Anonymity of NIST PQC Round 3 KEMs*

Keita Xagawa

NTT Social Informatics Laboratories, keita.xagawa.zv@hco.ntt.co.jp

Abstract. This paper investigates *anonymity* of all NIST PQC Round 3 KEMs: Classic McEliece, Kyber, NTRU, Saber, BIKE, FrodoKEM, HQC, NTRU Prime (Streamlined NTRU Prime and NTRU LPRime), and SIKE. We show the following results:

- NTRU is anonymous in the quantum random oracle model (QROM) if the underlying deterministic PKE is strongly disjoint-simulatable. NTRU is collision-free in the QROM. A hybrid PKE scheme constructed from NTRU as KEM and appropriate DEM is anonymous and robust. (Similar results for BIKE, FrodoKEM, HQC, NTRU LPRime, and SIKE hold except for two of three parameter sets of HQC.)
- Classic McEliece is anonymous in the QROM if the underlying PKE is strongly disjoint-simulatable and a hybrid PKE scheme constructed from it as KEM and appropriate DEM is anonymous.
- Grubbs, Maram, and Paterson pointed out that Kyber and Saber have a gap in the current IND-CCA security proof in the QROM (EUROCRYPT 2022). We found that Streamlined NTRU Prime has another technical obstacle for the IND-CCA security proof in the QROM.

Those answer the open problem to investigate the anonymity and robustness of NIST PQC Round 3 KEMs posed by Grubbs, Maram, and Paterson (EUROCRYPT 2022).
We use strong disjoint-simulatability of the underlying PKE of KEM and strong pseudorandomness and smoothness/sparseness of KEM as the main tools, which will be of independent interest.

Keywords: anonymity, robustness, post-quantum cryptography, NIST PQC standardization, KEM, PKE, quantum random model

1 Introduction

Public-key encryption (PKE) allows us to send a message to a receiver confidentially if the receiver's public key is available. However, a ciphertext of PKE may reveal the receiver's public key, and the recipient of the ciphertext will be identified. This causes trouble in some applications, and researchers study the anonymity of PKE. Roughly speaking, PKE is said to be *anonymous* [BBDP01] if a ciphertext hides the receiver's information. Anonymous primitive is often used in the context of privacy-enhancing technologies.
A ciphertext of anonymous PKE indicates (computationally) no information of a receiver. Thus, when a receiver receives a ciphertext, it should decrypt the ciphertext into a message and verify the message in order to check if the ciphertext is sent to the receiver or not. There may be a ciphertext from which two (or more) recipients can obtain messages in this situation, and this causes trouble in some applications, e.g., auction protocols [Sak00]. Intuitively speaking, PKE is said to be *robust* [ABN10] if only the intended receiver can obtain a meaningful message from a ciphertext.
Both anonymity and robustness are important and useful properties beyond the standard IND-CCA security. Anonymous PKE is an important building primitive for anonymous credential systems [CL01], auction protocols [Sak00], (weakly) anonymous authenticated key exchange [BCGNP09, FSXY13, FSXY15, SSW20], and so on. Robust PKE has an application for searchable encryption [ABC+05] and auction [Sak00].

Previous works on anonymity and robustness of KEM and hybrid PKE: Mohassel [Moh10] studied the anonymity and robustness of a special KEM/DEM framework, a hybrid PKE with KEM that is implemented by a PKE with random plaintext. He showed that even if anonymous KEM and DEM sometimes fail to lead to an anonymous hybrid PKE by constructing a counterexample.
Grubbs, Maram, and Paterson [GMP21a] discussed anonymity and robustness of *post-quantum* KEM schemes and KEM/DEM framework in the quantum random oracle model (QROM). They also studied the anonymity and robustness of the hybrid PKE based on KEM with implicit rejection. On the variants of the Fujisaki-Okamoto (FO) transform [FO99, FO13], they showed that anonymity and collision-freeness of KEMs obtained by the FO transform with implicit rejection and its variant[1], and they lead to anonymous, robust hybrid PKEs

* This article is based on an earlier article: Keita Xagawa: Anonymity of NIST PQC Round 3 KEMs, EUROCRYPT 2022, © IACR 2022

[1] A variant of the FO transform with implicit rejection using 'pre-key' technique. They wrote "a variant of the $\mathsf{FO}^{\not\perp}$ transform" in their paper.

Table 1. Summary of anonymity and robustness of NIST PQC Round 3 KEM candidates (finalists and alternate candidates) and the hybrid PKEs using them. In the first row, IND = Indistinguishability, SPR = Strong Pseudorandomness, ANO = Anonymity, CF = Collision Freeness, and ROB = Robustness under chosen-ciphertext attacks in the QROM. Y = Yes, N = No, ? = Unknown. The underline implies our new findings.

	KEM					Hybrid PKE		
Name	IND	SPR	ANO	CF	ROB	ANO	ROB	
Classic McEliece [ABC+20]	Y	Y	Y	N	N	Y	N	Section K
Kyber [SAB+20]	?	?	?	?	N	?	?	Section L
NTRU [CDH+20]	Y	Y	Y	Y	N	Y	Y	Section 5
Saber [DKR+20]	?	?	?	?	N	?	?	Section M
BIKE [ABB+20]	Y	Y	Y	Y	N	Y	Y	Section N
FrodoKEM [NAB+20]	Y	Y	Y	Y	N	Y	Y	Section O
HQC-192 [AAB+20]	Y	Y	Y	Y	Y	Y	Y	Section P
HQC-128/256 [AAB+20]	Y	N	N	Y	Y	N	Y	Section P
Streamlined NTRU Prime [BBC+20]	?	?	?	?	N	?	?	Section Q
NTRU LPRime [BBC+20]	Y	Y	Y	Y	N	Y	Y	Section R
SIKE [JAC+20]	Y	Y	Y	Y	N	Y	Y	Section S

from appropriate assumptions. They also showed anonymity and robustness of KEM obtained by a variant of the FO transform with explicit rejection and key-confirmation hash[2] and showed that it leads to anonymous, robust hybrid PKE from appropriate assumptions.
They examined NIST PQC Standardization finalists (Classic McEliece [ABC+20], Kyber [SAB+20], NTRU [CDH+20], and Saber [DKR+20]). They showed the following results:

- Classic McEliece: They found that Classic McEliece is not collision-free. Since their anonymity proof in [GMP21a, Theorem 5] strongly depends on the collision-freeness of the underlying PKE, we cannot apply their anonymity proof to Classic McEliece. They also showed that the hybrid PKE fails to achieve robustness since Classic McEliece is not collision-free.
- Kyber: They found that Kyber's anonymity (and even IND-CCA security) has two technical obstacles ('pre-key' and 'nested random oracles') in the QROM.
- NTRU: NTRU's anonymity has another technical obstacle: Their proof technique requires the computation of a key of KEM involving a message and a ciphertext, but, in NTRU, the computation of a key of NTRU involves only a message. The robustness of the hybrid PKE with NTRU is unclear.
- Saber: They insisted that they show Saber's anonymity and IND-CCA security and the robustness of the hybrid PKE with Saber in the QROM, because they considered that Saber employs the FO transform with 'pre-key'. Unfortunately, Saber in [DKR+20] also uses both 'pre-key' and 'nested random oracles' as Kyber, and their proofs cannot be applied to Saber. See their slides [GMP21b]. (Fortunately, FrodoKEM can be shown anonymous and lead to anonymous, robust hybrid PKE, because FrodoKEM employs the FO transform with 'pre-key'.)

Unfortunately, we do not know whether all four finalists are anonymous or not, although the effort of Grubbs et al. and their clean and modular framework. Grubbs et al. left several open problems: One of them is the anonymity and robustness of NTRU; the other important one is the anonymity of Classic McEliece.

1.1 Our Contribution

We investigate anonymity and robustness of *all* NIST PQC Round 3 KEM candidates and obtain Table 1. This answers the open problems posed by Grubbs et al.
In order to investigate anonymity, we first study strong pseudorandomness of PKE/KEM instead of studying anonymity directly. To show strong pseudorandomness of the hybrid PKE, we study strong pseudorandomness and introduce smoothness and sparseness of KEM. We then show such properties of KEM obtained by the variants of the FO transform if the underlying deterministic PKE is strongly disjoint-simulatable. We finally study the properties of NIST PQC Round 3 KEM candidates. See the details in the following.

Anonymity through strong pseudorandomness, sparseness, and smoothness: Our starting point is *strong pseudorandomness* instead of anonymity. We say PKE/KEM/DEM is *strongly pseudorandom* if its ciphertext is

[2] They modify 'key-confirmation hash' to involve a ciphertext on input.

indistinguishable from a random string chosen by a simulator on input the security parameter.[3] It is easy to show that strong pseudorandomness implies anonymity.
Using this notion, we attempt to follow the IND-CCA security proof of the KEM/DEM framework [CS02], that is, we try to show that the hybrid PKE from strongly pseudorandom KEM/DEM is also strongly pseudorandom, which implies that the hybrid PKE is anonymous. If we directly try to prove anonymity against chosen-ciphertext attacks (ANON-CCA security) of the hybrid PKE, then we will need to simulate *two* decryption oracles as Grubbs et al. Considering pseudorandomness allows us to treat a *single* key and oracle and simplifies the security proof. Unfortunately, we face another obstacle in the security proof when considering pseudorandomness.
To resolve the obstacle, we define *sparseness* of KEM with explicit rejection and *smoothness* of KEM with implicit rejection: We say KEM with explicit rejection is *sparse* if a ciphertext c chosen by a simulator is decapsulated into $\perp$ with overwhelming probability. We say KEM with implicit rejection is *smooth* if, given a ciphertext c chosen by a simulator, any efficient adversary cannot distinguish a random key from a decapsulated key. This definition imitates the smoothness of the hash proof system [CS02]. Those notions help us to prove the pseudorandomness of the hybrid PKE.

Pseudorandomness, smoothness, and collision-freeness of the FO variants: In order to treat the case for Classic McEliece and NTRU, in which the underlying PKE is deterministic, we treat SXY [SXY18], variants of U [HHK17], and variants of HU [JZM19]. Modifying the IND-CCA security proofs of them, we show that the obtained KEM is strongly pseudorandom and smooth if the underlying PKE is strongly disjoint-simulatable [SXY18]. We also show that the obtained KEM is collision-free if the underlying deterministic PKE is collision-free. We finally note that our reductions are *tight* as a bonus.
Grubbs et al. [GMP21a] discussed a barrier to show anonymity of NTRU (and Classic McEliece implicitly), which stems from the design choice $K = \mathsf{H}(\mu)$ instead of $K = \mathsf{H}(\mu, c)$, where μ is a plaintext and c is a ciphertext. In addition, their proof technique requires the underlying PKE to be collision-free. Since the underlying PKE of Classic McEliece lacks collision freeness, they left the proof of anonymity of Classic McEliece as an open problem. Both barriers stem from the fact that we need to simulate *two* decapsulation oracles in the proof of ANON-CCA-security. We avoid those technical barriers by using a stronger notion, strong pseudorandomness against chosen-ciphertext attacks (SPR-CCA security); in the proof of SPR-CCA-security, we only need to simulate a *single* decapsulation oracle.

Application to NIST PQC Round 3 KEM candidates: Using the above techniques, we solve open problems posed by Grubbs et al. and extend the study of finalists and alternative candidates of NIST PQC Round 3 KEMs as depicted in Table 1.
We found the following properties (we omit the detail of the assumptions):

- Classic McEliece is anonymous and the hybrid PKE using it is anonymous, which is in the full version.
- NTRU is anonymous and collision-free. The hybrid PKE using it is anonymous and robust. See Section 5. Similar results for BIKE, HQC (HQC-196)[4], NTRU LPRime, and SIKE hold.
- We found that Streamlined NTRU Prime has another technical obstacle to anonymity: the key and key-confirmation hash involve a 'pre-key' problem.[5] While this is not a big problem for the IND-CCA security in the ROM, we fail to show the IND-CCA security in the QROM. See Section Q for the discussion.

Remark 1.1. Bernstein [Ber21] suggests to use *quantum indifferentiability* of the domain extension of quantum random oracles in [Zha19, Section 5]. While we did not check the detail, this quantum indifferentiability would solve the problems on 'pre-key' of Kyber, Saber, and Streamlined NTRU Prime.

Open problems: We leave showing anonymity and the IND-CCA security of Kyber, Saber, and Streamlined NTRU Prime in the QROM as an important open problem as Grubbs et al. posed.

