An Isogeny-based Dealer-Less Threshold Signature Scheme

WANG YUSEN [1] MIYAJI ATSUKO [1, 2]

Abstract: A threshold HHS [9] (Hard Homogeneous Spaces) signature scheme had been proposed by Luca De Feo et al [2]. Generally, CSI-FiSh [4] and its ancestors [1][8] can be adapted into this threshold scheme. In this work, we mainly focus on the basic version of CSI-FiSh proposed by Stolbunov [8].

In the isogeny-based threshold signature scheme sketched by Luca De Feo et al, a dealer is necessary in order to split the secret key into shares and to securely distribute them to all participants. However, in certain conditions, a trusted dealer which is a trusted third party is not permitted or does not exist. Therefore we proposed a dealer-less version based on the threshold scheme of Luca De Feo et al. We use Joint Random-Secret Sharing to let all participants exchange information with each other and thus can collaborate to generate a secret isogeny that is not revealed to any participants.

In this work, we clarify some fundamental theories such as Hard Homogeneous Spaces, Joint Random-Secret Sharing, and Shamir's Secret Sharing. After that, we briefly take a review of the threshold HHS signature scheme proposed by Luca De Feo et al. We give the whole process of our dealer-less threshold scheme and then illustrate the comparison with the original one with a dealer. We believe that our dealer-less version has a wider range of applications.

Keywords: Hard Homogeneous Spaces (HHS); Threshold signature; Joint Random-Secret Sharing (JRSS); Isogeny

1. Introduction

Isogeny-based cryptography has attracted much attention in recent years, and many Isogeny-based methods such as SIDH [6] and CSIDH [1] have given many ideas and hints to later scholars. Therefore, identification protocols and signature schemes based on the isogeny have emerged in an endless stream recently. For example, CSI-FiSh is a signature scheme proposed by optimizing CSIDH, and it satisfies HHS which is also the main topic in this work. SQISign [5] is an outstanding signature scheme generated by Luca De Feo et al. In their work, more specifically, the corresponding identification protocol uses challenge isogeny instead of challenge bits. This means we do not have to repeat the identification protocol and it is no doubt that the efficiency of SQISign is much better than most of the isogeny-based signature schemes.

Although specialists have come up with a lot of isogeny-based signature schemes, the implementations, such as multi-signature and threshold signature, are still need to be exploited.

Nowadays, a lot of threshold signature schemes based on RSA and DLP are already generated. However, until 2019, thanks to De Feo et al, we have an isogeny-based threshold signature scheme based on Shamir's secret sharing and Hard homogeneous spaces.

A k-of-n threshold signature scheme is to split a secret key into secret shares and then send shares to different participants. With the cooperation of k or more than k parties, decryption or signing can be completed successfully. If there are fewer than k parties, they are not able to do so.

A dealer sometimes is necessary for a threshold signature scheme. Dealer is a trusted third party and it can split the secret key and distribute shares to all participants. However, we are always trying to propose the dealer-less version because in some particular conditions we are not able to get a trusted third party.

In our work, a dealer is not necessary because the n participants can exchange information (however secretly) and generate secret shares by themselves based on Joint Random-Secret Sharing. Then they are able to collaborate in order to generate a secret key but no one knows the exact secret key.

Our contributions are to propose a dealer-less threshold signature scheme based on isogeny to fulfill the assumption that there is no trusted third party. We construct such a dealer-less threshold scheme by using Joint Random Secret Sharing. We also illustrate the comparison between the original one with ours in the last part of this work.

2. Preliminaries

2.1 Isogeny

An isogeny is a map between two elliptic curves, and is denoted by $\phi: E \to E'$. The following properties should be satisfied:
(1) ϕ is a morphism. That is to say, by using two rational expressions $F(x,y)$ and $G(x,y)$, it can be proved that

$$\phi\big((x,y)\big) = \big(F(x,y), G(x,y)\big).$$

(2) The equation $\phi(\mathcal{O}) = \mathcal{O}$ holds. Here $\mathcal{O}$ is the point at infinity of an elliptic curve.

