Research Article

Mingping Qi*

An efficient post-quantum KEM from CSIDH

https://doi.org/10.1515/jmc-2022-0007
received December 12, 2021; accepted May 03, 2022

Abstract: The SIDH and CSIDH are now the two most well-known post-quantum key exchange protocols from the supersingular isogeny-based cryptography, which have attracted much attention in recent years and served as the building blocks of other supersingular isogeny-based cryptographic schemes. The famous SIKE is a post-quantum key encapsulation mechanism (KEM) constructed on the SIDH, motivated by which, this article presents a new post-quantum KEM-based on the CSIDH, which is thereby named as CSIKE. The presented CSIKE has much higher computation efficiency in the decapsulation part by involving an additional tag in the encapsulation results. The new CSIKE is formally proved to be IND-CCA secure under the standard isogeny-based quantum resistant security assumption. Moreover, by comparing the new CSIKE with the only two existing CSIDH-based KEM schemes, i.e., CSIDH-PSEC-KEM and CSIDH-ECIES-KEM, it can be easily found that the new CSIKE has a slightly longer encapsulation size than CSIDH-PSEC-KEM and CSIDH-ECIES-KEM, but (i) it beats the CSIDH-PSEC-KEM by the improvement of approximately 50% in decapsulation speed, and (ii) it has a certain advantage over the CSIDH-ECIES-KEM in security since in the random oracle model, the security proof for CSIDH-ECIES-KEM needs to rely on the stronger CSI-GDH assumption, while the new CSIKE just needs to rely on the basic CSI-CDH assumption.

Keywords: supersingular isogeny, CSIDH, KEM, elliptic curve

MSC 2020: 94A60

1 Introduction

Key encapsulation mechanism (KEM) is an asymmetric cryptographic primitive, which is typically used to construct a hybrid public key encryption (PKE) scheme by integrating a decapsulation mechanism (DEM). The KEM-DEM framework was first formalized by Cramer and Shoup [1], in which the KEM part is a probabilistic algorithm used to simultaneously generate a random key and an encryption of this key, whereas the DEM part is a deterministic algorithm used to encrypt messages with arbitrary length under the key generated by the corresponding KEM. Up to now, many KEM schemes based on traditional cryptographic hard problems have been presented including [2–4]. In practice, KEM has been widely used to secure communications over the Internet, and the widely used TLS [5] handshake protocol has implicitly employed KEM schemes as reported in [6] by Krawczyk et al.

In recent years, the research and development of quantum computers have made great progress, and Google and IBM are at the forefront of the world in this regard as reported in refs [7,8]. To deal with the challenges against traditional public key cryptosystems brought by the emerging quantum computers, post-quantum cryptography (PQC) has gained much attention along with the development of quantum computers, and some related significant results have been presented. Currently, the promising candidates for PQC include the lattice, code, multivariate, hash and supersingular isogeny-based cryptography. Specifically, in terms of post-quantum KEM schemes, there are many options including the schemes in refs

* **Corresponding author: Mingping Qi,** School of Cybersecurity, Northwestern Polytechnical University, Xi'an 710072, People's Republic of China, e-mail: mpqi_math@163.com

[9–12], where the SIKE [10] is the KEM instance based on supersingular isogeny. Compared with other types of PQC, supersingular isogeny-based PQC has its obvious advantage that its key is relatively shorter for the same security strength. To this end, this work focuses on the research of the supersingular isogeny-based KEM with the aim to provide a secure and efficient one for option.

1.1 Related works

Historically, isogeny-based cryptography can be dated back to a talk of Couveignes in 1997 at the ENS, and after that talk, Couveignes wrote a note titled "Hard Homogeneous Spaces" [13], which was neither published nor posted publicly until 2006. Independent from Couveignes's work, Rostovtsev and Stolbunov rediscovered isogeny-based cryptography and thought this type of cryptography can resist the quantum computer attacks, and they also posted publicly their discoveries in the literature [14] in 2006. The cryptographic proposals in refs [13,14] were based on the hard problem of finding an isogeny between two given ordinary elliptic curves, which was believed to be quantum resistant. Nevertheless, Childs et al. [15] later presented a subexponential-time quantum algorithm, which can be used to recover the private keys in the ordinary elliptic curve isogeny-based cryptographic schemes. Childs et al.'s subexponential-time quantum algorithm needs to rely on the commutativity of the order denoted by $\mathcal{O}$ associated with the corresponding endomorphism ring, which implies that it is not available when the associated endomorphism ring is noncommutative. Motivated by this finding, Jao and De Feo [16] turned to use supersingular elliptic curves to construct cryptographic scheme since the full endomorphism ring of a supersingular elliptic curve is noncommutative, and the seminal supersingular isogeny-based Diffie–Hellman (SIDH) key exchange protocol was thereby presented. Since the SIDH [16] protocol was presented in 2011, supersingular isogeny-based PQC has gained much attention, and many meaningful results have been then proposed. Galbraith et al. [17] gave a full security analysis to the supersingular isogeny-based cryptography. The authors in refs [18–21], have respectively presented their techniques for efficiently implementing the SIDH protocol. Galbraith et al. [22] also presented identification protocols and signature schemes from supersingular isogenies. Currently, the state-of-the-art supersingular isogeny-based PQC is the key encapsulation mechanism SIKE [10].

On the other hand, De Feo et al. [23] revisited Couveignes–Rostovtsev–Stolbunov's cryptographic proposals in [13,14] based on ordinary elliptic curve isogenies and presented algorithmic improvement techniques to accelerate the DH-type key exchange protocol in them. Although De Feo et al.'s acceleration techniques lead to various improvements, the computation costs required to perform the cryptographic schemes based on ordinary elliptic isogenies in practice remain discouraging. A really major improvement to make the Couveignes–Rostovtsev–Stolbunov scheme practical was made by Castryck et al. [24] by creatively instantiating the Couveignes–Rostovtsev–Stolbunov scheme on the supersingular elliptic curves over a prime field $\mathbb{F}_p$, with the restriction to just consider the commutative $\mathbb{F}_p$-rational subring of the associated endomorphism ring. As a result, they presented the well-known CSIDH (meaning commutative supersingular isogeny-based Diffie–Hellman) key exchange protocol. Since the CSIDH [24] protocol was presented, many research results relevant to it were raised gradually, including [25–27].