Organization: Section 2 reviews the QROM, definitions of primitives, and the results of Grubbs et al. [GMP21a]. In addition, it also shows strong pseudorandomness implies anonymity. Section 3 studies the strong pseudorandomness of the KEM/DEM framework. Section 4 studies SXY's security properties. Section 5 examines the anonymity and robustness of NTRU.

[3] If the simulator can depend on an encryption key, then we just say pseudorandom.
[4] HQC-128/256 is not anonymous because the parity of the ciphertext leaks the parity of the encapsulation key. See Section P for the detail.
[5] The key and key-confirmation value on a plaintext μ and an encapsulation key ek is computed as $K = \mathsf{H}(k, c_0, c_1)$ and $h = \mathsf{F}(k, \mathsf{Hash}(ek))$, where $k = \mathsf{H}_3(\mu)$ and (c_0, c_1) is a main body of a ciphertext.

Appendix highlights: The appendices contain the properties of the variants of the FO transform, those for T in Section D, those for a variant of U in Section E, and those for variants of HU in Section F, Section G, Section H, Section I, and Section J. The appendices examine the other NIST PQC Round 3 KEM candidates, Classic McEliece in Section K, Kyber in Section L, Saber in Section M, BIKE in Section N, FrodoKEM in Section O, HQC in Section P, NTRU Prime (Streamlined NTRU Prime in Section Q and NTRU LPRime in Section R), and SIKE in Section S, as summarized in Table 1.

Version notes: This is the 2022-09-22 version and we correct mistakes on HQC; HQC-192 is anonymous, but HQC-128 and HQC-256 are not.

2 Preliminaries

Notations: A security parameter is denoted by κ. We use the standard O-notations. DPT, PPT, and QPT stand for deterministic polynomial time, probabilistic polynomial time, and quantum polynomial time, respectively. A function $f(\kappa)$ is said to be *negligible* if $f(\kappa) = \kappa^{-\omega(1)}$. We denote a set of negligible functions by $\mathrm{negl}(\kappa)$. For a distribution χ, we often write "$x \leftarrow \chi$," which indicates that we take a sample x according to χ. For a finite set S, $U(S)$ denotes the uniform distribution over S. We often write "$x \leftarrow S$" instead of "$x \leftarrow U(S)$." For a set S and a deterministic algorithm A, $\mathsf{A}(S)$ denotes the set $\{\mathsf{A}(x) \mid x \in S\}$. If inp is a string, then "$\mathsf{out} \leftarrow \mathsf{A}(\mathsf{inp})$" denotes the output of algorithm A when run on input inp. If A is deterministic, then out is a fixed value and we write "$\mathsf{out} := \mathsf{A}(\mathsf{inp})$." We also use the notation "$\mathsf{out} := \mathsf{A}(\mathsf{inp}; r)$" to make the randomness r explicit.
For a statement P (e.g., $r \in [0,1]$), we define $\mathrm{boole}(P) = 1$ if P is satisfied and 0 otherwise.
For two finite sets $\mathcal{X}$ and $\mathcal{Y}$, $\mathcal{F}(\mathcal{X}, \mathcal{Y})$ denotes a set of all mappings from $\mathcal{X}$ to $\mathcal{Y}$.

Lemma 2.1 (Generic distinguishing problem with bounded probabilities [HKSU20, Lemma 2.9], adapted). *Let $\mathcal{X}$ be a finite set. Let $\delta \in [0,1]$. Let $\mathsf{F}\colon \mathcal{X} \to \{0,1\}$ be the following function: for each $x \in \mathcal{X}$, $\mathsf{F}(x) = 1$ with probability $\delta_x \le \delta$ and $\mathsf{F}(x) = 0$ else. Let $\mathsf{Z}\colon \mathcal{X} \to \{0,1\}$ be the zero function, that is, $\mathsf{Z}(x) = 0$ for all x. If an unbounded-time quantum adversary $\mathcal{A}$ makes queries to F or Z at most Q times, then we have*

$$\left|\Pr[b \leftarrow \mathcal{A}^{\mathsf{F}(\cdot)}() : b = 1] - \Pr[b \leftarrow \mathcal{A}^{\mathsf{Z}(\cdot)}() : b = 1]\right| \le 8(Q+1)^2\delta.$$

where all oracle accesses of $\mathcal{A}$ can be quantum.

Quantum random oracle model: Roughly speaking, the quantum random oracle model (QROM) is an idealized model where a hash function is modeled as a publicly and quantumly accessible random oracle. In this paper, we model a quantum oracle $O\colon \{0,1\}^n \to \{0,1\}^m$ as a mapping $|x\rangle\,|y\rangle \mapsto |x\rangle\,|y \oplus O(x)\rangle$, where $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$. See [BDF+11] for a more detailed description of the model.
We review some useful lemmas for the properties of the quantum random oracle (QRO). The first one states that QRO is PRF. See [SXY18] and [JZC+18] for the proof.

Lemma 2.2 (QRO is PRF). *Let ℓ be a positive integer. Let $\mathcal{X}$ and $\mathcal{Y}$ be finite sets. Let $\mathsf{H}_{\mathsf{prf}}\colon \{0,1\}^\ell \times \mathcal{X} \to \mathcal{Y}$ and $\mathsf{H}_q\colon \mathcal{X} \to \mathcal{Y}$ be two independent random oracles. If an unbounded-time quantum adversary $\mathcal{A}$ makes queries to the random oracles at most Q times, then we have*

$$\left|\Pr[s \leftarrow \mathcal{M}, b \leftarrow \mathcal{A}^{\mathsf{H}_{\mathsf{prf}}(\cdot,\cdot),\mathsf{H}_{\mathsf{prf}}(s,\cdot)}() : b = 1] - \Pr[b \leftarrow \mathcal{A}^{\mathsf{H}_{\mathsf{prf}}(\cdot,\cdot),\mathsf{H}_q(\cdot)}() : b = 1]\right| \le 2Q \cdot 2^{-\ell/2},$$

where all oracle accesses of $\mathcal{A}$ can be quantum.

The second one states that QRO is collision-resistant.

Lemma 2.3 (QRO is collision-resistant [Zha15, Theorem 3.1]). *There is a universal constant C such that the following holds: Let $\mathcal{X}$ and $\mathcal{Y}$ be finite sets. Let $\mathsf{H}\colon \mathcal{X} \to \mathcal{Y}$ be a random oracle. If an unbounded-time quantum adversary $\mathcal{A}$ makes queries to H at most Q times, then we have*

$$\Pr_{\mathsf{H},\mathcal{A}}[(x,x') \leftarrow \mathcal{A}^{\mathsf{H}(\cdot)}() : x \ne x' \wedge \mathsf{H}(x) = \mathsf{H}(x')] \le C(Q+1)^3/|\mathcal{Y}|,$$

where all oracle accesses of $\mathcal{A}$ can be quantum.

Remark 2.1. We implicitly assume that $|\mathcal{X}| = \Omega(|\mathcal{Y}|)$, because of the birthday bound.

The thrid one states that two QROs are claw-free.

Lemma 2.4 (QROs are claw-free). *There is a universal constant C such that the following holds: Let $\mathcal{X}_0, \mathcal{X}_1$, and $\mathcal{Y}$ be finite sets. Let $N_0 = |\mathcal{X}_0|$ and $N_1 = |\mathcal{X}_1|$. Without loss of generality, we assume $N_0 \leq N_1$. Let $\mathsf{H}_0 \colon \mathcal{X}_0 \to \mathcal{Y}$ and $\mathsf{H}_1 \colon \mathcal{X}_1 \to \mathcal{Y}$ be two random oracles. If an unbounded-time quantum adversary $\mathcal{A}$ makes queries to H_0 and H_1 at most Q_0 and Q_1 times, respectively, then we have*

$$\Pr[(x_0, x_1) \leftarrow \mathcal{A}^{\mathsf{H}_0(\cdot),\mathsf{H}_1(\cdot)}() : \mathsf{H}_0(x_0) = \mathsf{H}_1(x_1)] \leq C(Q_0 + Q_1 + 1)^3/|\mathcal{Y}|,$$

where all oracle accesses of $\mathcal{A}$ can be quantum.

The following proof is due to Hosoyamada [Hos21]:

Proof. Let us reduce the problem to the collision-finding problem as follows: We assume that $\mathcal{X}_0$ and $\mathcal{X}_1$ are efficiently enumerable. Given $\mathsf{H} \colon [N_0 + N_1] \to \mathcal{Y}$, we define $\mathsf{H}_0 \colon \mathcal{X}_0 \to \mathcal{Y}$ and $\mathsf{H}_1 \colon \mathcal{X}_1 \to \mathcal{Y}$ by $\mathsf{H}_0(x) = \mathsf{H}(\mathsf{index}_0(x))$ and $\mathsf{H}_1(x) = \mathsf{H}(\mathsf{index}_1(x) + N_0)$, where $\mathsf{index}_i \colon \mathcal{X}_i \to [N_i]$ is an index function which returns the index of x in $\mathcal{X}_i$. H_0 and H_1 are random since H is a randomly chosen. If $\mathcal{A}$ finds the claw (x_0, x_1) for H_0 and H_1 with Q_0 and Q_1 queries, then we can find a collision $(\mathsf{index}_0(x_0), \mathsf{index}_1(x_1) + N_0)$ for H with $Q_0 + Q_1$ queries. Using Lemma 2.3, we obtain the bound as we wanted. □

2.1 Public-Key Encryption (PKE)

The model for PKE schemes is summarized as follows:

Definition 2.1. *A PKE scheme* PKE *consists of the following triple of PPT algorithms* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$*:*

- $\mathsf{Gen}(1^\kappa; r_g) \to (ek, dk)$*: a key-generation algorithm that on input 1^κ, where κ is the security parameter, and randomness $r_g \in \mathcal{R}_{\mathsf{Gen}}$, outputs a pair of keys (ek, dk). ek and dk are called the encryption key and decryption key, respectively.*
- $\mathsf{Enc}(ek, \mu; r_e) \to c$*: an encryption algorithm that takes as input encryption key ek, message $\mu \in \mathcal{M}$, and randomness $r_e \in \mathcal{R}_{\mathsf{Enc}}$, and outputs ciphertext $c \in \mathcal{C}$.*
- $\mathsf{Dec}(dk, c) \to \mu/\perp$*: a decryption algorithm that takes as input decryption key dk and ciphertext c and outputs message $\mu \in \mathcal{M}$ or a rejection symbol $\perp \notin \mathcal{M}$.*

We review δ-correctness in Hofheinz, Hövelmanns, and Kiltz [HHK17].

Definition 2.2 (δ-correctness [HHK17]). *Let $\delta = \delta(\kappa)$. We say* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* δ-correct *if*

$$\mathrm{Exp}_{(ek,dk)\leftarrow\mathsf{Gen}(1^\kappa)}\left[\max_{\mu\in\mathcal{M}} \Pr[c \leftarrow \mathsf{Enc}(ek, \mu) : \mathsf{Dec}(dk, c) \neq \mu]\right] \leq \delta.$$

In particular, we say that PKE is perfectly correct *if $\delta = 0$.*

We also define a key pair's accuracy.

Definition 2.3 (Accuracy [XY19]). *We say that a key pair (ek, dk) is* accurate *if for any $\mu \in \mathcal{M}$,*

$$\Pr_{c\leftarrow\mathsf{Enc}(ek,\mu)}[\mathsf{Dec}(dk, c) = \mu] = 1.$$

If a key pair is not accurate, then we call it inaccurate*. We note that if* PKE *is deterministic and δ-correct, then*

$$\Pr_{(ek,dk)\leftarrow\mathsf{Gen}(1^\kappa)}[(ek, dk) \text{ is inaccurate}] \leq \delta.$$

Security notions: We review onewayness under chosen-plaintext attacks (OW-CPA), onewayness under chosen-ciphertext attacks (OW-CCA), indistinguishability under chosen-plaintext attacks (IND-CPA), indistinguishability under chosen-ciphertext attacks (IND-CCA) [RS92, BDPR98]. We define pseudorandomness under chosen-ciphertext attacks (PR-CCA) and its strong version (SPR-CCA) with simulator $\mathcal{S}$ as a generalization of IND$-CCA-security in [vH04, Hop05]. We also review anonymity (ANON-CCA) [BBDP01], collision-freeness (WCFR-CCA and SCFR-CCA) [Moh10], and robustness (WROB-CCA and SROB-CCA) [Moh10]. We additionally define extended collision-freeness (XCFR), in which any efficient adversary cannot find a colliding ciphertext even if the adversary is given two decryption keys.