We also call E and E' are isogenous if there is an isogeny between them. If an isogeny is a bijection, then it is an isomorphism. E and E' are isomorphic. If in the isogeny $\phi(\mathcal{O}) = \mathcal{O}$, all points of elliptic curve E map to point at infinity $\mathcal{O}$, then this isogeny is a zero isogeny. A zero isogeny can be

1 Osaka University, 1-1 Yamadaoka, Suita,Osaka 565-0871 Japan
2 Japan Advanced Institute of Science and Technology, 1 Chome-1 Asahidai, Nomi, Ishikawa 923-1211

denoted by [0]. Therefore, an isogeny satisfies $\phi(E) = E'$ or $\phi(E) = \mathcal{O}$.

We also introduce several theories of isogeny:

- An isogeny from an elliptic curve E to the other curve E' is a homomorphism, so the following equation holds

$$\phi(P + Q) = \phi(P) + \phi(Q)$$

- For an elliptic curve E and its finite subgroup $\Phi \subset E$, there exist another elliptic curve E' and an isogeny denoted by $\phi: E \to E'$.

- For an isogeny $\phi: E \to E'$ ($\phi \neq [0]$), there exists another isogeny $\phi': E' \to E$ satisfies $\phi' \circ \phi = [n]$. Here $deg(\phi) = n$.

Let E be an elliptic curve over a field F_p, and E is determined as an ordinary elliptic curve or a supersingular elliptic curve as follows:

If $E[p] \simeq Z/pZ$, then E is ordinary;
If $E[p] = 0$, then E is supersingular.

A supersingular isogeny represents a map between two supersingular elliptic curves.

2.2 Shamir's secret sharing

Shamir's secret sharing [10] is based on Lagrange's interpolation formula. And it is to construct a threshold secret sharing and is well used in the threshold signature scheme.

Firstly, a dealer generates the following polynomial

$$f(x) = s + \sum_{i=1}^{k-1} c_i x^i;$$

s is the secret generated randomly by the dealer from Z/qZ. And $c_i \in Z/qZ$. ($1 \leq i \leq k-1$) are random coefficients. By using this formula, the dealer is able to compute different shares $s_j = f(j)$ $(1 \leq j \leq n)$, then distribute shares to all participants $\mathcal{P}_i$.

Obviously, the parameter s can be computed by $f(0)$, or by using Lagrange's interpolation formula, k or more than k participants are able to recompute secret s through the following formula

$$s = f(0) = \sum_{i \in S} f(i) \cdot \prod_{j \in S, j \neq i} \frac{j}{j-i}$$

$S \subset \{1,2,\dots,n\}$ and the cardinality of S is at least k. Basically, we are able to construct a threshold signature scheme by using Shamir's secret sharing.

2.3 Hard homogeneous spaces

Hard homogeneous spaces [9] were first proposed by Couveignes. HHS can be considered as a finite principal homogeneous space with more properties.

Generally, a principal homogeneous space is a set $\mathcal{E}$ which is able to do transitive group action with a group $\mathcal{G}$, which means

$$\mathcal{G} \times \mathcal{E} \to \mathcal{E}$$
$$g * E = E'$$

If the group $\mathcal{G}$ is finite, then we need the cardinalities of $\mathcal{G}$ and $\mathcal{E}$ to be the same, in other words, $\#\mathcal{E} = \#\mathcal{G}$. For a principal homogeneous space, the following properties should be satisfied:

- For any $g, g' \in \mathcal{G}$ and $E \in \mathcal{E}$, $g' * (g * E) = (g'g) * E$;
- $e * E = E$ if and only if $e \in \mathcal{G}$ is the identity element;
- There exists a unique element $g \in \mathcal{G}$ for any $E, E' \in \mathcal{E}$ that satisfies $g * E = E'$.

In Hard homogeneous spaces, we assume that group $\mathcal{G}$ and set $\mathcal{E}$ can be represented by strings. But we don't care about the detail of representation. And the following problems should be easy to solve

- Given strings g, g', check whether they can represent elements in group $\mathcal{G}$ and whether they are equal or not. Given $g, g' \in \mathcal{G}$, compute g^{-1}, gg' and check whether $g = g'$;
- Get a random element of group $\mathcal{G}$ uniformly;
- Given a string E, check if it represents an element in set $\mathcal{E}$;
- Given $E \in \mathcal{E}, g \in \mathcal{G}$, compute $g * E$.