As for KEM, it can be traced back to the seminal work [28] presented by Cramer and Shoup, which is the first really practical public key encryption scheme formally proved secure against the adaptive chosen ciphertext attack in the standard model under the decisional Diffie–Hellman (DDH) security assumption. Later, Cramer and Shoup formalized the notion of the KEM and its security against adaptive chosen ciphertext attack (i.e., what is now called IND-CCA) in the literature [1], an extended version of [28] with some results originally presented in ref. [29]. Meanwhile, in ref. [1], Cramer and Shoup presented a KEM scheme and proved it to be IND-CCA secure under the DDH assumption. Since then, using an IND-CCA secure KEM together with a secure symmetric key encryption scheme (i.e, DEM) to construct an IND-CCA secure hybrid public key encryption scheme became a typical method. In 2004, Kurosawa and Desmedt [2] presented a more efficient IND-CCA secure hybrid PKE than the basic Cramer and Shoup scheme [28], which

is an interesting hybrid PKE scheme since it has shown that the KEM does not have to be IND-CCA secure to construct an IND-CCA secure hybrid PKE scheme. In fact, the KEM part of Kurosawa and Desmedt's scheme was later reported in refs [30,31] to be not IND-CCA secure. In 2007, Kiltz [3] proposed a KEM, whose security was formally proved in the standard model under the Gap Hashed Diffie–Hellman (GHDH) assumption. Later, Kurosawa and Le Phong [32] revisited the Kurosawa and Desmedt's KEM scheme, i.e., the KEM part in the hybrid PKE scheme [2], and turned it to be an IND-CCA secure KEM under the DDH assumption by using a simple technique, i.e., involving a tag to authenticate the encapsulation ciphertext, which is also adopted in this work. Along with the development of the KEM scheme, some of them with reliable security assurance and good performance have been standardized by some International Organizations to help users to implement proper KEM schemes in practice, such as the KEM schemes standardized in ref. [4]. In addition to constructing specific KEM schemes, some modular construction methods for KEM were also proposed, including the well-known FO transformation [33], the REACT transformation [34] and GEM transformation [35].

However, it should be noted that the underlying building blocks for traditional KEM schemes are based on either the discrete logarithm problem or the integer factorization problem, which are solvable on quantum computes by using the well-known Shor quantum algorithm [36]. To this end, developing quantum resistant KEM schemes is necessary to withstand the security threats posed by quantum computers. Currently, there are many post-quantum KEM schemes that have been presented, such as the code-based KEM [9,11,12] and lattice-based KEM [37,38]. This article mainly focuses on the supersingular isogeny-based post-quantum KEM. In terms of this type of KEM, the most well-known instance is SIKE [10], which has been submitted to the NIST's PQC project [39] to compete for the PQC standards. Now, SIKE has been accepted as an alternate candidate in the third round for NIST's further evaluation [40]. In fact, SIKE is an instantiation of the transformation $U^{\not\perp}$ presented by Hofheinz et al. [41] in the supersingular isogeny context. Besides, Yoneyama [42] presented two post-quantum KEM variants of the ISO/IEC standards based on the CSIDH, named as CSIDH-PSEC-KEM and CSIDH-ECIES-KEM, respectively.

Motivated by the construction method of the famous SIKE [10] based on SIDH [16], this article presents a new secure and efficient post-quantum KEM based on the CSIDH [24], and the resulting new post-quantum KEM scheme is thereby named as CSIKE in this article. The newly presented CSIKE has much higher computation efficiency in the decapsulation part by involving an additional tag in the encapsulation part. The new post-quantum CSIKE from CSIDH is formally proved to be IND-CCA secure in the classic random oracle model. Moreover, the new CSIKE and the relevant KEM schemes CSIDH-PSEC-KEM and CSIDH-ECIES-KEM are compared from the theoretical viewpoint, and the comparison results show that the computation efficiency of the new CSIKE is comparable with the CSIDH-ECIES-KEM in both the encapsulation and decapsulation parts, while the new CSIKE has certain security advantage over the CSIDH-ECIES-KEM. Also, the computation efficiency of the new CSIKE is comparable with the CSIDH-PSEC-KEM in the encapsulation part, but is nearly 50% higher than that of CSIDH-PSEC-KEM in the decapsulation part. Therefore, the presented CSIKE in this work is a good alternative option to some extent.

1.2 Organization

Section 2 introduces some preliminaries, Section 3 presents the new post-quantum KEM scheme CSIKE from CSIDH key exchange protocol, and Section 4 formally proves that CSIKE is IND-CCA secure. Section 5 evaluates the CSIKE by comparing it with two existing relevant KEM schemes based on CSIDH from the theoretical viewpoint, then Section 6 concludes this article.

2 Preliminaries

First, some conventions are remarked here that throughout this article, "$\leftarrow_R$" means sampling a random value from a set, and the symbol ~ on a letter or string indicates a Montgomery elliptic curve defined over a

finite $\mathbb{F}_p$, e.g., $\tilde{a}$ denotes the Montgomery elliptic curve $y^2 = x^3 + ax^2 + x$, where $a \in \mathbb{F}_p$, "$*$" denotes the class group action, and $|\alpha|$ denotes the bit length of a variable α.

2.1 CSIDH

The original article introducing the CSIDH protocol is presented in ref. [24], and readers can refer to [24] for more details. Here, CSIDH is briefly reviewed as follows.