Definition 2.4 (Security notions for PKE). *Let* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a PKE scheme. Let $\mathcal{D}_{\mathcal{M}}$ be a distribution over the message space $\mathcal{M}$.*
For any $\mathcal{A}$ and goal-atk $\in \{\text{ind-cca}, \text{pr-cca}, \text{anon-cca}\}$*, we define its* goal-atk *advantage against* PKE *as follows:*

$$\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{PKE}[,\mathcal{S}],\mathcal{A}}(\kappa) := \left|2\Pr[\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{PKE}[,\mathcal{S}],\mathcal{A}}(\kappa) = 1] - 1\right|,$$

where $\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{PKE}[,\mathcal{S}],\mathcal{A}}(\kappa)$ *is an experiment described in Figure 1.*
For any $\mathcal{A}$ *and* goal-atk $\in$ {ow-cca, wcfr-cca, scfr-cca, xcfr, wrob-cca, srob-cca}*, we define its* goal-atk *advantage against* PKE *as follows:*

$$\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{PKE}[,\mathcal{D}_{\mathcal{M}}],\mathcal{A}}(\kappa) \coloneqq \Pr[\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{PKE}[,\mathcal{D}_{\mathcal{M}}],\mathcal{A}}(\kappa) = 1],$$

where $\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{PKE}[,\mathcal{D}_{\mathcal{M}}],\mathcal{A}}(\kappa)$ *is an experiment described in Figure 1.*
For GOAL-ATK $\in$ {IND-CCA, PR-CCA, ANON-CCA, OW-CCA, WCFR-CCA, SCFR-CCA, XCFR, WROB-CCA, SROB-CCA}*, we say that* PKE *is* GOAL-ATK*-secure if* $\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{PKE}[,\mathcal{D}_{\mathcal{M}},\mathcal{S}],\mathcal{A}}(\kappa)$ *is negligible for any QPT adversary* $\mathcal{A}$*. We also say that* PKE *is* SPR-CCA*-secure if it is* PR-CCA*-secure, and its simulator ignores ek. We also say that* PKE *is* GOAL-CPA*-secure if it is* GOAL-CCA*-secure even without the decryption oracle.*

Disjoint simulatability: We review disjoint simulatability defined in [SXY18].

Definition 2.5 (Disjoint simulatability [SXY18]). *Let* $\mathcal{D}_{\mathcal{M}}$ *denote an efficiently sampleable distribution on a set* $\mathcal{M}$*. A deterministic PKE scheme* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *with plaintext and ciphertext spaces* $\mathcal{M}$ *and* $\mathcal{C}$ *is* $\mathcal{D}_{\mathcal{M}}$-disjoint-simulatable *if there exists a PPT algorithm* $\mathcal{S}$ *that satisfies the followings:*

- *(Statistical disjointness:)*

$$\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) \coloneqq \max_{(ek,dk)\in\mathsf{Gen}(1^\kappa;\mathcal{R}_{\mathsf{Gen}})} \Pr[c \leftarrow \mathcal{S}(1^\kappa, ek) : c \in \mathsf{Enc}(ek, \mathcal{M})]$$

  is negligible.
- *(Ciphertext-indistinguishability:) For any QPT adversary* $\mathcal{A}$*, its* ds-ind *advantage* $\mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}}(\kappa)$ *is negligible: The advantage is defined as*

$$\mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}}(\kappa) \coloneqq \left|2\Pr[\mathsf{Expt}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}}(\kappa) = 1] - 1\right|,$$

  where $\mathsf{Expt}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}}(\kappa)$ *is an experiment described in Figure 1 and* $\mathcal{S}$ *is a PPT simulator.*

Liu and Wang gave a slightly modified version of statistical disjointness in [LW21]. As they noted, their definition below is enough to show the security proof:

$$\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) \coloneqq \Pr[(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa), c \leftarrow \mathcal{S}(1^\kappa, ek) : c \in \mathsf{Enc}(ek, \mathcal{M})]$$

Definition 2.6 (strong disjoint-simulatability). *We call* PKE *has* strong disjoint-simulatability *if* $\mathcal{S}$ *ignores ek.*

Remark 2.2. We note that a deterministic PKE scheme produced by TPunc [SXY18] or Punc [HKSU20] is *not strongly disjoint-simulatable*, because their simulator outputs a random ciphertext $\mathsf{Enc}(ek, \hat{\mu})$ of a special plaintext $\hat{\mu}$.

2.2 Key Encapsulation Mechanism (KEM)

The model for KEM schemes is summarized as follows:

Definition 2.7. *A KEM scheme* KEM *consists of the following triple of polynomial-time algorithms* $(\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$*:*

- $\overline{\mathsf{Gen}}(1^\kappa) \to (ek, dk)$*: a key-generation algorithm that on input* 1^κ*, where* κ *is the security parameter, outputs a pair of keys* (ek, dk)*. ek and dk are called the encapsulation key and decapsulation key, respectively.*
- $\overline{\mathsf{Enc}}(ek) \to (c, K)$*: an encapsulation algorithm that takes as input encapsulation key ek and outputs ciphertext* $c \in \mathcal{C}$ *and key* $K \in \mathcal{K}$*.*
- $\overline{\mathsf{Dec}}(dk, c) \to K/\perp$*: a decapsulation algorithm that takes as input decapsulation key dk and ciphertext c and outputs key K or a rejection symbol* $\perp \notin \mathcal{K}$*.*

Definition 2.8 (δ-correctness). *Let* $\delta = \delta(\kappa)$*. We say that* $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$ *is* δ-correct *if*

$$\Pr[(ek, dk) \leftarrow \overline{\mathsf{Gen}}(1^\kappa), (c, K) \leftarrow \overline{\mathsf{Enc}}(ek) : \overline{\mathsf{Dec}}(dk, c) \neq K] \leq \delta(\kappa).$$

In particular, we say that KEM *is* perfectly correct *if* $\delta = 0$*.*

$\underline{\mathsf{Expt}^{\text{ow-cca}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{A}}(\kappa)}$

$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
$\mu^* \leftarrow \mathcal{D}_{\mathcal{M}}$
$c^* \leftarrow \mathsf{Enc}(ek, \mu^*)$
$\mu' \leftarrow \mathcal{A}^{\textsc{Dec}_{c^*}(\cdot)}(ek, c^*)$
return $\mathsf{boole}(\mu' = \mathsf{Dec}(dk, c^*))$

$\underline{\textsc{Dec}_a(c)}$

if $c = a$ **then return** $\perp$
$\mu := \mathsf{Dec}(dk, c)$
return μ

$\underline{\textsc{Dec}'_a(\mathsf{id}, c)}$

if $c = a$ **then return** $\perp$
$\mu := \mathsf{Dec}(dk_{id}, c)$
return μ

$\underline{\mathsf{Expt}^{\text{ind-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)}$

$b \leftarrow \{0,1\}$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
$(\mu_0, \mu_1, state) \leftarrow \mathcal{A}_1^{\textsc{Dec}_\perp(\cdot)}(ek)$
$c^* \leftarrow \mathsf{Enc}(ek, \mu_b)$
$b' \leftarrow \mathcal{A}_2^{\textsc{Dec}_{c^*}(\cdot)}(c^*, state)$
return $\mathsf{boole}(b = b')$

$\underline{\mathsf{Expt}^{\text{pr-cca}}_{\mathsf{PKE},\mathcal{S},\mathcal{A}}(\kappa)}$

$b \leftarrow \{0,1\}$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
$(\mu, state) \leftarrow \mathcal{A}_1^{\textsc{Dec}_\perp(\cdot)}(ek)$
$c_0^* \leftarrow \mathsf{Enc}(ek, \mu)$
$c_1^* \leftarrow \mathcal{S}(1^\kappa, ek)$
$b' \leftarrow \mathcal{A}_2^{\textsc{Dec}_{c_b^*}(\cdot)}(c_b^*, state)$
return $\mathsf{boole}(b = b')$

$\underline{\mathsf{Expt}^{\text{anon-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)}$

$b \leftarrow \{0,1\}$
$(ek_0, dk_0) \leftarrow \mathsf{Gen}(1^\kappa)$
$(ek_1, dk_1) \leftarrow \mathsf{Gen}(1^\kappa)$
$(\mu, state) \leftarrow \mathcal{A}_1^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$
$c^* \leftarrow \mathsf{Enc}(ek_b, \mu)$
$b' \leftarrow \mathcal{A}_2^{\textsc{Dec}'_{c^*}(\cdot,\cdot)}(c^*, state)$
return $\mathsf{boole}(b = b')$

$\underline{\mathsf{Expt}^{\text{wcfr-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \mathsf{Gen}(1^\kappa)$
$(ek_1, dk_1) \leftarrow \mathsf{Gen}(1^\kappa)$
$(\mu, b) \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$
$c \leftarrow \mathsf{Enc}(ek_b, \mu)$
$\mu' \leftarrow \mathsf{Dec}(dk_{1-b}, c)$
return $\mathsf{boole}(\mu = \mu' \neq \perp)$

$\underline{\mathsf{Expt}^{\text{scfr-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \mathsf{Gen}(1^\kappa)$
$(ek_1, dk_1) \leftarrow \mathsf{Gen}(1^\kappa)$
$c \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$
$\mu_0 \leftarrow \mathsf{Dec}(dk_0, c)$
$\mu_1 \leftarrow \mathsf{Dec}(dk_1, c)$
return $\mathsf{boole}(\mu_0 = \mu_1 \neq \perp)$

$\underline{\mathsf{Expt}^{\text{xcfr}}_{\mathsf{PKE},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \mathsf{Gen}(1^\kappa)$
$(ek_1, dk_1) \leftarrow \mathsf{Gen}(1^\kappa)$
$c \leftarrow \mathcal{A}(ek_0, dk_0, ek_1, dk_1)$
$\mu_0 \leftarrow \mathsf{Dec}(dk_0, c)$
$\mu_1 \leftarrow \mathsf{Dec}(dk_1, c)$
return $\mathsf{boole}(\mu_0 = \mu_1 \neq \perp)$

$\underline{\mathsf{Expt}^{\text{wrob-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \mathsf{Gen}(1^\kappa)$
$(ek_1, dk_1) \leftarrow \mathsf{Gen}(1^\kappa)$
$(\mu, b) \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$
$c \leftarrow \mathsf{Enc}(ek_b, \mu)$
$\mu' \leftarrow \mathsf{Dec}(dk_{1-b}, c)$
return $\mathsf{boole}(\mu' \neq \perp)$

$\underline{\mathsf{Expt}^{\text{srob-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \mathsf{Gen}(1^\kappa)$
$(ek_1, dk_1) \leftarrow \mathsf{Gen}(1^\kappa)$
$c \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$
$\mu_0 \leftarrow \mathsf{Dec}(dk_0, c)$
$\mu_1 \leftarrow \mathsf{Dec}(dk_1, c)$
return $\mathsf{boole}(\mu_0 \neq \perp \wedge \mu_1 \neq \perp)$

$\underline{\mathsf{Expt}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}}(\kappa)}$

$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
$\mu* \leftarrow \mathcal{D}_{\mathcal{M}}$
$c_0^* := \mathsf{Enc}(ek, \mu^*)$
$c_1^* \leftarrow \mathcal{S}(1^\kappa, ek)$
$b' \leftarrow \mathcal{A}(ek, c_b^*)$
return $\mathsf{boole}(b = b')$

Fig. 1. Games for PKE schemes

Security notions: We review indistinguishability under chosen-plaintext attacks (IND-CPA) and indistinguishability under chosen-ciphertext attacks (IND-CCA) [RS92, BDPR98]. We define pseudorandomness under chosen-ciphertext attacks (PR-CCA) with simulator $\mathcal{S}$ as a generalization of IND$-CCA-security in [vH04, Hop05] and its strong version (SPR-CCA). We also review anonymity (ANON-CCA), collision-freeness (WCFR-CCA and SCFR-CCA), and robustness (WROB-CCA and SROB-CCA) [GMP21a]. We also define *smoothness* under chosen-ciphertext attacks (denoted by SMT-CCA) by following smoothness of hash proof system [CS02]:

Definition 2.9 (Security notions for KEM). *Let* $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$ *be a KEM scheme.*
For any $\mathcal{A}$ *and* goal-atk $\in$ {ind-cca, pr-cca, smt-cca, anon-cca}*, we define its* goal-atk *advantage against* KEM *as follows:*

$$\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{KEM}[,\mathcal{S}],\mathcal{A}}(\kappa) := \left|2\Pr[\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{KEM}[,\mathcal{S}],\mathcal{A}}(\kappa) = 1] - 1\right|,$$

where $\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{KEM}[,\mathcal{S}],\mathcal{A}}(\kappa)$ *is an experiment described in Figure 1.*
For any $\mathcal{A}$ *and* goal-atk $\in$ {wcfr-cca, scfr-cca, wrob-cca, srob-cca}*, we define its* goal-atk *advantage against* KEM *as follows:*

$$\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{KEM},\mathcal{A}}(\kappa) := \Pr[\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1],$$

where $\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ *is an experiment described in Figure 1.*
For GOAL-ATK $\in$ {IND-CCA, PR-CCA, SMT-CCA, ANON-CCA, WCFR-CCA, SCFR-CCA, WROB-CCA, SROB-CCA}*, we say that* KEM *is* GOAL-ATK*-secure if* $\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{KEM}[,\mathcal{S}],\mathcal{A}}(\kappa)$ *is negligible for any QPT adversary* $\mathcal{A}$*. We say that* KEM *is* SPR-CCA*-secure (or* SSMT-CCA*-secure) if it is* PR-CCA*-secure (or* SMT-CCA*-secure) and its simulator ignores ek, respectively. We say that* KEM *is* wANON-CCA*-secure if it is* ANON-CCA*-secure where we modify the input* (ek_0, ek_1, c^*, K^*) *into* (ek_0, ek_1, c^*)*. We also say that* KEM *is* GOAL-CPA*-secure if it is* GOAL-CCA*-secure even without the decapsulation oracle.*

We additionally define ϵ-sparseness.

