ' such that*

$$\epsilon' \geq \frac{\epsilon^2}{q} - \frac{\epsilon}{h}$$

Proof. We take the adversary A which uses E_s as public key and wrap it inside another algorithm A' which does not make queries to signature oracle. The algorithm A' excutes the following steps for hash queries $h_1, \cdots, h_q$

1. $C = (C_1, \cdots, C_n) \xleftarrow{\$} \{E, E_s\}^n$.
2. $s \leftarrow (K_1, \cdots, K_n)$ where each K_i is randomly chosen point of C_i such that $\langle K_i \rangle = \ker \gamma_i \cong \mathbb{Z}/\ell_R^{e_R}\mathbb{Z}$ for an isogeny $\gamma_i : C_i \to \gamma(C_i)$.
3. $r \leftarrow (\gamma_1(C_1), \cdots, \gamma_n(C_n))$.
4. Define $H(r||m) := h_i$. If this value has already been defined, go to (1) and choose another C.

Now we apply the forking lemma to $T_{A'}$ to obtain a pair of tuples (r, e, s) and (r, e', s'). Then there exist at least one bit of e' different from that of e at some position, which reveals the secret isogeny $\phi : E \to E_s$ by Lemma 1. □

7 Size of parameters and signatures

In this section we discuss on size of domain parameters and signatures, and propose an efficient method for compressing signature.

Size of parameters The domain parameters $\mathcal{D} := (\ell_S, e_S, \ell_R, e_R, f, p, E, P_S, Q_S, P_R, Q_R)$ have the bit-size about $10 \log_2 p$, which is the sum of the size of the j-invariant of initial elliptic curve and the sum of the sizes of the x-coordinates of four generating points. Note that E is defined over $\mathbb{F}_p$ so that the size of the j-invariant of E is $\log_2 p$. Since the size of $\ell_S, e_S, \ell_R, e_R, f$ is negligible with respect to p, E, P_S, Q_S, P_R, Q_R, we only consider the size of p, E, P_S, Q_S, P_R, Q_R.

The size of public key is $2 \log_2 p$, which is the size of the j-invariant of E_s. The size of a private key is $2\log_2 p$ which is the size of the x-coordinate of a generating point of a kernel.

The size of signature appeared in Section 5 is about $n + 2n \log_2 p$ which is the sum of the sizes of e and the size of $s = \sum_{i=1}^{n} \log_2(x\text{-coordinate}(K_i))$. A signature can be compressed moe as described bellow.

Compressing signature For a signing process, wee need to generate n points $S + R_1, \cdots, S + R_n \in E[\ell_R^{e_R}]$ and their images $\phi(R_1), \cdots, \phi(R_n)$ where R_i's are randomly chosen point of E of order $\ell_R^{e_R}$. We propose the following method : at first signer generates two linearly independent points $U, V \in E[\ell_R^{e_R}]$ such that $\langle U, V \rangle = E[\ell_R^{e_R}]$. The signer can use an efficiently computable non-degenerated bilinear pairing map on $E[\ell_R^{e_R}]$ to validate U and V. Let $U_S = \phi(U), V_S = \phi(V)$ be the images of U, V in E_S. Then, by the discussion of [7], a point of order $\ell_R^{e_R}$ can be sampled in the form $U + [\ell_R m]V$ where an integer m is chosen uniformly at random from $\{1, 2, \cdots, \ell_R^{e_R - 1} - 1\}$. Set $G := S + U$. Then we see that $G + [\ell_R m]V = S + R$ where R is a point of order $\ell_R^{e_R}$, and that $\phi(R) = U_S + [\ell_R m]V_S$. Thus both the kernel $\langle S + R \rangle$ of α, and the kernel $\langle \phi(R) \rangle$ of β are represented in a positive integer m less than $\ell_R^{e_R - 1}$. Therefore signer generates a sequence of integers $m_1, \cdots, m_n$ where each m_i represents the i-th generating points of kernels $S + R_i = G + [\ell_R m_i]V \in E$ and $\phi(R_i) = U_S + [\ell_R m_i]V_S \in E_S$. Signer puts the integer m_i in the i-th place when $b_i = 0$, or a generating point $\phi(R_i)$ when $b_i = 1$. It should be noted that U or a pair (U_S, V_S) must not be revealed, since then the secret key S will be revealed.

The following algorithms are processes for generating and verifying compressed signatures. The domain parameter and key pair settings are equivalent to those in Section 5.