Setup. Let p be a large prime of the form $p = 4\ell_1\ell_2 \cdots \ell_n - 1$, where $\ell_i \quad (i = 1, 2, \ldots, n)$ are small distinct odd primes, fix a supersingular elliptic curve $E_0 : y^2 = x^3 + x$ defined over the finite field $\mathbb{F}_p$, with $\mathbb{F}_p$-rational endomorphism ring $\mathcal{O} = \mathbb{Z}[\pi]$. Let $\mathcal{ELL}(\mathcal{O})$ be the set of supersingular elliptic curves over $\mathbb{F}_p$ with the $\mathbb{F}_p$-rational endomorphism ring $\mathcal{O}$, and $\mathcal{G} = \mathrm{cl}(\mathcal{O})$ be the corresponding ideal class group. **Note**: $[\mathfrak{g}] \leftarrow_R \mathcal{G}$, *in the CSIDH setting*, means randomly sampling a vector $(e_1, e_2, \ldots, e_n)$ from the range $\{-M, \ldots, M\}$ and then setting $[\mathfrak{g}] = [\mathfrak{l}_1^{e_1} \cdots \mathfrak{l}_n^{e_n}] \in \mathrm{cl}(\mathcal{O})$, where M satisfies that $2M + 1 \geq \sqrt[n]{\#\mathrm{cl}(\mathcal{O})}$ and $\mathfrak{l}_i = (\ell, \pi - 1)$. Then, the public parameters of the original CSIDH [24] protocol consist of $param = \{p, E_0, \ell_i, M\}$.

Key exchange session. Assumed as usual the two communication parties are Alice and Bob, then the CSIDH key exchange session is performed as follows:

(i) Alice randomly generates a $[\mathfrak{a}] \leftarrow_R \mathcal{G}$ as her private key and computes the Montgomery elliptic curve $\widetilde{A} = [\mathfrak{a}] * E_0$: $y^2 = x^3 + Ax^2 + x$. Thus, the corresponding public key is A, and Alice sends A to Bob.

(ii) Bob randomly generates a $[\mathfrak{b}] \leftarrow_R \mathcal{G}$ as his private key and computes the Montgomery elliptic curve $\widetilde{B} = [\mathfrak{b}] * E_0$: $y^2 = x^3 + Bx^2 + x$. Thus, the corresponding public key is B, and Bob sends B to Alice.

(iii) On receiving B, Alice computes the shared secret S by $\widetilde{S} = [\mathfrak{a}] * \widetilde{B} = [\mathfrak{a}][\mathfrak{b}] * E_0$: $y^2 = x^3 + Sx^2 + x$.

(iv) On receiving A, Bob computes the shared secret S by $\widetilde{S} = [\mathfrak{b}] * \widetilde{A} = [\mathfrak{b}][\mathfrak{a}] * E_0$: $y^2 = x^3 + Sx^2 + x$.

Security assumption. Similar to the classic computational Diffie–Hellman (CDH) security assumption with respect to the traditional Diffie–Hellman protocol situation, the standard CSI-CDH security assumption from the CSIDH is described as follows.

Definition 1. (CSI-CDH assumption [42]). Given the parameters $param$, $\widetilde{A} = [\mathfrak{a}] * E_0$, and $\widetilde{B} = [\mathfrak{b}] * E_0$, the CSI-CDH problem is to compute $[\mathfrak{a}][\mathfrak{b}] * E_0$. The CSI-CDH assumption states that the advantage for any probabilistic polynomial time (PPT) adversary $\mathcal{A}$ to solve the CSI-CDH problem defined as follows:

$$\mathrm{Adv}_{param}^{\text{CSI-CDH}}(\mathcal{A}) = \Pr[\mathcal{A}(param, A, B) = [\mathfrak{a}][\mathfrak{b}] * E_0],$$

which is negligible.

2.2 Key encapsulation mechanism

Generally, a KEM consists of three algorithms, i.e., the key generation algorithm, the encapsulation algorithm and the decapsulation algorithm, which are typically denoted by **KG**$(\cdot)$, **Encap**$(\cdot)$ and **Decap**$(\cdot,\cdot)$, respectively. With the security parameter λ as input, the key generation algorithm **KG**(1^λ) outputs a public–private key pair (pk, sk). The encapsulation algorithm **Encap**(pk) generally returns a pair (c, K), where c is the encapsulation ciphertext of the secret encapsulation key K. The decapsulation algorithm **Decap**(sk, c) with the private key sk and ciphertext c as its input, returns the corresponding secret key K. The correctness of KEM ensures that **Decap**$(sk, c) = K$.

2.2.1 IND-CCA security of KEM

It has been widely accepted that a secure KEM scheme should achieve the encapsulation key indistinguishable against the chosen-ciphertext attack (IND-CCA). Formally, the IND-CCA security of a KEM scheme is defined by the following security experiment $\mathbf{Exp}^{\text{IND-CCA}}_{\text{KEM},\mathcal{A}}(\lambda)$,

---

$\underline{\mathbf{Exp}^{\text{IND-CCA}}_{\text{KEM},\mathcal{A}}(\lambda)}$:
$(pk, sk) \leftarrow \mathbf{KG}(1^\lambda)$;
$K_0^* \leftarrow_R$ Keyspace(λ); $(K_1^*, c^*) \leftarrow \mathbf{Encap}(pk)$;
$b \leftarrow_R \{0, 1\}$;
$b' \leftarrow \mathcal{A}^{\text{DECAP}(sk, c \neq c^*)}(pk, c^*, K_b^*)$;
if $b' = b$
  return 1;
else
  return 0;

---

where $\mathcal{A}$ is provided an access to the decapsulation oracle $\text{DECAP}(sk, \cdot)$ and gains the corresponding secret $\text{DECAP}(sk, \cdot)$ $K \leftarrow \text{DECAP}(sk, c)$ on input c with the restriction that $\mathcal{A}$ cannot query $\text{DECAP}(sk, \cdot)$ on the target ciphertext c^*. Then, the advantage of an adversary $\mathcal{A}$ against the security of the KEM is defined as follows:

$$\text{Adv}^{\text{IND-CCA}}_{\text{KEM}}(\mathcal{A}) = |\Pr[\mathbf{Exp}^{\text{IND-CCA}}_{\text{KEM},\mathcal{A}}(\lambda) = 1] - \frac{1}{2}|.$$

The corresponding KEM scheme is said to be IND-CCA secure if $\text{Adv}^{\text{IND-CCA}}_{\text{KEM}}(\mathcal{A})$ is negligible in the security parameter λ for any PPT adversary $\mathcal{A}$.