Definition 2.10 (ϵ-sparseness). *Let* $\mathcal{S}$ *be a simulator for the* PR-CCA *security. We say that* KEM *is* ϵ-sparse *if*

$$\Pr[(ek, dk) \leftarrow \overline{\mathsf{Gen}}(1^\kappa), c \leftarrow \mathcal{S}(1^\kappa, ek) : \overline{\mathsf{Dec}}(dk, c) \neq \perp] \leq \epsilon.$$

2.3 Data Encapsulation Mechanism (DEM)

The model for DEM schemes is summarized as follows:

Definition 2.11. *A DEM scheme* DEM *consists of the following pair of polynomial-time algorithms* (E, D) *with key space* $\mathcal{K}$ *and message space* $\mathcal{M}$*:*

- $\mathsf{E}(K, \mu) \to d$*: an encapsulation algorithm that takes as input key* K *and data* μ *and outputs ciphertext* d*.*
- $\mathsf{D}(K, d) \to m/\perp$*: a decapsulation algorithm that takes as input key* K *and ciphertext* d *and outputs data* μ *or a rejection symbol* $\perp \notin \mathcal{M}$*.*

Definition 2.12 (Correctness). *We say* $\mathsf{DEM} = (\mathsf{E}, \mathsf{D})$ *has* perfect correctness *if for any* $K \in \mathcal{K}$ *and any* $\mu \in \mathcal{M}$*, we have*

$$\Pr[d \leftarrow \mathsf{E}(K, \mu) : \mathsf{D}(K, d) = \mu] = 1.$$

Security notions: We review indistinguishability under chosen-ciphertext attacks (IND-CCA), pseudorandomness under chosen-ciphertext attacks (PR-CCA) and pseudorandomness under one-time chosen-ciphertext attacks (PR-OTCCA). We also review the integrity of ciphertext (INT-CTXT). Robustness of DEM (FROB and XROB) are taken from Farshim, Orlandi, and Roşi [FOR17].

Definition 2.13 (Security notions for DEM). *Let* $\mathsf{DEM} = (\mathsf{E}, \mathsf{D})$ *be a DEM scheme whose key space is* $\mathcal{K}$*. For* $\mu \in \mathcal{M}$*, let* $C_{|\mu|}$ *be a ciphertext space defined by the length of message* μ*.*
For any $\mathcal{A}$ *and* goal-atk $\in$ {ind-cca, pr-cca, pr-otcca}*, we define its* goal-atk *advantage against* DEM *as follows:*

$$\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{DEM},\mathcal{A}}(\kappa) := \left|2\Pr[\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{DEM},\mathcal{A}}(\kappa) = 1] - 1\right|,$$

where $\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{DEM},\mathcal{A}}(\kappa)$ *is an experiment described in Figure 1.*
For any $\mathcal{A}$ *and* goal-atk $\in$ {int-ctxt, frob, xrob}*, we define its* goal-atk *advantage against* DEM *as follows:*

$$\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{DEM},\mathcal{A}}(\kappa) := \Pr[\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{DEM},\mathcal{A}}(\kappa) = 1],$$

where $\mathsf{Expt}^{\text{goal-atk}}_{\mathsf{DEM},\mathcal{A}}(\kappa)$ *is an experiment described in Figure 1.*
For GOAL-ATK $\in$ {IND-CCA, PR-CCA, PR-OTCCA, INT-CTXT, FROB, XROB}*, we say that* DEM *is* GOAL-ATK*-secure if* $\mathsf{Adv}^{\text{goal-atk}}_{\mathsf{DEM},\mathcal{A}}(\kappa)$ *is negligible for any QPT adversary* $\mathcal{A}$*.*

$\underline{\mathsf{Expt}^{\text{ind-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)}$

$b \leftarrow \{0,1\}$

$(ek, dk) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(c^*, K_0^*) \leftarrow \overline{\mathsf{Enc}}(ek)$;

$K_1^* \leftarrow \mathcal{K}$

$b' \leftarrow \mathcal{A}^{\textsc{Dec}_{c^*}(\cdot)}(ek, c^*, K_b^*)$

return $\mathsf{boole}(b = b')$

$\underline{\textsc{Dec}_a(c)}$

if $c = a$ **then return** $\perp$

$K := \overline{\mathsf{Dec}}(dk, c)$

return K

$\underline{\textsc{Dec}'_a(\mathsf{id}, c)}$

if $c = a$ **then return** $\perp$

$K := \overline{\mathsf{Dec}}(dk_{\mathsf{id}}, c)$

return K

$\underline{\mathsf{Expt}^{\text{pr-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa)}$

$b \leftarrow \{0,1\}$

$(ek, dk) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(c_0^*, K_0^*) \leftarrow \overline{\mathsf{Enc}}(ek)$;

$(c_1^*, K_1^*) \leftarrow \mathcal{S}(1^\kappa, ek) \times \mathcal{K}$

$b' \leftarrow \mathcal{A}^{\textsc{Dec}_{c_b^*}(\cdot)}(ek, c_b^*, K_b^*)$

return $\mathsf{boole}(b = b')$

$\underline{\mathsf{Expt}^{\text{smt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa)}$

$b \leftarrow \{0,1\}$

$(ek, dk) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(c^*, K_0^*) \leftarrow \mathcal{S}(1^\kappa, ek) \times \mathcal{K}$

$K_1^* \leftarrow \overline{\mathsf{Dec}}(dk, c^*)$

$b' \leftarrow \mathcal{A}^{\textsc{Dec}_{c^*}(\cdot)}(ek, c^*, K_b^*)$

return $\mathsf{boole}(b = b')$

$\underline{\mathsf{Expt}^{\text{anon-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)}$

$b \leftarrow \{0,1\}$

$(ek_0, dk_0) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(ek_1, dk_1) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(c^*, K^*) \leftarrow \overline{\mathsf{Enc}}(ek_b)$;

$b' \leftarrow \mathcal{A}^{\textsc{Dec}'_{c^*}(\cdot,\cdot)}(ek_0, ek_1, c^*, K^*)$

return $\mathsf{boole}(b = b')$

$\underline{\mathsf{Expt}^{\text{wcfr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(ek_1, dk_1) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$b \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$

$(c, K_b) \leftarrow \overline{\mathsf{Dec}}(ek_b)$

$K_{1-b} \leftarrow \overline{\mathsf{Dec}}(dk_{1-b}, c)$

return $\mathsf{boole}(K_0 = K_1 \neq \perp)$

$\underline{\mathsf{Expt}^{\text{scfr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(ek_1, dk_1) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$c \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$

$K_0 \leftarrow \overline{\mathsf{Dec}}(dk_0, c)$

$K_1 \leftarrow \overline{\mathsf{Dec}}(dk_1, c)$

return $\mathsf{boole}(K_0 = K_1 \neq \perp)$

$\underline{\mathsf{Expt}^{\text{wrob-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(ek_1, dk_1) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$b \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$

$(c, K_b) \leftarrow \overline{\mathsf{Dec}}(ek_b)$

$K_{1-b} \leftarrow \overline{\mathsf{Dec}}(dk_{1-b}, c)$

return $\mathsf{boole}(K_{1-b} \neq \perp)$

$\underline{\mathsf{Expt}^{\text{srob-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)}$

$(ek_0, dk_0) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$(ek_1, dk_1) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$

$c \leftarrow \mathcal{A}^{\textsc{Dec}'_\perp(\cdot,\cdot)}(ek_0, ek_1)$

$K_0 \leftarrow \overline{\mathsf{Dec}}(dk_0, c)$

$K_1 \leftarrow \overline{\mathsf{Dec}}(dk_1, c)$

return $\mathsf{boole}(K_0 \neq \perp \wedge K_1 \neq \perp)$

Fig. 2. Games for KEM schemes

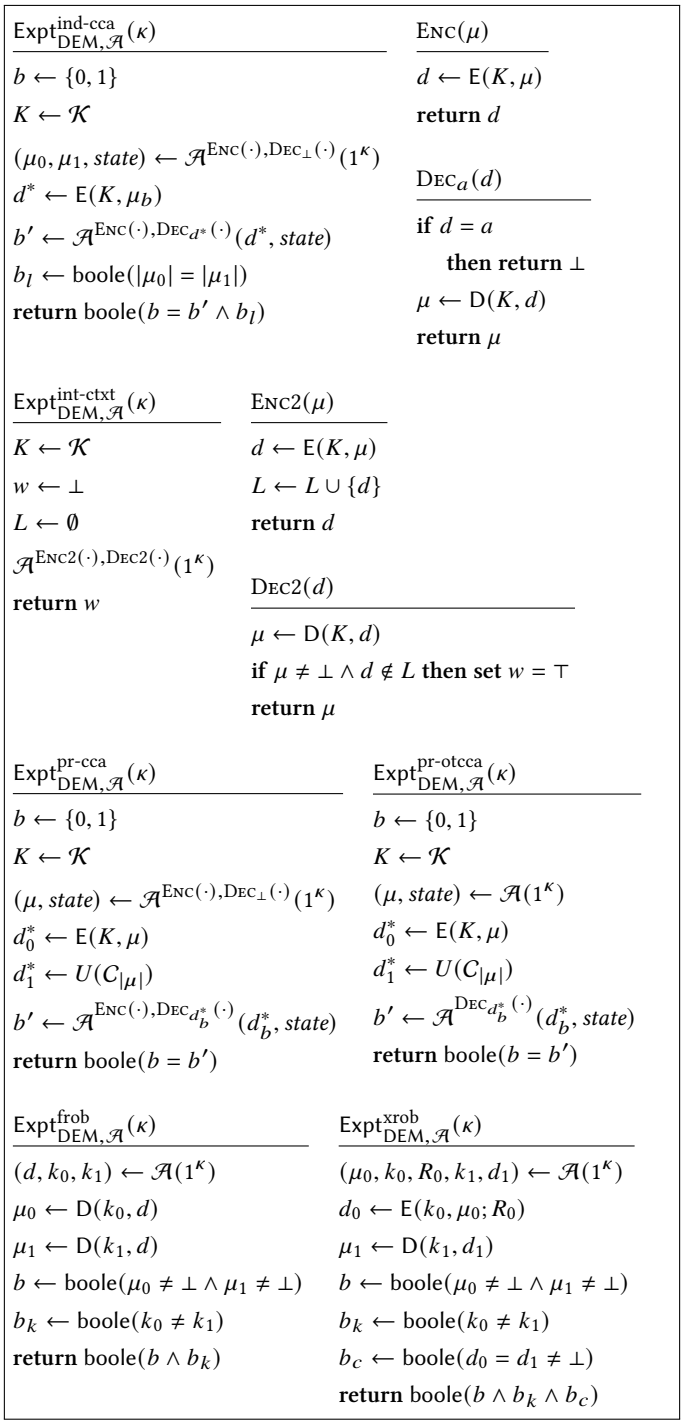

Fig. 3. Games for DEM schemes

2.4 Review of Grubbs, Maram, and Paterson [GMP21a]

Grubbs et al. studied KEM's anonymity and hybrid PKE's anonymity and robustness by extending the results of Mohassel [Moh10]. We use $\mathsf{KEM}^{\perp}$ and $\mathsf{KEM}^{\not\perp}$ to indicate KEM with explicit rejection and implicit rejection, respectively. For KEM with explicit rejection, they showed the following theorem which generalizes Mohassel's theorem [Moh10]:

Theorem 2.1 ([GMP21a, Theorem 1]). *Let* $\mathsf{PKE}_{\mathsf{hy}} = \mathsf{Hyb}[\mathsf{KEM}^{\perp}, \mathsf{DEM}]$*, a hybrid PKE scheme obtained by composing* $\mathsf{KEM}^{\perp}$ *and* DEM*. (See Figure 4.)*

1. *If* $\mathsf{KEM}^{\perp}$ *is* wANON-CPA*-secure,* IND-CCA*-secure,* WROB-CCA*-secure, and* δ*-correct and* DEM *is* INT-CTXT*-secure, then* $\mathsf{PKE}_{\mathsf{hy}}$ *is* ANON-CCA*-secure.*
2. *If* $\mathsf{KEM}^{\perp}$ *is* SROB-CCA*-secure (and* WROB-CCA*-secure), then* $\mathsf{PKE}_{\mathsf{hy}}$ *is* SROB-CCA*-secure (and* WROB-CCA*-secure), respectively.*

Grubbs et al. [GMP21a] then treat KEM with implicit rejection, which is used in all NIST PQC Round 3 KEM candidates except HQC. Their results are related to the FO transform with implicit rejection, which is decomposed into two transforms, T and $\mathsf{U}^{\not\perp}$: T transforms a probabilistic PKE scheme PKE into a deterministic PKE scheme PKE_1 with a random oracle G; $\mathsf{U}^{\not\perp}$ transforms a deterministic PKE scheme PKE_1 into a probabilistic KEM KEM with a random oracle H. Roughly speaking, they showed the following two theorems on robustness and anonymity of hybrid PKE from KEM with implicit rejection:

Theorem 2.2 (Robustness of $\mathsf{PKE}_{\mathsf{hy}}$ [GMP21a, Theorem 2]). *Let* $\mathsf{PKE}_{\mathsf{hy}} = \mathsf{Hyb}[\mathsf{KEM}^{\not\perp}, \mathsf{DEM}]$*. If* $\mathsf{KEM}^{\not\perp}$ *is* SCFR-CCA*-secure (and* WCFR-CCA*-secure) and* DEM *is* FROB*-secure (and* XROB*-secure), then* $\mathsf{PKE}_{\mathsf{hy}}$ *is* SROB-CCA*-secure (and* WROB-CCA*-secure), respectively.*

Theorem 2.3 (Anonymity of $\mathsf{PKE}_{\mathsf{hy}}$ using $\mathsf{FO}^{\not\perp}$ [GMP21a, Theorem 7]). *Let* $\mathsf{PKE}_{\mathsf{hy}} = \mathsf{Hyb}[\mathsf{KEM}^{\not\perp}, \mathsf{DEM}]$*. If* PKE *is* δ*-correct, and* γ*-spreading,* $\mathsf{PKE}_1 = \mathsf{T}[\mathsf{PKE}, \mathsf{G}]$ *is* WCFR-CPA*-secure,* $\mathsf{KEM}^{\not\perp} = \mathsf{FO}^{\not\perp}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$ *is* ANON-CCA*-secure and* IND-CCA*-secure,* DEM *is* INT-CTXT*-secure, then* $\mathsf{PKE}_{\mathsf{hy}}$ *is* ANON-CCA*-secure.*

They also showed that the following theorem:

Theorem 2.4 (Anonymity of $\mathsf{KEM}^{\not\perp}$ using $\mathsf{FO}^{\not\perp}$ [GMP21a, Theorem 5]). *If* PKE *is* wANON-CPA*-secure,* OW-CPA*-secure, and* δ*-correct, and* $\mathsf{PKE}_1 = \mathsf{T}[\mathsf{PKE}, \mathsf{G}]$ *is* SCFR-CPA*-secure, then a KEM scheme* $\mathsf{KEM} = \mathsf{FO}^{\not\perp}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$ *is* ANON-CCA*-secure.*

Grubbs et al. reduced from the wANON-CPA-security of PKE to the ANON-CCA-security of KEM. We note that there are two decapsulation oracles in the security game of the ANON-CCA-security of KEM. Thus, they need to simulate *both* decapsulation oracles without secrets. Jiang et al. [JZC+18] used the simulation trick that replaces $\mathsf{H}(\mu, c)$ with $\mathsf{H}_q(\mathsf{Enc}(ek, \mu))$ if $c = \mathsf{Enc}(ek, \mu)$ and $\mathsf{H}'_q(\mu, c)$ else, which helps the simulation of the decapsulation oracle without secrets in the QROM. Grubbs et al. extended this trick to simulate *two* decapsulation oracles by replacing $\mathsf{H}(\mu, c)$ with $\mathsf{H}_{q,i}(\mathsf{Enc}(ek_i, \mu))$ if $c = \mathsf{Enc}(ek_i, \mu)$ and $\mathsf{H}'_q(\mu, c)$ else. Notice that this extended simulation heavily depends on the fact that H takes μ and c and the SCFR-CCA-security of PKE_1. If the random oracle takes μ only, their trick fails the simulation.

2.5 Strong Pseudorandomness Implies Anonymity

We observe that strong pseudorandomness of PKE/KEM immediately implies anonymity of PKE/KEM, which may be folklore. We give the proof for PKE for completeness.

Theorem 2.5. *If* PKE *or* KEM *is* SPR-CCA*-secure, then it is* ANON-CCA*-secure.*

Proof. Here we only consider the case for PKE, since the proof for the case for KEM is obtained by the similar way. Let us define four games $\mathsf{Game}_{i,b}$ for $i, b \in \{0, 1\}$:

- $\mathsf{Game}_{0,b}$ for $b \in \{0, 1\}$: This is the original game $\mathsf{Expt}^{\text{anon-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)$ with $b = 0$ and 1.
- $\mathsf{Game}_{1,b}$ for $b \in \{0, 1\}$: This game is the same as $\mathsf{Game}_{0,b}$ except that the target ciphertext is randomly taken from $\mathcal{S}(1^\kappa) \times U(C_{\mathsf{DEM},|\mu|})$.

Let $S_{i,b}$ be the event that the adversary outputs 1 in $\mathsf{Game}_{i,b}$.
It is easy to see that there exist two adversaries $\mathcal{A}_{10}$ and $\mathcal{A}_{11}$ whose running times are the same as that of $\mathcal{A}$ satisfying

$$\left|\Pr[S_{0,b}] - \Pr[S_{1,b}]\right| \leq \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{PKE},\mathcal{S},\mathcal{A}_{1b}}(\kappa).$$

In addition, we have

$$\Pr[S_{1,0}] = \Pr[S_{1,1}]$$

since the distribution of the target ciphertext in both game is $\mathcal{S}(1^\kappa) \times U(C_{\mathsf{DEM},|\mu|})$. Hence, we have

$$\begin{aligned}
\mathsf{Adv}^{\text{anon-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa) &= \left|\Pr[S_{0,0}] - \Pr[S_{0,1}]\right| \\
&\leq \left|\Pr[S_{0,0}] - \Pr[S_{1,0}]\right| + \left|\Pr[S_{1,0}] - \Pr[S_{1,1}]\right| + \left|\Pr[S_{1,1}] - \Pr[S_{0,1}]\right| \\
&\leq \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{PKE},\mathcal{S},\mathcal{A}_{10}}(\kappa) + \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{PKE},\mathcal{S},\mathcal{A}_{11}}(\kappa).
\end{aligned}$$

This completes the proof. □

3 Strong Pseudorandomness of Hybrid PKE

The hybrid PKE $\mathsf{PKE}_{\mathsf{hy}} = (\mathsf{Gen}_{\mathsf{hy}}, \mathsf{Enc}_{\mathsf{hy}}, \mathsf{Dec}_{\mathsf{hy}})$ constructed from $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$ and $\mathsf{DEM} = (\mathsf{E}, \mathsf{D})$ is summarized as in Figure 4

$\mathsf{Gen}_{\mathsf{hy}}(1^\kappa)$	$\mathsf{Enc}_{\mathsf{hy}}(ek, \mu)$	$\mathsf{Dec}_{\mathsf{hy}}(dk, ct = (c, d))$
$(ek, dk) \leftarrow \overline{\mathsf{Gen}}(1^\kappa)$	$(c, K) \leftarrow \overline{\mathsf{Enc}}(ek)$	$K' \leftarrow \overline{\mathsf{Dec}}(dk, c)$
return (ek, dk)	$d \leftarrow \mathsf{E}(K, \mu)$	**if** $K' = \perp$ **then return** $\perp$
	return $ct := (c, d)$	$\mu' \leftarrow \mathsf{D}(K', d)$
		if $\mu' = \perp$ **then return** $\perp$
		return μ'

Fig. 4. $\mathsf{PKE}_{\mathsf{hy}} = \mathsf{Hyb}[\mathsf{KEM}, \mathsf{DEM}]$

We show the following two theorems on strong pseudorandomness and anonymity of a hybrid PKE:

Theorem 3.1 (Case for KEM with explicit rejection). *Let* $\mathsf{PKE}_{\mathsf{hy}} = (\mathsf{Gen}_{\mathsf{hy}}, \mathsf{Enc}_{\mathsf{hy}}, \mathsf{Dec}_{\mathsf{hy}})$ *be a hybrid encryption scheme obtained by composing a KEM scheme* $\mathsf{KEM}^{\perp} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$ *and a DEM scheme* $\mathsf{DEM} = (\mathsf{E}, \mathsf{D})$ *that share key space* $\mathcal{K}$*. If* $\mathsf{KEM}^{\perp}$ *is* SPR-CCA*-secure,* δ*-correct with negligible* δ*, and* ϵ*-sparse and* DEM *is* PR-OTCCA*-secure and* INT-CTXT*-secure, then* $\mathsf{PKE}_{\mathsf{hy}}$ *is* SPR-CCA*-secure (and* ANON-CCA*-secure).*
Formally speaking, for any $\mathcal{A}$ *against the* SPR-CCA *security of* $\mathsf{PKE}_{\mathsf{hy}}$*, there exist* $\mathcal{A}_{23}$ *against the* SPR-CCA *security of* $\mathsf{KEM}^{\perp}$*,* $\mathcal{A}_{34}$ *against the* SPR-OTCCA *security of* DEM*, and* $\mathcal{A}_{45}$ *against the* INT-CTXT *security of* DEM *such that*

$$\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa) \le \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM}^{\perp}, \mathcal{S}, \mathcal{A}_{23}}(\kappa) + \mathsf{Adv}^{\text{spr-otcca}}_{\mathsf{DEM}, \mathcal{A}_{34}}(\kappa) + \mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{DEM}, \mathcal{A}_{45}}(\kappa) + \delta + \epsilon.$$

Theorem 3.2 (Case for KEM with implicit rejection). *Let* $\mathsf{PKE}_{\mathsf{hy}} = (\mathsf{Gen}_{\mathsf{hy}}, \mathsf{Enc}_{\mathsf{hy}}, \mathsf{Dec}_{\mathsf{hy}})$ *be a hybrid encryption scheme obtained by composing a KEM scheme* $\mathsf{KEM}^{\not\perp} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}})$ *and a DEM scheme* $\mathsf{DEM} = (\mathsf{E}, \mathsf{D})$ *that share key space* $\mathcal{K}$*. If* $\mathsf{KEM}^{\not\perp}$ *is* SPR-CCA*-secure,* SSMT-CCA*-secure, and* δ*-correct with negligible* δ *and* DEM *is* PR-OTCCA*-secure, then* $\mathsf{PKE}_{\mathsf{hy}}$ *is* SPR-CCA*-secure (and* ANON-CCA*-secure).*
Formally speaking, for any $\mathcal{A}$ *against the* SPR-CCA *security of* $\mathsf{PKE}_{\mathsf{hy}}$*, there exist* $\mathcal{A}_{23}$ *against the* SPR-CCA *security of* $\mathsf{KEM}^{\not\perp}$*,* $\mathcal{A}_{34}$ *against the* SPR-OTCCA *security of* DEM*, and* $\mathcal{A}_{45}$ *against the* SSMT-CCA *security of* $\mathsf{KEM}^{\not\perp}$ *such that*

$$\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa) \le \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM}^{\not\perp}, \mathcal{S}, \mathcal{A}_{23}}(\kappa) + \mathsf{Adv}^{\text{spr-otcca}}_{\mathsf{DEM}, \mathcal{A}_{34}}(\kappa) + \mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM}^{\not\perp}, \mathcal{S}, \mathcal{A}_{45}}(\kappa) + \delta.$$

We here prove Theorem 3.2 and give the proof of Theorem 3.1 in subsection B.1.