Signature generation (compressed version)
INPUT : a domain parameter $\mathcal{D}$, a private key S,
a message $m \in \mathcal{M}$, a randomly chosen *seed*.
OUTPUT : the digital signature (e, s) for m.
Perform the following steps :

1. Init $s \leftarrow ()$.

2. Choose points $U, V \in E$ at random such that $\langle U, V\rangle = E[\ell_R^{e_R}]$.
3. Compute the points $U_S = \phi(U)$, $V_S = \phi(V)$.
4. Put $G = S + U$.
5. Append G, V to s.
6. For $i = 1, \cdots, n$ do the following steps :
   (a) Choose $m_i \in \{1, 2, \cdots, \ell_R^{e_R - 1} - 1\}$ at random.
   (b) Put $K_i = U_S + [\ell_R m_i]V_S$.
   (c) Compute isogeny $\beta_i : E \to E_i$ with $\ker \beta_i = \langle K_i\rangle$.
7. Put $r = (E_1, \cdots, E_n)$.
8. Compute $e = H(r||m)$.
9. Put $(b_1, \cdots, b_n) = e$ with $b_i = 0$ or 1.
10. For $i = 1, \cdots, n$
    If $b_i = 0$ then append m_i to s,
    else append K_i to s.
11. Output (e, s).

The output of signature is of the form $s = (G, V, \{m_1 \text{ or } K_1\}, \cdots, \{m_n \text{ or } K_n\})$.

Signature verification (compressed version)
INPUT : a domain parameter $\mathcal{D}$, a message $m \in \mathcal{M}$, a public key E_s,
a digital signature (e', s') where $s' = (G', V', \{m_1' \text{ or } K_1'\}, \cdots, \{m_n' \text{ or } K_n'\})$.
OUTPUT : status (TRUE or FALSE)
Perform the following steps :

1. Put $(b_1', \cdots, b_n') = e'$ with $b_i' = 0$ or 1.
2. For $i = 1, \cdots, n$, do the following steps :
   (a) If $b_i' = 0$, then compute isogeny $\alpha_i : E \to E_i'$ with $\ker \alpha_i = \langle G' + [\ell_R m_i']V'\rangle$.
   If $b_i' = 1$, then compute isogeny $\beta_i : E_S \to E_i'$ with $\ker \beta_i = \langle K_i'\rangle$.
3. Put $r' = (E_1', \cdots, E_n')$.
4. If $e' = H(r'||m)$, then output TRUE
   else output FALSE.

Then the following result holds.

Theorem 2. *The signature size of the proposed digital signature scheme is*

$$n + c + \left(\tfrac{5}{4}n + 3\right)\log_2 p,$$

at average where n is the size of hash, and c is the size of seed.

Proof. We represent a generating point by its x-coordinate, and need not care the y-coordinate since $\langle G\rangle = \langle -G\rangle$ for a point $G \in E$. Let $|G|$ denote the bit size of x-coordinate of G. Note that we can choose V in order that V has x-coordinate in $\mathbb{F}_p$. The signature size is, (e size) + (s size) = (hash size) + $|G|+|V|+\log_2(seed)+\frac{1}{2}\sum|m_i|+\frac{1}{2}\sum|K_i| = n+\log_2 p^2+\log_2 p+c+\frac{n}{2}\log_2 p^{\frac{1}{2}}+\frac{n}{2}\log_2 p^2 = n + c + \left(\frac{5}{4}n + 3\right)\log_2 p$, at average. □

We see that now the signature size of our proposed scheme becomes 5/6 times smaller than that of [10]. This is possible since triangular structure of ZKP-II which uses only one elliptic curve as a commitment, while the same technique can not be applied to the first scheme of [10].

Speed We compare the speed of our proposed scheme with *the first scheme (TFS)* of Galbraith et al. [10], which is said to be significantly efficient than the second scheme. TFS computes $2n$ isogenies for signature generation while our proposed scheme computes the isogeny ϕ one time and β, n times, which is two times faster than TFS. For signature verification we compute either α or β depending on bits of e. Note that the computational cost of α is approximately equal to the sum of the computational costs of ϕ and β. The verification cost is equal to TFS.

Counting the cost of isogeny computations, we can predict the speed referring to the benchmark result on a 3.4 GHz Intel Core i7-2600 and Intel Core i7-4770 for Sandybridge and Haswell, respectively [7].

Operation	Sandy Bridge	Haswell
Alice's keygen	50	46
Bob's keygen	57	52
Alice's shared key	47	44
Bob's shared key	55	50
Total	207	192

Costello at al. Speed (cc $\times 10^6$)

Since Costello et al. aim at 192-bit classical security (128-bit quantum security), to aim at the same security level we take $n = 384$. Then, a signature generation process takes about $19,200 \times 10^6$cc, since it needs about 384 isogeny computations. In the same manner a verification process will take $28,800 \times 10^6$cc at average.