2.3 Key derivation function

A key derivation function KDF: $\mathcal{K}(\lambda) \rightarrow \{0, 1\}^{f(\lambda)}$ is required in the newly presented CSIKE, which can output a computational random number with variable length for any random input $u \in \mathcal{K}(\lambda)$. Its security is formally defined by a distinguished algorithm $\mathcal{D}$ as follows:

$$\text{Adv}^{\text{KDF}}_{\mathcal{D}}(\lambda) = |\Pr[\mathcal{D}(\text{KDF}(u \leftarrow_R \mathcal{K}(\lambda))) = 1] - \Pr[\mathcal{D}(K \leftarrow_R \{0, 1\}^{f(\lambda)}) = 1]|.$$

A cryptographic key derivation function KDF should satisfy that $\text{Adv}^{\text{KDF}}_{\mathcal{D}}(\lambda)$ is negligible for any PPT distinguisher $\mathcal{D}$.

3 The new post-quantum KEM: CSIKE

This section details the new post-quantum KEM scheme CSIKE. Like the now well-known SIKE, CSIKE uses the classic hashed ElGamal public-key encryption scheme instantiated in CSIDH setting (specifically denoted by PKE_C for easy to quote in this work, and $\text{PKE}_C = (\mathbf{KG}, \mathbf{Enc}, \mathbf{Dec})$ is depicted in Figure 1) as its building blocks, as done in ref. [41]. The IND-CPA security of PKE_C can be formally proved under the standard CSI-CDH security assumption with the completely analogous method to prove the classic hashed ElGamal public-key encryption scheme under the classic CDH security assumption. Thus, its concrete proof is omitted here, and refs. [43,44], can be referred to for the proof details.

Let $param = \{p, E_0, \ell_i, M\}$ be the same system parameters with the CSIDH [24], and λ be the system security parameter. Also, the cryptographic functions: $G(\cdot)$, $F(\cdot)$, $H(\cdot)$ are instantiated with the SHA-3

$\underline{\mathbf{KG}(1^\lambda):}$	$\underline{\mathbf{Enc}(\mathrm{pk}, m, [\mathfrak{r}] \leftarrow_R \mathcal{G}):}$	$\underline{\mathbf{Dec}(c, \mathrm{sk}):}$
$[\mathfrak{s}] \leftarrow_R \mathcal{G}$	$\widetilde{R} = [\mathfrak{r}] * E_0$	$\widetilde{S'} = [\mathfrak{s}] * \widetilde{R}$
$\widetilde{\mathrm{pk}} = [\mathfrak{s}] * E_0$	$\widetilde{S} = [\mathfrak{r}] * \widetilde{\mathrm{pk}}$	$m = c[2] \oplus F(S')$
Return $(\mathrm{pk}, \mathrm{sk} = [\mathfrak{s}])$	$c = (R, m \oplus F(S))$	

Figure 1: The ElGamal-type public-key encryption scheme PKE_C based on CSIDH.

derived function SHAKE256 [45] as done in SIKE [10], where it should be noted that $G : \{0,1\}^* \to \mathcal{G}$. Meanwhile, an additional key derivation function $\mathrm{KDF}:\{0,1\}^* \to \{0,1\}^{2\lambda}$ is used in the new CSIKE scheme, which is also instantiated with SHAKE256 for simplicity. As usual, CSIKE consists of three algorithms, i.e., the key generation algorithm **KG**$(\cdot)$, the encapsulation algorithm **Encap**$(\cdot)$ and the decapsulation algorithm **Decap**$(\cdot,\cdot)$, which are also specifically depicted in Figure 2 in addition to the following detailed descriptions.

- **KG**(1^λ): When inputting the security parameter λ, the key pair generation algorithm **KG**(1^λ) randomly generates $[\mathfrak{s}] \leftarrow_R \mathcal{G}$. It then computes the public key $\widetilde{\mathrm{pk}} = [\mathfrak{s}] * E_0$. Also, it randomly generates $s \leftarrow_R \{0,1\}^\lambda$ as a part of the full secret key. Finally, **KG** returns the public–private key pair $(\mathrm{pk}, \mathrm{sk} = (s, [\mathfrak{s}]))$.
- **Encap**(pk): When inputting the public key pk to the encapsulation algorithm **Encap**$(\cdot)$, **Encap** randomly generates $m \leftarrow_R \{0,1\}^\lambda$, which is then used along with the public key pk to compute the ephemeral secret $[\mathfrak{r}] = G(m, \mathrm{pk})$ by using the $G(\cdot)$. Then, **Encap** computes the ephemeral public key $\widetilde{R} = [\mathfrak{r}] * E_0$. Meanwhile, the shared secret S is computed by $\widetilde{S} = [\mathfrak{r}] * \widetilde{\mathrm{pk}}$, which is then used to encrypt m by $c[2] = m \oplus F(S)$. Then, the complete ciphertext is $c = (R, m \oplus F(S))$, where $c[1] = R$. Please note here that until now the aforementioned encapsulation steps are similar to SIKE, while the following steps are different from SIKE. Specifically, **Encap** proceeds to use the key derivation function KDF to compute the encapsulation key $K = (k_s, k_a) = \mathrm{KDF}(m, c)$, then computes the authentication tag $\tau = H(k_a, c)$. Finally, **Encap** returns (c, τ) as the encapsulation results and k_s as the shared secret key.
- **Decap**$((c, \tau), \mathrm{sk})$: When inputting the encapsulation results (c, τ) together with the private key sk to the decapsulation algorithm **Decap**$(\cdot,\cdot)$, **Decap** uses the partial secret key $[\mathfrak{s}]$ to compute the shared secret $\widetilde{S'} = [\mathfrak{s}] * \widetilde{R}$. Then, **Decap** decrypts the ciphertext c to get the secret m' by $m' = c[2] \oplus F(S')$. Similarly, please note here that until now the aforementioned decapsulation steps are similar to SIKE, while the following steps are different from SIKE. Specifically, **Decap** proceeds to compute $K = (k_s, k_a) = \mathrm{KDF}(m', c)$ and check whether $\tau = H(k_a, c)$ holds. If not, **Decap** recomputes $K = (k_s, k_a) = \mathrm{KDF}(s, c)$ by using the additional secret key s. Finally, **Decap** returns the shared secret key k_s.