Proof of Theorem 3.2 Let us consider Game_i for $i = 0, \ldots, 6$. We summarize the games in Table 2. Let S_i denote the event that the adversary outputs $b' = 1$ in Game_i.
Let $\mathcal{S}$ be the simulator for the SPR-CCA security of $\mathsf{KEM}^{\not\perp}$. We define $\mathcal{S}_{\mathsf{hy}}(1^\kappa, |\mu^*|) := \mathcal{S}(1^\kappa) \times U(C_{|\mu^*|})$ as the simulator for the SPR-CCA security of $\mathsf{PKE}_{\mathsf{hy}}$.
The security proof is similar to the security proof of the IND-CCA security of KEM/DEM [CS03] for $\mathsf{Game}_0, \ldots, \mathsf{Game}_4$. We need to take care of pseudorandom ciphertexts when moving from Game_4 to Game_5 and require the SSMT-CCA security of $\mathsf{KEM}^{\not\perp}$.

Game_0: This is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa)$ with $b = 0$. Given μ^*, the challenge ciphertext is computed as follows:

$$(c^*, K^*) \leftarrow \overline{\mathsf{Enc}}(ek); d^* \leftarrow \mathsf{E}(K^*, \mu^*);\ \text{return } ct^* = (c^*, d^*).$$

We have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: In this game, c^* and K^* are generated before invoking $\mathcal{A}$ with ek. This change is just conceptual, and we have

$$\Pr[S_0] = \Pr[S_1].$$

Table 2. Summary of Games for the Proof of Theorem 3.2

Game	c^* and K^*	d^*	Decryption	Justification
Game_0	$\overline{\mathsf{Enc}}(ek)$	$\mathsf{E}(K^*, \mu^*)$		
Game_1	$\overline{\mathsf{Enc}}(ek)$ at first	$\mathsf{E}(K^*, \mu^*)$		conceptual change
Game_2	$\overline{\mathsf{Enc}}(ek)$ at first	$\mathsf{E}(K^*, \mu^*)$	use K^* if $c = c^*$	δ-correctness of $\mathsf{KEM}^{\not\perp}$
Game_3	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$ at first	$\mathsf{E}(K^*, \mu^*)$	use K^* if $c = c^*$	SPR-CCA security of $\mathsf{KEM}^{\not\perp}$
Game_4	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$ at first	$U(\mathcal{C}_{\lvert\mu^*\rvert})$	use K^* if $c = c^*$	SPR-OTCCA security of DEM
Game_5	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$ at first	$U(\mathcal{C}_{\lvert\mu^*\rvert})$		SSMT-CCA security of $\mathsf{KEM}^{\not\perp}$
Game_6	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$	$U(\mathcal{C}_{\lvert\mu^*\rvert})$		conceptual change

Game_2: In this game, the decryption oracle uses K^* if $c = c^*$ instead of $K = \overline{\mathsf{Dec}}(dk, c^*)$. Game_1 and Game_2 differ if correctly generated ciphertext c^* with K^* is decapsulated into different $K \neq K^*$ or $\perp$, which violates the correctness and occurs with probability at most δ. Hence, the difference of Game_1 and Game_2 is bounded by δ, and we have

$$|\Pr[S_1] - \Pr[S_2]| \leq \delta.$$

We note that this corresponds to the event BadKeyPair in [CS03].

Game_3: In this game, the challenger uses random (c^*, K^*) and uses K^* in DEM. The challenge ciphertext is generated as follows:

$$(c^*, K^*) \leftarrow \mathcal{S}(1^\kappa) \times U(\mathcal{K}); d^+ \leftarrow \mathsf{E}(K^*, \mu^*);\ \text{return } ct^* = (c^*, d^+).$$

The difference is bounded by the SPR-CCA security of $\mathsf{KEM}^{\not\perp}$: There is an adversary $\mathcal{A}_{23}$ whose running time is approximately the same as that of $\mathcal{A}$ satisfying

$$|\Pr[S_2] - \Pr[S_3]| \leq \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM}^{\not\perp}, \mathcal{S}, \mathcal{A}_{23}}(\kappa).$$

We omit the detail of $\mathcal{A}_{23}$ since it is straightforward.

Game_4: In this game, the challenger uses random d^*. The challenge ciphertext is generated as follows:

$$(c^*, K^*) \leftarrow \mathcal{S}(1^\kappa) \times U(\mathcal{K}); d^* \leftarrow U(\mathcal{C}_{\lvert\mu^*\rvert});\ \text{return } ct^* = (c^*, d^*).$$

The difference is bounded by the SPR-OTCCA security of DEM: There is an adversary $\mathcal{A}_{34}$ whose running time is approximately the same as that of $\mathcal{A}$ satisfying

$$|\Pr[S_3] - \Pr[S_4]| \leq \mathsf{Adv}^{\text{spr-otcca}}_{\mathsf{DEM}, \mathcal{A}_{34}}(\kappa).$$

We omit the detail of $\mathcal{A}_{34}$ since it is straightforward.

Game_5: We replace the decryption oracle defined as follows: If given $ct = (c^*, d)$, the decryption oracle uses $K = \overline{\mathsf{Dec}}(dk, c^*)$ instead of K^*.
The difference is bounded by the SSMT-CCA security of $\mathsf{KEM}^{\not\perp}$: There is an adversary $\mathcal{A}_{45}$ whose running time is approximately the same as that of $\mathcal{A}$ satisfying

$$|\Pr[S_4] - \Pr[S_5]| \leq \mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM}^{\not\perp}, \mathcal{S}, \mathcal{A}_{45}}(\kappa).$$

We omit the detail of $\mathcal{A}_{45}$ since it is straightforward.

Game_6: We finally change the timing of the generation of (c^*, K^*). This change is just conceptual, and we have

$$\Pr[S_5] = \Pr[S_6].$$

Notice that this is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa)$ with $b = 1$, thus, we have

$$\Pr[S_6] = \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing the (in)equalities, we obtain the bound in the statement as follows:

$$\begin{aligned}\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa) = |\Pr[S_0] - \Pr[S_6]| &\leq \sum_{i=0}^{5} |\Pr[S_i] - \Pr[S_{i+1}]| \\ &\leq \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM}^{\not\perp}, \mathcal{S}, \mathcal{A}_{23}}(\kappa) + \mathsf{Adv}^{\text{spr-otcca}}_{\mathsf{DEM}, \mathcal{A}_{34}}(\kappa) + \mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM}^{\not\perp}, \mathcal{S}, \mathcal{A}_{45}}(\kappa) + \delta.\end{aligned}$$

□

4 Properties of SXY

Let us review SXY [SXY18] as known as $\mathsf{U}^{\not\perp}_m$ with explicit re-encryption check [HHK17].
Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme. Let $\mathcal{M}$, $\mathcal{C}$, and $\mathcal{K}$ be a plaintext, ciphertext, and key space of PKE, respectively. Let $\mathsf{H}\colon \mathcal{M} \to \mathcal{K}$ and $\mathsf{H}_{\mathsf{prf}}\colon \{0,1\}^{\ell} \times \mathcal{C} \to \mathcal{K}$ be hash functions modeled by random oracles. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{SXY}[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$ is defined as in Figure 5.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, c)$, where $\overline{dk} = (dk, ek, s)$
1 : $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	1 : $\mu \leftarrow \mathcal{D}_{\mathcal{M}}$	1 : $\mu' \leftarrow \mathsf{Dec}(dk, c)$
2 : $s \leftarrow \{0,1\}^{\ell}$	2 : $c := \mathsf{Enc}(ek, \mu)$	2 : **if** $\mu' = \perp$ or $c \neq \mathsf{Enc}(ek, \mu')$
3 : $\overline{dk} := (dk, ek, s)$	3 : $K := \mathsf{H}(\mu)$	3 : **then return** $K := \mathsf{H}_{\mathsf{prf}}(s, c)$
4 : **return** $(ek, \overline{dk})$	4 : **return** (c, K)	4 : **else return** $K := \mathsf{H}(\mu')$

Fig. 5. $\mathsf{KEM} = \mathsf{SXY}[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$

4.1 SPR-CCA Security

We first show that KEM is strongly pseudorandom if the underlying PKE is strongly disjoint-simulatable.

Theorem 4.1 (Case of derandomized PKE). *Let* PKE_0 *be a probabilistic PKE. Let us consider a derandomized PKE* $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable and* δ*-correct with negligible* δ*, then* $\mathsf{KEM} = \mathsf{SXY}[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$ *is* SPR-CCA*-secure.*
Formally speaking, for any $\mathcal{A}$ *against the* SPR-CCA *security of* KEM *issuing at most* q_{DEC} *queries to the decapsulation oracle and* q_{G}, q_{H}, *and* $q_{\mathsf{H}_{\mathsf{prf}}}$ *queries to* G, H, *and* $\mathsf{H}_{\mathsf{prf}}$, *respectively, there exists* $\mathcal{A}_{34}$ *against ciphertext-indistinguishability of* PKE *such that*

$$\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) \le \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4\delta + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{DEC}}) \cdot 2^{-\ell/2} + 16(q_{\mathsf{G}} + q_{\mathrm{DEC}} + 2)^2\delta + 16(q_{\mathsf{G}} + q_{\mathsf{H}} + 2)^2\delta.$$

Theorem 4.2 (Case for non-derandomized PKE). *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable and* δ*-correct with negligible* δ*, then* $\mathsf{KEM} = \mathsf{SXY}[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$ *is* SPR-CCA*-secure.*
Formally speaking, for any $\mathcal{A}$ *against the* SPR-CCA *security of* KEM *issuing at most* q_{DEC} *queries to the decapsulation oracle and* q_{G}, q_{H}, *and* $q_{\mathsf{H}_{\mathsf{prf}}}$ *queries to* G, H, *and* $\mathsf{H}_{\mathsf{prf}}$, *respectively, there exists* $\mathcal{A}_{34}$ *against ciphertext-indistinguishability of* PKE *such that*

$$\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A},\mathcal{S}}(\kappa) \le \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4\delta + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{DEC}}) \cdot 2^{-\ell/2}.$$

For simplicity, we here prove Theorem 4.2 because it is simple and suffices for the NTRU case. We give the security proof of Theorem 4.1 in subsection B.2.

Proof of Theorem 4.2: We use the game-hopping proof. We consider Game_i for $i = 0, \ldots, 8$. We summarize the games in Table 3. Let S_i denote the event that the adversary outputs $b' = 1$ in game Game_i. Let Acc be an event that a key pair (ek, dk) is accurate. Let $\neg$Acc denote the event that a key pair (ek, dk) is inaccurate. We note that we have $\Pr[\neg\mathsf{Acc}] \le \delta$ since PKE is deterministic. We extend the security proof for IND-CCA security of SXY in [SXY18, XY19, LW21].