Also the following problem should be hard enough to solve

- (Vectorization) Given E, E' decide an element $g \in \mathcal{G}$ that $g * E = E'$;
- (Parallelization) Given E_1, E_2, E_3 and $g \in \mathcal{G}$, compute a unique E_4 that satisfies $g * E_1 = E_2$ and $g * E_3 = E_4$.

It is obvious that CSI-FiSh satisfies all properties of HHS. In the next two sections, we discuss the threshold HHS signature proposed by De Feo et al and we adapt the basic version of CSI-FiSh into the threshold HHS signature scheme.

Most of the exist isogeny-based signature schemes do not satisfy hard homogeneous spaces, but the precomputation of CSI-FiSh perfectly satisfies all properties of hard homogeneous spaces. Therefore, we are able to adapt the precomputation of CSI-FiSh into CSIDH, although CSIDH is not a perfect HHS scheme.

2.4 Joint random-secret sharing

JRSS [11] (short for Joint random-secret sharing) can be used in a dealer-less threshold signature scheme. By using JRSS, all participants can exchange information, compute their own secret shares, and then collaborate to generate the secret key. However, the secret key will not be revealed to any participants. Here are details about JRSS [3].

Each player $\mathcal{P}_i$:

- samples $t+1$ values $a_0^{(i)}, \dots, a_t^{(i)} \in Z_p$ denoted by r_i;
- constructs a polynomial $R_i(x) = \sum_{j=0}^{t} a_j^{(i)} x^j$;
- for $\forall j = (1, \dots, n)$, computes and secretly sends $R_i(j)$ to the corresponding player $\mathcal{P}_j$;
- sums the values that received from others which denoted by $\sigma(i) = \sum_{j=1}^{n} R_j(i)$;

Therefore each $\sigma(i)$ is a point on a t-degree polynomial $\sigma(x)$. Here we consider $\sigma(i)$ as a secret share s_i and $\sigma(0) = \sum_{j=1}^{n} R_j(0) = \sum_{j=1}^{n} (\sum_{j^*=0}^{k} a_{j^*}^{(j)} 0^{j^*}) = \sum_{i=1}^{n} a_0^{(i)}$ is considered to be the secret key s. We are also able to compute s by using the following formula:

$$\sigma(0) = \sum_{i=1}^{k} \sigma(i) \cdot \prod_{j \in S, j \neq i} \frac{j}{j-i}$$

In our dealer-less threshold signature scheme, we use joint random-secret sharing to let participants exchange the values of polynomial in secret channels and they are able to generate secret share and collaborate to propose secret key.

3. Threshold signature based on HHS

Here we clarify threshold HHS signature scheme [2]. We use $[a]$ to denote g^a, and use $[a]E$ to denote $g^a * E$, g is an element of order q in group $\mathcal{G}$ ($a \in Z/qZ$, $g \in \mathcal{G}$ and $E \in \mathcal{E}$). And it should be noticed that based on HHS, $[a_1][a_2]E = [a_1 + a_2]E$. The participant set is denoted by S.

Firstly, a dealer generates a random element $E_0 \in \mathcal{E}$, and set $E_j^0 = E_0 (1 \leq j \leq t)$. For each participant $\mathcal{P}_i$, $i \in S$, he checks whether E_j is an element of set $\mathcal{E}$, if not, then aborts the whole protocol. Then $\mathcal{P}_i$ samples an integer $b_{i,j}$ from Z/qZ randomly, and outputs $E_j^k \leftarrow [b_{i,j}]E_j^{k-1}$ (k is considered to be a count number, and is initially 1. For each action of a participant, $k = k + 1$).

After the last participant outputting his E_j^k, dealer will generate challenge bits

$$c_1, \dots, c_j, \dots, c_t \leftarrow H(E_1^k, \dots, E_j^k, \dots, E_t^k, m).$$

Finally, for each participant $\mathcal{P}_i$, $i \in S$, outputs $z_{i,j} = b_{i,j} - c_j * s_i * L_{0,i}^S$, here $L_{0,i}^S$. Thus $z_j = \sum_{i \in S} z_{i,j}$.

The threshold signature is $(c_1, \dots, c_j, \dots, c_t, z_1, \dots, z_j, \dots, z_t)$.