Except for the depended isogeny graphs, the other main differences of the new CSIKE from SIKE have also been highlighted in Figure 2 using the gray background. Specifically, in the CSIKE, an additional

$\underline{\mathbf{KG}(1^\lambda):}$	$\underline{\mathbf{Encap}(\mathrm{pk}):}$	$\underline{\mathbf{Decap}((c, \tau), \mathrm{sk}):}$
$[\mathfrak{s}] \leftarrow_R \mathcal{G}$	$m \leftarrow_R \{0,1\}^\lambda$	$\widetilde{S'} = [\mathfrak{s}] * \widetilde{R}$
$\widetilde{\mathrm{pk}} = [\mathfrak{s}] * E_0$	$[\mathfrak{r}] = G(m, \mathrm{pk})$	$m' = c[2] \oplus F(S')$
$s \leftarrow_R \{0,1\}^\lambda$	$\widetilde{R} = [\mathfrak{r}] * E_0$	$(k_s, k_a) = KDF(m', c)$
Return $(\mathrm{pk}, \mathrm{sk} = (s, [\mathfrak{s}]))$	$\widetilde{S} = [\mathfrak{r}] * \widetilde{\mathrm{pk}}$	If $\tau \neq H(k_a, c)$
	$c = (R, m \oplus F(S))$	$(k_s, k_a) = KDF(s, c)$
	$(k_s, k_a) = KDF(m, c)$	Return k_s
	$\tau = H(k_a, c)$	
	Return $((c, \tau), k_s)$	

Figure 2: The presented supersingular isogeny-based key encapsulation mechanism CSIKE.

authentication tag τ is also output along with the corresponding encapsulation ciphertext, which exactly helps to avoid performing the partial re-encryption steps in the decapsulation algorithm, making the CSIKE reduce its class group action computation by one time in the decapsulation part.

Particularly, although the method of constructing the secure and efficient post-quantum CSIKE scheme from CSIDH in this study can also be applied in the SIDH setting, the resulting KEM scheme from SIDH will be no longer secure due to the Galbraith et al. [17] attack on supersingular isogeny cryptosystems. This finding was first pointed out by an anonymous reviewer, and thanks again for this anonymous reviewer.

4 Security proof

In this section, the presented CSIKE is formally proved in the random oracle model by tightly reducing its IND-CCA security to the IND-CPA security of the underlying public-key encryption scheme PKE_C, whose IND-CPA security proof under the standard CSI-CDH security assumption can be easily obtained by referring to refs [43,44].

Theorem 1. *Let $\mathcal{A}$ be a PPT adversary against the IND-CCA security of the new CSIKE and assume $\mathcal{A}$ performs at most q_D queries to the decapsulation oracle* Decap, *q_G, q_H and q_K queries to the random oracles G, H, and* KDF. *Then, it holds that*

$$\mathrm{Adv}^{\text{IND-CCA}}_{\text{CSIKE}}(\mathcal{A}) \leq \frac{2q_G + q_K + q_H + 1}{2^{\lambda}} + 2 \cdot \mathrm{Adv}^{\text{KDF}}_{\mathcal{D}}(\lambda) + 3 \cdot \mathrm{Adv}^{\text{IND-CPA}}_{\mathsf{PKE}_C}(\mathcal{B}),$$

where λ is the system security parameter, $\mathrm{Adv}^{\text{KDF}}_{\mathcal{D}}(\lambda)$ denotes the advantages of a distinguisher $\mathcal{D}$ against the key derivation function KDF *and $\mathrm{Adv}^{\text{IND-CPA}}_{\mathsf{PKE}_C}(\mathcal{B})$ denotes the advantages of an adversary $\mathcal{B}$ against the IND-CPA security of the underlying public-key encryption scheme PKE_C.*

Proof. The proof is similar to the proof provided in ref. [41], which is proceeded by incrementally defining a sequence of games from the original IND-CCA security game G_0 of KEM to the end game G_3. Also, denote by $\mathsf{G}_i^{\mathcal{A}} \Rightarrow 1$ the event that the corresponding game G_i ($i = 0, 1, 2, 3$) outputs 1, i.e., $\mathcal{A}$ correctly guesses the random bit b in each game G_i.

Game G_0: This game corresponds to the original IND-CCA security game of KEM, so according to the definition, it holds that

$$\mathrm{Adv}^{\text{IND-CCA}}_{\text{CSIKE}}(\mathcal{A}) = \left| \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \frac{1}{2} \right|. \tag{1}$$

Game G_1: In this game, the decapsulation oracle Decap and random oracles are simulated as usual but with the modifications that (i) for the decapsulation oracle Decap, the special case $(k_s, k_a) = \mathrm{KDF}(s, c)$ is replaced with $(k_s, k_a) \leftarrow_R \{0,1\}^{2\lambda}$; (ii) for the random oracle $\mathrm{KDF}(m, c)$, if $m = s$, then abort this game. Then, it holds that

$$|\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1]| \leq \mathrm{Adv}^{\text{KDF}}_{\mathcal{D}}(\lambda) + \frac{q_K}{2^{\lambda}}. \tag{2}$$

Game G_2: In this game, the decapsulation oracle Decap is modified so that the secret private key is no longer used in the decapsulation operations. To this end, the random oracle KDF is also modified accordingly to cope with the modification of the Decap oracle. Specifically, the modified Decap oracle and random oracles are depicted in Figure 3. Then, it can be easily seen that G_2 and G_1 are indistinguishable from the view of the adversary $\mathcal{A}$ under the security assumption of the key derivation function if the event E_1 does not occur. Thus, according to the security assumption of the KDF and the difference lemma [46], it holds that

$$|\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]| \leq \mathrm{Adv}^{\text{KDF}}_{\mathcal{D}}(\lambda) + \frac{q_H}{2^{\lambda}}. \tag{3}$$