Game_0: This game is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 0$. Thus, we have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Table 3. Summary of games for the proof of Theorem 4.2

Game	H	c^*	K^*	Decapsulation valid c	Decapsulation invalid c	Justification
Game_0	$\mathsf{H}(\cdot)$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s,c)$	
Game_1	$\mathsf{H}(\cdot)$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	Lemma 2.2
$\mathsf{Game}_{1.5}$	$\mathsf{H}'_q(\mathsf{Enc}(ek,\cdot))$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	a key pair's accuracy
Game_2	$\mathsf{H}_q(\mathsf{Enc}(ek,\cdot))$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	a key pair's accuracy
Game_3	$\mathsf{H}_q(\mathsf{Enc}(ek,\cdot))$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	a key pair's accuracy
Game_4	$\mathsf{H}_q(\mathsf{Enc}(ek,\cdot))$	$\mathcal{S}(1^\kappa)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	ciphertext indistinguishability
Game_5	$\mathsf{H}_q(\mathsf{Enc}(ek,\cdot))$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	statistical disjointness
Game_6	$\mathsf{H}_q(\mathsf{Enc}(ek,\cdot))$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	a key pair's accuracy
$\mathsf{Game}_{6.5}$	$\mathsf{H}'_q(\mathsf{Enc}(ek,\cdot))$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	a key pair's accuracy
Game_7	$\mathsf{H}(\cdot)$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	a key pair's accuracy
Game_8	$\mathsf{H}(\cdot)$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s,c)$	Lemma 2.2

Game_1: This game is the same as Game_0 except that $\mathsf{H}_{\mathsf{prf}}(s,c)$ in the decapsulation oracle is replace with $\mathsf{H}_q(c)$ where $\mathsf{H}_q : \mathcal{C} \to \mathcal{K}$ is another random oracle. We remark that $\mathcal{A}$ cannot access H_q directly.
As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_0] - \Pr[S_1]| \leq 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{DEC}}) \cdot 2^{-\ell/2},$$

where $q_{\mathsf{H}_{\mathsf{prf}}}$ and q_{DEC} denote the number of queries to $\mathsf{H}_{\mathsf{prf}}$ and DEC the adversary makes, respectively.
In addition, according to Lemma A.1, for any $p \geq 0$, we have

$$|\Pr[S_1] - p| \leq |\Pr[S_1 \wedge \mathrm{Acc}] - p| + \delta.$$

$\mathsf{Game}_{1.5}$: This game is the same as Game_1 except that the random oracle $\mathsf{H}(\cdot)$ is simulated by $\mathsf{H}'_q(\mathsf{Enc}(ek,\cdot))$ where $\mathsf{H}'_q : \mathcal{C} \to \mathcal{K}$ is yet another random oracle. We remark that the decapsulation oracle and the generation of K^* also use $\mathsf{H}'_q(\mathsf{Enc}(ek,\cdot))$ as $\mathsf{H}(\cdot)$.
If the key pair (ek, dk) is accurate, then $g(\mu) := \mathsf{Enc}(ek,\mu)$ is *injective*. Thus, if the key pair is accurate, then $\mathsf{H}'_q \circ g : \mathcal{M} \to \mathcal{K}$ is a random function and the two games Game_1 and $\mathsf{Game}_{1.5}$ are equal to each other. Thus, we have

$$\Pr[S_1 \wedge \mathrm{Acc}] = \Pr[S_{1.5} \wedge \mathrm{Acc}].$$

Game_2: This game is the same as $\mathsf{Game}_{1.5}$ except that the random oracle H is simulated by $\mathsf{H}_q \circ g$ instead of $\mathsf{H}'_q \circ g$.
A ciphertext c is said to be *valid* if we have $\mathsf{Enc}(ek, \mathsf{Dec}(dk,c)) = c$ and *invalid* otherwise.
Notice that, in $\mathsf{Game}_{1.5}$, H_q is used for *invalid* ciphertext, and an adversary cannot access a value of H_q for a valid ciphertext. In addition, in $\mathsf{Game}_{1.5}$, an adversary can access a value of H'_q on input a valid ciphertext and cannot access a value of H'_q on input an invalid ciphertext if the key pair is accurate. Thus, there is no difference between $\mathsf{Game}_{1.5}$ and Game_2 if the key pair is accurate and we have

$$\Pr[S_{1.5} \wedge \mathrm{Acc}] = \Pr[S_2 \wedge \mathrm{Acc}].$$

Game_3: This game is the same as Game_2 except that K^* is set as $\mathsf{H}_q(c^*)$ and the decapsulation oracle always returns $\mathsf{H}_q(c)$ as long as $c \neq c^*$.
If the key pair is accurate, for a valid ciphertext c and its decrypted result μ, we have $\mathsf{H}(\mu) = \mathsf{H}_q(\mathsf{Enc}(ek,\mu)) = \mathsf{H}_q(c)$. Thus, the two games Game_2 and Game_3 are equal to each other and we have

$$\Pr[S_2 \wedge \mathrm{Acc}] = \Pr[S_3 \wedge \mathrm{Acc}].$$

According to Lemma A.1, for any $p \geq 0$, we have

$$|\Pr[S_3 \wedge \mathrm{Acc}] - p| \leq |\Pr[S_3] - p| + \delta.$$

Game_4: This game is the same as Game_3 except that c^* is generated by $\mathcal{S}(1^\kappa)$.
The difference between two games Game_3 and Game_4 is bounded by the advantage of ciphertext indistinguishability in disjoint simulatability as in [XY19, Lemma 4.7]. The reduction algorithm is obtained straightforwardly, and we omit it. We have

$$|\Pr[S_3] - \Pr[S_4]| \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa).$$

Game_5: This game is the same as Game_4 except that $K^* \leftarrow \mathcal{K}$ instead of $K^* \leftarrow \mathsf{H}_q(c^*)$.
In Game_4, if $c^* \leftarrow \mathcal{S}(1^\kappa)$ is not in $\mathsf{Enc}(ek, \mathcal{M})$, then the adversary has no information about $K^* = \mathsf{H}_q(c^*)$ and thus, K^* looks uniformly at random. Hence, the difference between two games Game_4 and Game_5 is bounded by the statistical disjointness in disjoint simulatability as in [XY19, Lemma 4.8]. We have

$$|\Pr[S_4] - \Pr[S_5]| \leq \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

According to Lemma A.1, for any $p \geq 0$, we have

$$|\Pr[S_5] - p| \leq |\Pr[S_5 \wedge \mathsf{Acc}] - p| + \delta.$$

Game_6: This game is the same as Game_5 except that the decapsulation oracle is reset as $\textsc{Dec}$. Similar to the case for Game_2 and Game_3, if a key pair is accurate, the two games Game_5 and Game_6 are equal to each other as in the proof of [XY19, Lemma 4.5]. We have

$$\Pr[S_5 \wedge \mathsf{Acc}] = \Pr[S_6 \wedge \mathsf{Acc}].$$

$\mathsf{Game}_{6.5}$: This game is the same as Game_6 except that the random oracle H is simulated by $\mathsf{H}'_q \circ g$ where $\mathsf{H}'_q : \mathcal{C} \to \mathcal{K}$ is yet another random oracle as in $\mathsf{Game}_{1.2}$ instead of $\mathsf{H}_q \circ g$. If a key pair is accurate, then two games Game_6 and $\mathsf{Game}_{6.5}$ are equal to each other as the two games $\mathsf{Game}_{1.5}$ and Game_2 are equal to each other. We have

$$\Pr[S_6 \wedge \mathsf{Acc}] = \Pr[S_{6.5} \wedge \mathsf{Acc}].$$

Game_7: This game is the same as $\mathsf{Game}_{6.5}$ except that the random oracle $\mathsf{H}(\cdot)$ is set as the original. If a key pair is accurate, then the two games $\mathsf{Game}_{6.5}$ and Game_7 are equal to each other as the two games $\mathsf{Game}_{1.5}$ and Game_1 are equal to each other. We have

$$\Pr[S_{6.5} \wedge \mathsf{Acc}] = \Pr[S_7 \wedge \mathsf{Acc}].$$

According to Lemma A.1, for any $p \geq 0$, we have

$$|\Pr[S_7 \wedge \mathsf{Acc}] - p| \leq |\Pr[S_7] - p| + \delta.$$

Game_8: This game is the same as Game_7 except that $\mathsf{H}_q(c)$ in the decapsulation oracle is replaced by $\mathsf{H}_{\mathsf{prf}}(s, c)$.
As we discussed the difference between the two games Game_0 and Game_1, from Lemma 2.2 we have the bound

$$|\Pr[S_7] - \Pr[S_8]| \leq 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\textsc{Dec}}) \cdot 2^{-\ell/2}.$$

We note that this game is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 1$. Thus, we have

$$\Pr[S_8] = \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing those (in)equalities, we obtain the following bound:

$$\begin{aligned}\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = |\Pr[S_0] - \Pr[S_8]| &\leq \sum_{i=0}^{7} |\Pr[S_i] - \Pr[S_{i+1}]| \\ &\leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\textsc{Dec}}) \cdot 2^{-\ell/2} + 4\delta.\end{aligned}$$

4.2 SSMT-CCA Security

We next show that KEM is strongly smooth if the underlying PKE is strongly disjoint-simulatable.

Theorem 4.3. *Suppose that a ciphertext space $\mathcal{C}$ of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{SXY}[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$ *is* SSMT-CCA*-secure.*
Formally speaking, for any adversary $\mathcal{A}$ against SSMT-CCA *security of* KEM *issuing at most $q_{\mathsf{H}_{\mathsf{prf}}}$ and $q_{\textsc{Dec}}$ queries to $\mathsf{H}_{\mathsf{prf}}$ and* $\textsc{Dec}$*, we have*

$$\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) \leq 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\textsc{Dec}}) \cdot 2^{-\ell/2}.$$

Note that this bound is independent of whether PKE is deterministic or derandomized by T.

Table 4. Summary of games for the proof of Theorem 4.3: '$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$' implies that the challenger generates $c^* \leftarrow \mathcal{S}(1^\kappa)$ and returns $\perp$ if $c^* \in \mathsf{Enc}(ek, \mathcal{M})$.

Game	c^*	K^*	Decapsulation: valid c	Decapsulation: invalid c	Justification
Game_0	$\mathcal{S}(1^\kappa)$	random	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c)$	
Game_1	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	random	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c)$	statistical disjointness
Game_2	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	random	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	Lemma 2.2
Game_3	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$\mathsf{H}_q(c^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c^*)$ is hidden
Game_4	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$\mathsf{H}_{\mathsf{prf}}(s, c^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c)$	Lemma 2.2
Game_5	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$\overline{\mathsf{Dec}}(dk, c^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c)$	re-encryption check
Game_6	$\mathcal{S}(1^\kappa)$	$\overline{\mathsf{Dec}}(dk, c^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c)$	statistical disjointness

Proof: We use the game-hopping proof. We consider Game_i for $i = 0, \ldots, 6$. We summarize those games in Table 4. Let S_i denote the event that the adversary outputs $b' = 1$ in game Game_i.

Game_0: This game is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa)$ with $b = 0$. The challenge is generated as $c^* \leftarrow \mathcal{S}(1^\kappa)$ and $K^*_0 \leftarrow \mathcal{K}$. We have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: In this game, the challenge ciphertext is set as $\perp$ if c^* is in $\mathsf{Enc}(ek, \mathcal{M})$. Since the difference between two games Game_0 and Game_1 is bounded by statistical disjointness, we have

$$|\Pr[S_0] - \Pr[S_1]| \le \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Game_2: This game is the same as Game_1 except that $\mathsf{H}_{\mathsf{prf}}(s, c)$ in the decapsulation oracle is replace with $\mathsf{H}_q(c)$ where $\mathsf{H}_q : \mathcal{C} \to \mathcal{K}$ is another random oracle.
As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_1] - \Pr[S_2]| \le 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{DEC}}) \cdot 2^{-\ell/2}.$$

Game_3: This game is the same as Game_2 except that K^* is set as $\mathsf{H}_q(c^*)$ instead of chosen randomly. Since c^* is always outside of $\mathsf{Enc}(ek, \mathcal{M})$, $\mathcal{A}$ cannot obtain any information about $\mathsf{H}_q(c^*)$. Hence, the two games Game_2 and Game_3 are equal to each other and we have

$$\Pr[S_2] = \Pr[S_3].$$

Game_4: This game is the same as Game_3 except that $\mathsf{H}_q(\cdot)$ is replaced by $\mathsf{H}_{\mathsf{prf}}(s, \cdot)$. As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_3] - \Pr[S_4]| \le 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{DEC}}) \cdot 2^{-\ell/2}.$$

Game_5: This game is the same as Game_4 except that K^* is set as $\overline{\mathsf{Dec}}(dk, c^*)$ instead of $\mathsf{H}_{\mathsf{prf}}(s, c^*)$. Recall that c^* is always in *outside* of $\mathsf{Enc}(ek, \mathcal{M})$. Thus, we always have $\mathsf{Dec}(c^*) = \perp$ or $\mathsf{Enc}(ek, \mathsf{Dec}(c^*)) \neq c^*$ and, thus, $K^* = \mathsf{H}_{\mathsf{prf}}(s, c^*)$ in Game_5. Hence, the two games are equal to each other and we have

$$\Pr[S_4] = \Pr[S_5].$$

Game_6: We finally replace the way to compute c^*: In this game, the ciphertext is chosen by $\mathcal{S}(1^\kappa)$ as in Game_0. Again, since the difference between two games Game_5 and Game_6 is bounded by statistical disjointness, we have

$$|\Pr[S_5] - \Pr[S_6]| \le \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Moreover, this game Game_6 is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa)$ with $b = 1$ and we have

$$\Pr[S_6] = \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing those (in)equalities, we obtain Theorem 4.3:

$$\begin{aligned}\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_6]| \\ &\le 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{DEC}}) \cdot 2^{-\ell/2}.\end{aligned}$$

4.3 SCFR-CCA Security

Finally, we show that KEM is strongly collision-free if the underlying PKE is strongly collision-free or extended collision-free.