The whole threshold HHS signature scheme is as Algorithm 1:

Algorithm 1 Threshold HHS signature scheme

Input Message m and participant set S;
Output A threshold signature.

1: Dealer generates a random element $E_0 \in \mathcal{E}$ and $E_j^0 = E_0$ $(1 \leq j \leq t)$;
2: $k = 0$;
3: **for** $i \in S$, $\mathcal{P}_i$ **do**
4: $k = k + 1$;
5: **for** $j \in [1, \lambda]$ **do**
6: **if** E_j is in $\mathcal{E}$ **then**
7: samples a random $b_{i,j}$ from $\mathbb{Z}/q\mathbb{Z}$;
8: outputs $E_j^k \leftarrow [b_{i,j}]E_j^{k-1}$
9: **else**
10: aborts.
11: **end if**
12: **end for**
13: **end for**
14: Dealer generates challenge bit list $c_1, ..., c_j, ..., c_t \leftarrow H(E_1^k, ..., E_j^k, ..., E_t^k, m)$;
15: **for** $i \in S$, $\mathcal{P}_i$ **do**
16: **for** $j \in [1, \lambda]$ **do**
17: outputs $z_{i,j} = b_{i,j} - c_j * s_i * L_{0,i}^S$, here $L_{0,i}^S$. Thus $z_j = \sum_{i \in S} z_{i,j}$.
18: **end for**
19: **end for**

4. Dealer-Less threshold signature scheme based on the basic version of CSI-FiSh

4.1 The basic version of CSI-FiSh

The basic version of CSI-FiSh was proposed by Stolbunov [8] and he generated such a signature scheme by using Fiat Shamir Transform [7] on the identification protocol sketched by Couveignes [9]. Therefore, we introduce the identification protocol firstly and then illustrate the signature scheme.

The prover owns two public keys E_0, E_1 and $E_1 = a * E_0$. Here a is an element of class group $\mathrm{Cl}(\mathcal{O})$ which is cyclic. g is the generator of class group $\mathrm{Cl}(\mathcal{O})$ and the group action g^a can be also denoted by a. The prover proposes another curve E_s which satisfies $E_s = [b]E_0$ and b is random sampled from Z_N, $N = \#\mathrm{Cl}(\mathcal{O})$. Then the prover will accept a challenge bit c and based on this challenge bit, he decides the information to reveal.

When the challenge bit $c = 0$, then the prover reveals element $z = b$ which represents an isogney from E_0 to E_s. When $c = 1$, the prover responds the verifier with an element $z = (b - a) mod N$ and this element represents the isogeny from E_1 to E_s.

The verifier then checks whether $E_s = [z]E_0$ or $E_s = [z]E_1$ when the challenge bit $c =$ 0 or 1 respectively. If so, then accept; he rejects otherwise. The diagram is as follows:

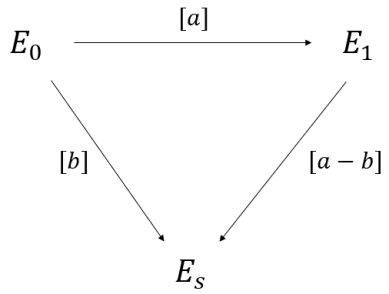

Figure 1

Stolbunov proposed the basic version of CSI-FiSh by using Fiat Shamir Transform. Here is the detail.

In signature generation, by inputting secret key a and public key E_0, E_1, the signer first randomly samples t integers b_i ($i \in [1, t]$) generates t different isogenies denoted by $E_0 \rightarrow E_{s,i}$ and $E_{s,i} = [b_i]E_0$. Then computes $c_1, \dots, c_i, \dots, c_t \leftarrow H(E_{s,1}, \dots, E_{s,i}, \dots, E_{s,t}, m)$, and $z_i = b_i - c_i a$. The signature is $(c_1, \dots, c_i, \dots, c_t, z_1, \dots, z_i, \dots, z_t)$.