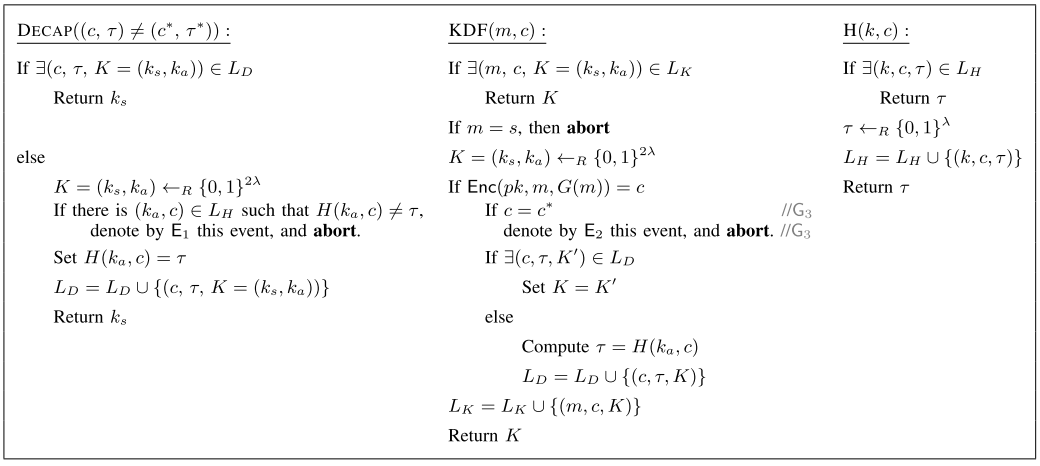

Figure 3: Oracle queries in games G_2 and G_3 for the theorem proof, where Enc is the encryption algorithm in PKE_C.

Game G_3: In this game, the event E_2 also depicted in Figure 3 is considered, and this game is immediately aborted if E_2 occurs. Then, according to the difference lemma, it holds that

$$|\Pr[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]| \leq \Pr[\mathsf{E}_2]. \tag{4}$$

In game G_3, it is obvious that $\mathcal{A}$ has no access to the $K^* = \mathrm{KDF}(m^*, c^*)$, which implies that

$$\Pr[\mathsf{G}_3^{\mathcal{A}} \Rightarrow 1] = \frac{1}{2}. \tag{5}$$

Thus, it just needs to bound $\Pr[\mathsf{E}_2]$ to complete this proof. To this end, as mentioned in ref. [41], an adversary C against the transformed public-key encryption scheme $\mathsf{PKE}_C^{\mathsf{T}}$'s one-wayness under the plaintext checking attacks (OW-PCA), is constructed by using the event E_2 as depicted in Figure 4. It should be noted here that $\mathsf{PKE}_C^{\mathsf{T}}$ is transformed from the underlying public-key encryption scheme PKE_C by using the specific T transformation [41], i.e., $\mathsf{PKE}_C^{\mathsf{T}} = \mathsf{T}[\mathsf{PKE}_C, G]$. Specifically, in $\mathsf{PKE}_C^{\mathsf{T}}$, the encryption is performed by $c = \mathsf{Enc}(pk, m, G(m))$ using G to generate ephemeral [r], while in the decryption algorithm the decrypted $m' = \mathsf{Dec}(sk, c)$ is output only when the checking test $\mathsf{Enc}(pk, m', G(m')) = c$ is passed. The simulation of C is perfect. The event E_2 implies that $\mathcal{A}$ has queried (m^*, c^*) to the random oracle KDF, so C can return the correct m^*. Thus, it obviously holds that

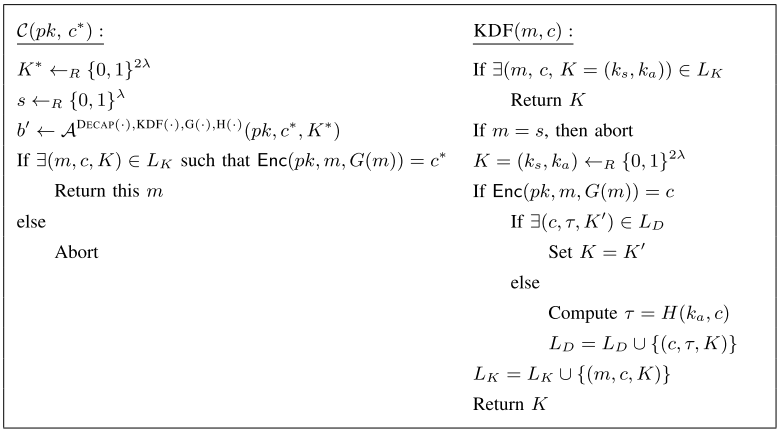

Figure 4: Adversary C against the OW-PCA security of the public-key encryption scheme $\mathsf{PKE}_C^{\mathsf{T}} = \mathsf{T}[\mathsf{PKE}_C, G]$.

Table 1: Comparison of the CSIKE and relevant KEM schemes from theoretical view, where isogen denotes isogeny computation

Scheme	Model	Assumption	Encapsulation size	Approximate computation costs	
				Encap	Decap
CSIDH-PSEC-KEM [42]	QROM	CSI-DDH	$\|p\| + \lambda$	2 isogen	2 isogen
CSIDH-ECIES-KEM [42]	ROM	CSI-GDH	$\|p\|$	2 isogen	1 isogen
CSIKE	ROM	CSI-CDH	$\|p\| + \lambda + \|\tau\|$	2 isogen	1 isogen

$$\Pr[\mathsf{E}_2] \leq \mathrm{Adv}^{\text{OW-PCA}}_{\mathsf{PKE}^{\mathsf{T}}_{C}}(\mathcal{C}). \tag{6}$$

Besides, according to Theorem 2 in [41], we have

$$\mathrm{Adv}^{\text{OW-PCA}}_{\mathsf{PKE}^{\mathsf{T}}_{C}}(\mathcal{C}) \leq \frac{2q_G + 1}{2^{\lambda}} + 3 \cdot \mathrm{Adv}^{\text{IND-CPA}}_{\mathsf{PKE}_{C}}(\mathcal{B}). \tag{7}$$

Then, collecting the inequalities (1)–(7), the theorem can be easily concluded. □