8 Conclusion

We have proposed a Fiat-Shamir type digital signature scheme based on supersingular isogeny problem. The signature scheme is proven to be EUF-CMA secure by the forking lemma. We provide a computational method for modified version of ZKP, which uses one curve as a commitment. Additionally, we also proposed a method for compressing signature by representing kernels with coefficients of generators. The size of signature and speed are improved significantly from the first version of supersigular isogeny digital signature scheme proposed by Galbraith et al. Our proposed scheme can be considered to be acceptable as a candidate of digital signature scheme against quantum algorithms.

References

1. R. Azarderakhsh, D. Jao, K. Kalach, B. Koziel, C. Leonardi, *Key compression for isogeny-based cryptosystems*, Proceedings of the 3rd ACM International Workshop on ASIA Public Key Cryptography. AsiaPKC '16, New York, NY, USA, ACM 1-10 (2016).

2. R. Balasubramanian and N. Koblitz, *The improbability that an elliptic curve has subexponential discrete log problem under the Menezes-Okamoto-Vanstone algorithm*, Journal of Cryptology, 11 (1998), 141-145.
3. M. Bellare, G. Neven, *Multi-signatures in the plain public-key model and a general forking lemma*, ACM Conference on Computer and Communications Security 2006, 390-399.
4. J. Biasse, D. Jao, and A. Sankar. *A quantum algorithm for computing isogenies between supersingular elliptic curves*, In W. Meier and D. Mukhopadhyay, editors, INDOCRYPT 2014, volume 8885 of LNCS (2014), 428-442.
5. R. Bröker. *Constructing supersingular elliptic curves*, J. Comb. Number Theory, 1(3) (2009), 269-273.
6. A. Childs, D. Jao, and V. Soukharev, *Constructing elliptic curve isogenies in quantum subexponential time*, Journal of Mathematical Cryptology, 8(1) (2013), 1-29.
7. C. Costello, P. Longa, and M. Naehrig, *Efficient algorithms for supersingular isogeny Diffie-Hellman*, Advances in Cryptology - CRYPTO 2016 Proceedings, Part I (2016), 572-601.
8. L. De Feo, D. Jao, and J. Plût, *Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenie*, J. Mathematical Cryptology, 8(3) (2014), 209-247.
9. U. Feige, A. Fiat, and A. Shamir, *Zero-knowledge proofs of identity*, Journal of Cryptology, 1(2) (1988), 77 - 94.
10. S. Galbraith and C. Petit and J. Silva, *Signature Schemes Based On Supersingular Isogeny Problems*, IACR Cryptology ePrint Archive, 2016:1154, 2016.
11. D. Kohel, K. Lauter, C. Petit, and J.-P. Tignol, *On the quaternion ℓ-isogeny path problem*, LMS Journal of Computation and Mathematics, 17A (2014), 418-432.
12. B. Koziel, R. Azarderakhsh, A. Jalali, D. Jao, and M. Mozaffari Kermani, *NEON-SIDH: Efficient implementation of supersingular isogeny Diffie-Hellman key exchange protocol on ARM*, in Proc. Conf. Cryptology and Network Security (CANS) (2016), 88-103.
13. B. Koziel, R. Azarderakhsh, M. Kermani, *Fast Hardware Architectures for Supersingular Isogeny Diffie-Hellman Key Exchange on FPGA*, INDOCRYPT 2016 (2016), 191-206.
14. S. Lang, *Elliptic functions*, 2nd edition, Springer GTM 112, 1987.
15. O. Goldreich, S. Micali, and A. 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) (1991), 690-728.
16. T. Seiichiro. *Claw finding algorithms using quantum walk*, Theoretical Computer Science, 410(50):5285-5297, 2009. Mathematical Foundations of Computer Science (MFCS 2007).
17. A. Stolbunov, *Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves*, Adv. in Math. of Comm. 4(2) (2010), 215-235 .
18. J. Vélu. *Isogénies entre courbes elliptiques*, C. R. Acad. Sci. Paris Sér. A273 (1971b), 238-241.
19. S. Galbraith, *Mathematics of Public Key Cryptography*, 1st edition, Cambridge University Press New York, NY, USA 2012.
20. W. Waterhouse, *Abelian Varieties over Finite Fields*, Ann. Se. E.N.S., (4), 2, (1969), 521-560.