In verification algorithm, by inputting public key E_0, E_1, signature $(c_1, \dots, c_i, \dots, c_t, z_1, \dots, z_i, \dots, z_t)$ and hash function H, verifier checks whether the output of $H(E'_{s,1}, \dots, E'_{s,i}, \dots, E'_{s,t}, m)$ is equal to $(c_1, \dots, c_i, \dots, c_t)$. When $c_i = 0$, $E'_{s,i} = [z_i]E_0$; when $c_i = 1$, $E'_{s,i} = [z_i]E_1$. If $H(E'_{s,1}, \dots, E'_{s,i}, \dots, E'_{s,t}, m)$ is equal to $(c_1, \dots, c_i, \dots, c_t)$ then verifier accepts, otherwise rejects.

4.2 Our dealer-less threshold signature scheme

In this section, we introduce our dealer-less threshold signature scheme based on the basic version of CSI-FiSh. The details are precisely given in Algorithm 2, Algorithm 3, and Algorithm 4. The main difference between the original one based on De Feo's

threshold scheme and ours is that ours works without a dealer. However, in our scheme, we need more secret channels due to the secret information exchange in key generation algorithm. We analyze the efficiency of De Feo's and ours in the next section.

In key generation algorithm, all n members first sample k+1 random integers $a_0^{(i)}, \dots, a_k^{(i)}$, and then each of them proposes his own polynomial $R_i(x) = \sum_{\alpha=0}^{k} a_\alpha^{(i)} x^\alpha$. Each participant computes $R_i(j)$ for $\forall j = (1, \dots, n)$. Relatively each participant $\mathcal{P}_i$ will receive $\sigma(i) = \sum_{j=1}^{n} R_j(i)$ and this polynomial can be considered as a secret share s_i. Then they collaborate to construct an isogeny from E_0 to E_s, which denoted by $E_s = [s]E_0$. However, the secret key s is not revealed to any participants. The public keys are E_0 and E_s. In De Feo's scheme, dealer generates a secret key s and splits it into secret shares s_i and sends them to corresponding participants secretly. That means both De Feo's scheme and ours needs secret channels.

In the signing algorithm and verification algorithm, De Feo's scheme and ours are almost the same. k participants generate t isogenies and send the elliptic curves to the next participant. Although there does exist an order of participants, the order will not affect the result and this is an essential property of HHS signature.

We need to clarify the reason why each participant has to generate t isogenies and it is definitely not an efficient choice. Because De Feo's threshold signature is based on an identification algorithm. The identification algorithm has to be repeated because it uses a challenge bit. We want a larger challenge space in order to decrease the probability of being attacked. When the identification algorithm is repeated for t times, because of Fait Shamir Transform, in De Feo's signature scheme and ours, participants have to generate t isogenies respectively. When it comes to challenge space, SQISign is definitely a better choice for using challenge isogeny instead of challenge bits. That means SQISign's corresponding identification protocol does not need to be repeated, and it shows that SQISign is much more efficient than most other isogeny-based signature schemes.

Finally, we get a threshold signature consisting of challenge bits c_j and z_j. Verifier will check whether the signature is valid or not. There are two different conditions, one is that when $c_j = 0$, all participants need to collaborate to reveal the isogeny from E_0 to $E_{k,j}$. The other one is that when $c_j = 1$, they need to reveal the isogeny from E_s to $E_{k,j}$. If the challenge bit-list recomputed by the verifier is equal to c, the verifier accepts, otherwise, he rejects.

Algorithm 2 Key Generation

Input All n members.
Output Public Key E_0, E_s.

for $1 \le i \le n$, $\mathcal{P}_i$ **do**
  Samples $k+1$ random integers $a_0^{(i)}, \dots, a_k^{(i)}$;
  Proposes $R_i(x) = \sum_{\alpha=0}^{k} a_\alpha^{(i)} x^\alpha$;
  for $\forall j = (1, \dots, n)$ sends $R_i(j)$ to corresponding $\mathcal{P}_j$;
  Receives $s_i = \sigma(i) = \sum_{j=1}^{n} R_j(i)$, R_i^i (s_i is considered to be a secret share) generated by $\mathcal{P}_i$ itself.
end for
for $1 \le i \le n$, $\mathcal{P}_i$ **do**
  if $i = 1$ **then**
    Generates $E_0 = E_0^*$ and sends $E_1^* = a_0^{(1)} E_0^*$ to $\mathcal{P}_1$;
  else if $1 < i < n$ **then**
    Sends $E_i^* = a_0^{(i)} E_{i-1}^*$ to $\mathcal{P}_{i+1}$;
  else
    Outputs $E_n^* = a_0^{(n)} E_{n-1}^*$.
  end if
end for
It is obvious that $s = \sigma(0) = \sum_{j=1}^{n} R_j(0) = \sum_{i=1}^{n} a_0^{(i)}$.
Secret Key: s, public Key: E_0, E_s.