5 Comparison

Compared with the only existing two CSIDH-based KEM schemes CSIDH-PSEC-KEM and CSIKE-ECIES-KEM, both presented in ref. [42], the encapsulation results in the new CSIKE involve an additional tag τ, whose size is equal to that of the hash output. Therefore, CSIKE's encapsulation size ($|p| + \lambda + |\tau|$) is slightly longer than that of the CSIDH-PSEC-KEM and CSIKE-ECIES-KEM whose encapsulation size are $|p| + \lambda$ and $|p|$, respectively. Exactly due to the involvement of this tag τ, the decapsulation algorithm avoids performing an additional isogeny computation for validity checking, which makes the new CSIKE have much higher computation efficiency than the CSIDH-PSEC-KEM in the decapsulation part since the isogeny computation is the most time-consumed operation in the isogeny-based cryptography. In fact, compared with the isogeny computation, other computation costs in these KEM schemes from CSIDH can be roughly omitted; thus, an approximate comparison of the CSIKE with the CSIDH-PSEC-KEM and CSIKE-ECIES-KEM from theoretical view by just considering the isogeny computation operation is summarized in Table 1. According to Table 1, it can be easily seen that the computation efficiency of the new CSIKE is better than the CSIDH-PSEC-KEM by improvement of approximate 50% in decapsulation part and is comparable with the CSIKE-ECIES-KEM in both encapsulation and decapsulation parts. In addition, the security model and assumption used to give formal security proof for these three KEM schemes are also summarized in Table 1, from which we can see that in the random oracle model, the stronger CSI-GDH assumption is required for proving CSIDH-ECIES-KEM, while only the basic CSI-CDH assumption is required for proving the new CSIKE. Therefore, it can be said that the presented CSIKE has a certain advantage over CSIDH-PSEC-KEM in computation efficiency and CSIKE-ECIES-KEM in the security aspect.

6 Conclusion

This article presents a new supersingular isogeny-based key encapsulation mechanism from the CSIDH key exchange protocol, and thus is named as CSIKE. The IND-CCA security of the presented CSIKE is tightly reduced in the random oracle model to the IND-CPA security of the underlying CSIDH-based hashed ElGamal public-key encryption scheme, whose IND-CPA security can be easily formally proved. Therefore, it can be said that this article has provided a new secure and practical post-quantum KEM scheme from CSIDH for resisting the possible quantum computer attacks. Although the security proof in the quantum random oracle

for CSIKE should be provided to fully explain its quantum-resistance, which is left as a future work. Moreover, how to efficiently implement the newly presented CSIKE is another future study work.

Acknowledgments: The author would like to express the deepest thanks to the editor and anonymous reviewers for their valuable comments and work for this article. This work was supported in part by the Natural Science Basic Research Program of Shaanxi Province of China under Grant 2021JQ-123 and in part by the Fundamental Research Funds for the Central Universities (No. 31020200QD011).

Conflict of interest: Authors state no conflict of interest.

References

[1] Cramer R, Shoup V. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM J Comput. 2003;33(1):167–226.
[2] Kurosawa K, Desmedt Y. A new paradigm of hybrid encryption scheme. In: Franklin M, editor. Advances in Cryptology - CRYPTO 2004. Berlin, Heidelberg: Springer; 2004. p. 426–42.
[3] Kiltz E. Chosen-Ciphertext secure key-encapsulation based on gap hashed Diffie–Hellman. In: Okamoto T, Wang X, editors. Public Key Cryptography - PKC 2007. Berlin, Heidelberg: Springer; 2007. p. 282–97.
[4] Shoup V. ISO/IEC 18033-2: 2006: Information technology-security techniques-encryption algorithms-part 2: Asymmetric ciphers. International Organization for Standardization, Geneva, Switzerland. 2006. p. 44.
[5] Dierks T, Rescorla E. The transport layer security (TLS) protocol version 1.2. 2008.
[6] Krawczyk H, Paterson KG, Wee H. On the security of the TLS protocol: A systematic analysis. In: Annual Cryptology Conference. Springer; 2013. p. 429–48.
[7] Arute F, Arya K, Babbush R, Bacon D, Bardin JC, Barends R, et al. Quantum supremacy using a programmable superconducting processor. Nature. 2019;574(7779):505–10.
[8] Gambetta J. IBM's Roadmap For Scaling Quantum Technology; September 15, 2020. https://www.ibm.com/blogs/research/2020/09/ibm-quantum-roadmap/.
[9] Baldi M, Barenghi A, Chiaraluce F, Pelosi G, Santini P. LEDAkem: A post-quantum key encapsulation mechanism based on QC-LDPC codes. In: International Conference on Post-Quantum Cryptography. Springer; 2018. p. 3–24.
[10] Jao D, Azarderakhsh R, Campagna M, Costello C, DeFeo L, Hess B, et al. SIKE: Supersingular isogeny key encapsulation. Submission to the NIST standardization process on post-quantum cryptography. 2017.
[11] Kuznetsov A, Lutsenko M, Kiian N, Makushenko T, Kuznetsova T. Code-based key encapsulation mechanisms for post-quantum standardization. In: 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT). IEEE; 2018. p. 276–81.
[12] Maram V. On the security of NTS-KEM in the quantum random oracle model; 2020. https://eprint.iacr.org/2020/150. Cryptology ePrint Archive, Report 2020/150.
[13] Couveignes JM. Hard homogeneous spaces; 2006. https://eprint.iacr.org/2006/291. Cryptology ePrint Archive, Report 2006/291.
[14] Rostovtsev A, Stolbunov A. Public-key cryptosystem based on isogenies; 2006. http://eprint.iacr.org/2006/145. Cryptology ePrint Archive, Report 2006/145. Available from: http://eprint.iacr.org/2006/145/.
[15] Childs A, Jao D, Soukharev V. Constructing elliptic curve isogenies in quantum subexponential time. J Math Cryptol. 2014;8(1):1–29.
[16] Jao D, De Feo L. Towards quantum resistant cryptosystems from supersingular elliptic curve isogenies. In: International Workshop on Post-Quantum Cryptography. Springer; 2011. p. 19–34.
[17] Galbraith SD, Petit C, Shani B, Ti YB. On the security of supersingular isogeny cryptosystems. In: Advances in Cryptology-ASIACRYPT 2016: 22nd International Conference on the Theory and Application of Cryptology and Information Security, December 4–8, 2016, Proceedings, Part I 22. Hanoi, Vietnam: Springer; 2016. p. 63–91.
[18] Costello C, Longa P, Naehrig M. Efficient algorithms for supersingular isogeny Diffie–Hellman. In: Robshaw M, Katz J, editors. Advances in Cryptology - CRYPTO 2016: 36th Annual International Cryptology Conference. Berlin Heidelberg: Springer; 2016. p. 572–601. doi: 10.1007/978-3-662-53018-4_21.
[19] Koziel B, Azarderakhsh R, Kermani MM. A high-performance and scalable hardware architecture for Isogeny-based cryptography. IEEE Trans Comput. 2018;67(11):1594–609.
[20] Azarderakhsh R, Jao D, Kalach K, Koziel B, Leonardi C. Key compression for isogeny-based cryptosystems. In: Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography. ACM; 2016. p. 1–10.
[21] Costello C, Jao D, Longa P, Naehrig M, Renes J, Urbanik D. Efficient compression of SIDH public keys. Cham: Springer International Publishing; 2017. p. 679–706. doi: 10.1007/978-3-319-56620-7_24.

[22] Galbraith SD, Petit C, Silva J. Identification protocols and signature schemes based on supersingular isogeny problems. In: International Conference on the Theory and Application of Cryptology and Information Security; 2017. p. 3–33.
[23] De Feo L, Kieffer J, Smith B. Towards practical key exchange from ordinary isogeny graphs. In: International Conference on the Theory and Application of Cryptology and Information Security. Springer; 2018. p. 365–94.
[24] Castryck W, Lange T, Martindale C, Panny L, Renes J. CSIDH: an efficient post-quantum commutative group action. In: Peyrin T, Galbraith S, editors. Advances in Cryptology - ASIACRYPT 2018. Cham: Springer International Publishing; 2018. p. 395–427.
[25] Meyer M, Reith S. A faster way to the CSIDH. In: International Conference on Cryptology in India. Springer; 2018. p. 137–52.
[26] Bonnetain X, Schrottenloher A. Quantum security analysis of CSIDH. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer; 2020. p. 493–522.
[27] Peikert C. He gives C-sieves on the CSIDH. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer; 2020. p. 463–92.
[28] Cramer R, Shoup V. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: Annual International Cryptology Conference; 1998. p. 13–25.
[29] Shoup V. Using Hash functions as a hedge against chosen Ciphertext attack. In: Preneel B, editor.Advances in Cryptology – EUROCRYPT 2000. Berlin, Heidelberg: Springer Berlin Heidelberg; 2000. p. 275–88.
[30] Herranz J, Hofheinz D, Kiltz E. The Kurosawa-Desmedt key encapsulation is not Chosen-Ciphertext secure; 2006. https://eprint.iacr.org/2006/207. Cryptology ePrint Archive, Report 2006/207.
[31] Choi SG, Herranz J, Hofheinz D, Hwang JY, Kiltz E, Lee DH, et al. The Kurosawa-Desmedt key encapsulation is not chosen-ciphertext secure. Inform Process Lett. 2009;109(16):897–901.
[32] Kurosawa K, Le Phong T. Kurosawa-Desmedt key encapsulation mechanism, revisited. In: International Conference on Cryptology in Africa. Springer; 2014. p. 51–68.
[33] Fujisaki E, Okamoto T. Secure integration of asymmetric and symmetric encryption schemes. In: Annual International Cryptology Conference. Springer; 1999. p. 537–54.
[34] Okamoto T, Pointcheval D. REACT: Rapid enhanced-security asymmetric cryptosystem transform. In: Cryptographers Track at the RSA Conference. Springer; 2001. p. 159–74.
[35] Coron JS, Handschuh H, Joye M, Paillier P, Pointcheval D, Tymen C. GEM: A generic chosen-ciphertext secure encryption method. In: Cryptographers Track at the RSA Conference. Springer; 2002. p. 263–76.
[36] Shor PW. Algorithms for quantum computation: Discrete logarithms and factoring. In: Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on. IEEE; 1994. p. 124–34.
[37] Bos J, Ducas L, Kiltz E, Lepoint T, Lyubashevsky V, Schanck JM, et al. CRYSTALS-Kyber: a CCA-secure module-lattice-based KEM. In:2018 IEEE European Symposium on Security and Privacy (EuroS&P). IEEE; 2018. p. 353–67.
[38] Schanck JM, Hulsing A, Rijneveld J, Schwabe P. Technical report, National Institute of Standards and Technology, 2017; 2017.
[39] Announcing Request for Nominations for Public-Key Post-Quantum Cryptographic Algorithms; https://csrc.nist.gov/news/2016/public-key-post-quantum-cryptographic-algorithms.
[40] Alagic G, Alperin-Sheriff J, Apon D, Cooper D, Dang Q, Liu YK, et al. Status report on the second round of the NIST postquantum cryptography standardization process. National Institute of Standards and Technology; 2020.
[41] Hofheinz D, Hövelmanns K, Kiltz E. A modular analysis of the Fujisaki-Okamoto transformation. In: Theory of Cryptography Conference. Springer; 2017. p. 341–71.
[42] Yoneyama K. Post-quantum variants of ISO/IEC standards: compact chosen Ciphertext secure key encapsulation mechanism from isogeny. In: Proceedings of the 5th ACM Workshop on Security Standardisation Research Workshop; 2019. p. 13–21.
[43] Kiltz E, Malone-Lee J. A general construction of IND-CCA2 secure public key encryption. In: IMA International Conference on Cryptography and Coding. Springer; 2003. p. 152–66.
[44] Katz J, Lindell Y. Introduction to modern cryptography. CRC Press; 2014.
[45] Dworkin MJ. SHA-3 standard: Permutation-based hash and extendable-output functions; August 4, 2015. https://csrc.nist.gov/publications/detail/fips/202/final.
[46] Shoup V. OAEP reconsidered. In: Annual International Cryptology Conference. Springer; 2001. p. 239–59.