Algorithm 3 Signing

Input message m, k participants $\mathcal{P}_i$ $i \in S$.
Output A threshold signature on message m.

$N = \mathrm{Cl}(\mathcal{O})$;
for $1 \le i \le k$, $\mathcal{P}_i$ **do**
  if $1 \le i < k$ **then**
    Produces t different integers $b_{i,j} \in \mathbb{Z}_n, 1 \le j \le k$;
    Sends t different curves $E_{i,j} = [b_{i,j}]E_{i-1,j}$ to next participants $\mathcal{P}_{i+1}$.
  else
    $\mathcal{P}_k$ outputs $E_{k,j}$.
  end if
end for
Every party can do the following steps:
compute challenge bits $c = [c_1, \dots, c_j, \dots, c_t] \leftarrow H(E_{k,1}, \dots, E_{k,j}, \dots, E_{k,t}, m)$;
for $1 \le j \le t$ **do**
  $z_{i,j} = b_{i,j} - c_j * s_i * L_{0,i}^S$
  compute $z_j = \sum_{1 \le i \le k} z_{i,j}$.
end for
return $\sigma = (c_1, \dots, c_t, z_1, \dots, z_t)$.

Algorithm 4 Verification

Input message m, threshold signature σ and hash function H.
Output Valid or invalid.

for $1 \le j \le k$ **do**
  if $c_j = 0$ **then**
    $E_j = E_0$;
  else
    $E_j = E_s$.
  end if
end for
if $c' = H([z_1]E_1, \dots, [z_t]E_t, m) = c$ **then**
  return Valid;
else
  return Invalid.
end if

The diagram of our threshold signature scheme is as follows:

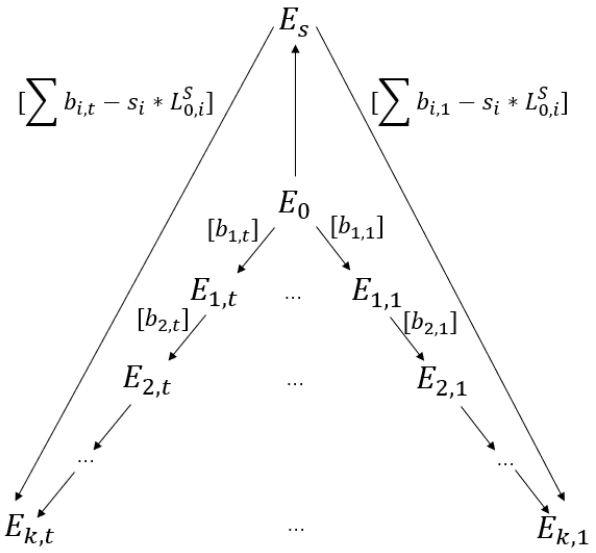

Figure 2

5. Comparison

We clarify the comparison of De Feo's scheme and ours. The details are given in table 1.

Table 1

	Ours	De Feo's
Key Generation	Slower	Faster
Signing	The same	
Verification	The same	
Signature Size	The same	
Dealer	Not exist	Exist
Signing order of participants	Not fixed	Not fixed
Communication amount in KeyGen	$\|l_1\| * n * (n-1) + \|l_3\| * (n-1)$	$\|l_2\| * n$
Communication amount in Signing	The same	

As the results in Table 1, the key generation of our scheme is not efficient as De Feo's, because ours works without a dealer, and we need more secret channels to exchange values of polynomial between each two participants to generate their own secret shares. And in De Feo's, this is done by dealer. Dealer is able to split secret key s into shares s_i and secretly sends to all participants.

The signing algorithm and verification algorithm of De Feo's and ours are the same. Therefore, our signature size is the same as De Feo's.

The main difference between ours and De Feo's is that ours works without a dealer. And this property is more in line with reality. We focus on the optimization of our scheme to make it more efficient and useful.

It should be mentioned that the result will not be affected if we change the order of participants. This is different from traditional isogeny computation and it is an essential property of HHS.

The communication amount of key generation in our scheme is $|l_1| * n * (n-1) + |l_3| * (n-1)$, and $|l_1|$, $|l_2|$, $|l_3|$ are respectively the size of $R_i(j)$, s_i, E. The reason why we need more communications has been illustrate in the part of key generation.

6. Conclusion

In this paper, we clarify Shamir's secret sharing which is the fundamental theorem of threshold cryptography. And we also introduce hard homogenous spaces which are definitely one of the most significant theorems in our work. By using HHS, it is able to adapt supersingular isogenies into a threshold scheme based on the threshold HHS signature scheme proposed by Luca De Feo et al. Moreover, we illustrate the detail of joint random-secret sharing which allows all participants to exchange information secretly and then generate their own secret shares without a dealer. Then all participants are able to collaborate to compute a secret isogeny but this isogeny is not reveal to any participants. After JRSS, we introduce the threshold HHS signature proposed by De Feo and then the basic version of CSI-FiSh is also mentioned. CSI-FiSh and its other ancestors are believed to be easily adapted into our dealer-less threshold scheme and also the original version by Luca De Feo et al.

We proposed a dealer-less threshold signature scheme based on the basic version of CSI-FiSh by using joint random-secret sharing. And our scheme is also a variant of De Feo's threshold HHS signature scheme. Ours satisfies the condition with no trusted third party and it seems more in line with reality.

We focus on improving the key generation algorithm such as using public channels and decreasing the communication amount in the future. We hope we can propose a better and more useful isogeny-based threshold signature scheme.

Reference

[1] Castryck W., Lange T., Martindale C., Panny L., Renes J. (2018) CSIDH: An Efficient Post-Quantum Commutative Group Action. In: Peyrin T., Galbraith S. (eds) Advances in Cryptology – ASIACRYPT 2018. ASIACRYPT 2018. Lecture Notes in Computer Science, vol 11274. Springer, Cham. https://doi.org/10.1007/978-3-030-03332-3_15.

[2] De Feo L, Meyer M. Threshold schemes from isogeny assumptions[C]//IACR International Conference on Public-Key Cryptography. Springer, Cham, 2020: 187-212..

[3] Ibrahim, Maged Hamada. "Efficient Dealer-Less Threshold Sharing of Standard RSA." *Int. J. Netw. Secur.* 8.2 (2009): 139-150.

[4] Beullens, W., Kleinjung, T., & Vercauteren, F. (2019, December). CSI-FiSh: Efficient isogeny based signatures through class group computations. In *International Conference on the Theory and Application of Cryptology and Information Security* (pp. 227-247). Springer, Cham.

[5] De Feo, L., Kohel, D., Leroux, A., Petit, C., & Wesolowski, B. (2020, December). SQISign: compact post-quantum signatures from quaternions and isogenies. In *International Conference on the Theory and Application of Cryptology and Information Security* (pp. 64-93). Springer, Cham.

[6] Jao D, De Feo L. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies[C]//International Workshop on Post-Quantum Cryptography. Springer, Berlin, Heidelberg, 2011: 19-34.

[7] Fiat, A., & Shamir, A. (1986, August). How to prove yourself: Practical solutions to identification and signature problems. In *Conference on the theory and application of cryptographic techniques* (pp. 186-194). Springer, Berlin, Heidelberg.

[8] Stolbunov, Anton. "Cryptographic schemes based on isogenies." (2012).

[9] Couveignes J M. Hard Homogeneous Spaces[J]. IACR Cryptol. ePrint Arch., 2006, 2006: 291.

[10] Shamir A. How to share a secret[J]. Communications of the ACM, 1979, 22(11): 612-613.

[11] Kaya K, Selçuk A A. A verifiable secret sharing scheme based on the chinese remainder theorem[C]//International Conference on Cryptology in India. Springer, Berlin, Heidelberg, 2008: 414-425.

Acknowledgments This work is partially supported by enPiT(Education Network for Practical Information Technologies) at MEXT, JSPS KAKENHI Grant Number JP21H03443, and Innovation Platform for Society 5.0 at MEXT.