Theorem 4.4. *If* PKE *is* SCFR-CCA*-secure (or* XCFR*-secure), then* $\mathsf{KEM} = \mathsf{SXY}[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$ *is* SCFR-CCA*-secure in the QROM.*

Proof. Suppose that an adversary against KEM's SCFR-CCA security outputs a ciphertext c which is decapsulated into $K \neq \bot$ by both $\overline{dk}_0$ and $\overline{dk}_1$, that is, $K = \overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c) \neq \bot$. For $i \in \{0,1\}$, we define μ'_i as an internal decryption result under dk_i, that is, $\mu'_i = \mathsf{Dec}(dk_i, c)$. For $i \in \{0,1\}$, we also define $\mu_i := \mu'_i$ if $c = \mathsf{Enc}(ek_i, \mu'_i)$ and $\mu_i := \bot$ otherwise.
We have five cases classified as follows:

- Case 1 ($\mu_0 = \mu_1 \neq \bot$): The condition that $\mu_0 = \mu_1 \neq \bot$ violates the SCFR-CCA security (or the XCFR security) of the underlying PKE and it is easy to make a reduction.
- Case 2 ($\bot \neq \mu_0 \neq \mu_1 \neq \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0) = \mathsf{H}(\mu_1)$ and we find a collision for H. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).
- Case 3 ($\mu_0 = \bot$ and $\mu_1 \neq \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}_{\mathsf{prf}}(s_0, c) = \mathsf{H}(\mu_1)$ and we find a claw $((s_0, c), \mu_1)$ of $\mathsf{H}_{\mathsf{prf}}$ and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
- Case 4 ($\mu_0 \neq \bot$ and $\mu_1 = \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0) = \mathsf{H}_{\mathsf{prf}}(s_1, c)$ and we find a claw $(\mu_0, (s_1, c))$ of H and $\mathsf{H}_{\mathsf{prf}}$. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
- Case 5 (The other cases): In this case, we find a collision $((s_0, c), (s_1, c))$ of $\mathsf{H}_{\mathsf{prf}}$, which is indeed collision if $s_0 \neq s_1$ which occurs with probability at lease $1 - 1/2^\ell$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

We conclude that the advantage of the adversary is negligible in any case. □

5 NTRU

We briefly review NTRU [CDH+20] in subsection 5.1, discuss the security properties of the underlying PKE, NTRU-DPKE, in subsection 5.2, and discuss the security properties of NTRU in subsection 5.3. We want to show that, under appropriate assumptions, NTRU is ANON-CCA-secure in the QROM, and NTRU leads to ANON-CCA-secure and SROB-CCA-secure hybrid PKE in the QROM. In order to do so, we show that the underlying NTRU-DPKE of NTRU is strongly disjoint-simulatable under the modified DSPR and PLWE assumptions and XCFR-secure in subsection 5.2. Since NTRU is obtained by applying SXY to NTRU-DPKE, the former implies that NTRU is SPR-CCA-secure and SSMT-CCA-secure in the QROM under those assumptions and the latter implies that NTRU is SCFR-CCA-secure in the QROM. Those three properties lead to the anonymity of NTRU and hybrid PKE in the QROM as we wanted.

5.1 Review of NTRU

Preliminaries: Φ_1 denotes the polynomial $x - 1$ and Φ_n denotes $(x^n - 1)/(x - 1) = x^{n-1} + x^{n-2} + \cdots + 1$. We have $x^n - 1 = \Phi_1\Phi_n$. R, $R/3$, and R/q denotes $\mathbb{Z}[x]/(\Phi_1\Phi_n)$, $\mathbb{Z}[x]/(3, \Phi_1\Phi_n)$, and $\mathbb{Z}[x]/(q, \Phi_1\Phi_n)$, respectively. S, $S/3$, and S/q denotes $\mathbb{Z}[x]/(\Phi_n)$, $\mathbb{Z}[x]/(3, \Phi_n)$, and $\mathbb{Z}[x]/(q, \Phi_n)$, respectively.
We say a polynomial *ternary* if its coefficients are in $\{-1, 0, +1\}$. $\underline{S3}(a)$ returns a canonical $S/3$-representative of $z \in \mathbb{Z}[x]$, that is, $b \in \mathbb{Z}[x]$ of degree at most $n - 2$ with ternary coefficients in $\{-1, 0, +1\}$ such that $a \equiv b \pmod{3, \Phi_n}$. Let $\mathcal{T}$ be a set of non-zero ternary polynomials of degree at most $n - 2$, that is, $\mathcal{T} = \{a = \sum_{i=0}^{n-2} a_i x^i : a \neq 0 \wedge a_i \in \{-1, 0, +1\}\}$. We say a ternary polynomial $v = \sum_i v_i x^i$ has the *non-negative correlation* property if $\sum_i v_i v_{i+1} \geq 0$. $\mathcal{T}_+$ is a set of non-zero ternary polynomials of degree at most $n - 2$ with *non-negative correlation* property. $\mathcal{T}(d)$ is a set of non-zero balanced ternary polynomials of degree at most $n - 2$ with Hamming weight d, that is, $\{a \in \mathcal{T} : |\{a_i : a_i = 1\}| = |\{a_i : a_i = -1\}| = d/2\}$.
The following lemma is due to Schanck [Sch21]. (See, e.g., [CDH+20] for this design choice.)

Lemma 5.1. *Suppose that* $(n, q) = (509, 2048), (677, 2048), (821, 4096),$ *or* $(701, 8192)$*, which are the parameter sets in NTRU. If* $r \in \mathcal{T}$*, then* r *has an inverse in* S/q*.*

Proof. Φ_n is irreducible over $\mathbb{F}_2$ if and only if n is prime and 2 is primitive element in $\mathbb{F}_n^\times$ (See e.g., Cohen et al. [CFA+05]). The conditions are satisfied for all $n = 509$, 677, 701, and 821. Hence, $\mathbb{Z}[x]/(2, \Phi_n)$ is a finite field and every polynomial r in $\mathcal{T}$ has an inverse in $\mathbb{Z}[x]/(2, \Phi_n)$. Such r is also invertible in $S/q = \mathbb{Z}[x]/(q, \Phi_n)$ with $q = 2^k$ for some k and, indeed, one can find it using the Newton method or the Hensel lifting. □

$\mathsf{Gen}(1^\kappa)$	$\mathsf{Enc}(h, (r,m) \in \mathcal{L}_r \times \mathcal{L}_m)$	$\mathsf{Dec}((f, f_p, h_q), c)$
$(f, g) \leftarrow \mathsf{Sample\_fg}()$	$\mu' \coloneqq \mathsf{Lift}(m)$	**if** $c \not\equiv 0 \bmod (q, \Phi_1)$
$f_q \coloneqq (1/f) \in S/q$	$c \coloneqq (h \cdot r + \mu') \in R/q$	**then return** $(0,0,1)$
$h \coloneqq (3 \cdot g \cdot f_q) \in R/q$	**return** c	$a \coloneqq (c \cdot f) \in R/q$
$h_q \coloneqq (1/h) \in S/q$		$m \coloneqq (a \cdot f_p) \in S/3$
$f_p \coloneqq (1/f) \in S/3$		$\mu' \coloneqq \mathsf{Lift}(m)$
$ek \coloneqq h, dk \coloneqq (f, f_p, h_q)$		$r \coloneqq ((c - \mu') \cdot h_q) \in S/q$
return (ek, dk)		**if** $(r,m) \in \mathcal{L}_r \times \mathcal{L}_m$
		then return $(r, m, 0)$
		else return $(0,0,1)$

Fig. 6. NTRU-DPKE

NTRU: NTRU involves four subsets $\mathcal{L}_f, \mathcal{L}_g, \mathcal{L}_r, \mathcal{L}_m$ of R. It uses $\mathsf{Lift}(m) \colon \mathcal{L}_m \to R$. NTRU has two types of parameter sets, NTRU-HPS and NTRU-HRSS, specified as later.

- NTRU-HPS: The parameters are defined as follows: $\mathcal{L}_f = \mathcal{T}$, $\mathcal{L}_g = \mathcal{T}(q/8 - 2)$, $\mathcal{L}_r = \mathcal{T}$, $\mathcal{L}_m = \mathcal{T}(q/8 - 2)$, and $\mathsf{Lift}(m) = m$.
- NTRU-HRSS: The parameters are defined as follows: $\mathcal{L}_f = \mathcal{T}_+$, $\mathcal{L}_g = \{\Phi_1 \cdot v \mid v \in \mathcal{T}_+\}$, $\mathcal{L}_r = \mathcal{T}$, $\mathcal{L}_m = \mathcal{T}$, and $\mathsf{Lift}(m) = \Phi_1 \cdot \underline{\mathsf{S3}}(m/\Phi_1)$.

It uses $\mathsf{Sample\_fg}()$ to sample f and g from $\mathcal{L}_f$ and $\mathcal{L}_g$. NTRU also uses $\mathsf{Sample\_rm}()$ to sample r and m from $\mathcal{L}_r$ and $\mathcal{L}_m$.

The underlying DPKE of NTRU, which we call NTRU-DPKE, is defined as Figure 6. We note that, for an encryption key h, we have $h \equiv 0 \pmod{(q, \Phi_1)}$, h is invertible in S/q, and $hr + m \equiv 0 \pmod{(q, \Phi_1)}$. (See [CDH+20, Section2.3].)

NTRU then applies SXY to NTRU-DPKE in order to obtain IND-CCA-secure KEM as in Figure 7, where $\mathsf{H} = \text{SHA3-256}$ and $\mathsf{H}_{\mathsf{prf}} = \text{SHA3-256}$. Since the lengths of their input spaces differ, we can treat them as different random oracles.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek = h)$	$\overline{\mathsf{Dec}}(\overline{dk} = (dk, s), c)$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	$\mathsf{coins} \leftarrow \{0,1\}^{256}$	$(r, m, \mathsf{fail}) \coloneqq \mathsf{Dec}(dk, c)$
$s \leftarrow \{0,1\}^{256}$	$(r,m) \leftarrow \mathsf{Sample\_rm}(\mathsf{coins})$	$k_1 \coloneqq \mathsf{H}(r, m)$
$\overline{dk} \coloneqq (dk, s)$	$c \coloneqq \mathsf{Enc}(h, (r,m))$	$k_2 \coloneqq \mathsf{H}_{\mathsf{prf}}(s, c)$
return $(ek, \overline{dk})$	$K \coloneqq \mathsf{H}(r, m)$	**if** $\mathsf{fail} = 0$ **then return** k_1
	return (c, K)	**else return** k_2

Fig. 7. NTRU

Rigidity: NTRU uses SXY, while its KEM version (Figure 7) seems to lack the re-encryption check. We note that NTRU implicitly checks $hr + \mathsf{Lift}(m) = c$ by checking if $(r,m) \in \mathcal{L}_r \times \mathcal{L}_m$ in NTRU-DPKE (Figure 6). See [CDH+20] for the details.

5.2 Properties of NTRU-DPKE

We show that NTRU-DPKE is strongly disjoint-simulatable and XCFR-secure.

We have known that the generalized NTRU PKE is pseudorandom [SS10] and disjoint-simulatable [SXY18] if the decisional small polynomial ratio (DSPR) assumption [LTV12] and the polynomial learning with errors (PLWE) assumption [SSTX09, LPR10] hold. See [SXY18, Section 3.3 of the ePrint version].

Let us adapt their arguments to NTRU-DPKE. We modify the DSPR and the PLWE assumptions as follows:

Definition 5.1. *Fix the parameter set. Define* $R' \coloneqq \{c \in R/q : c \equiv 0 \pmod{(q, \Phi_1)}\}$*, which is efficiently sampleable.*

- *The modified DSPR assumption: It is computationally hard to distinguish* $h := 3 \cdot g \cdot f_q \pmod{q, \Phi_1\Phi_n}$ *from* h'*, where* $(f, g) \leftarrow \mathsf{Sample\_fg}()$*,* $f_q \leftarrow (1/f) \bmod (q, \Phi_n)$*, and* $h' \leftarrow R'$*.*
- *The modified PLWE assumption: It is computationally hard to distinguish* $(h, hr + \mathsf{Lift}(m) \pmod{q, \Phi_1\Phi_n})$ *from* (h, c') *with* $h, c' \leftarrow R'$ *and* $(r, m) \leftarrow \mathsf{Sample\_rm}()$*.*

We can show NTRU-DPKE is strongly disjoint-simulatable under those two assumptions:

Lemma 5.2. *Suppose that the modified DSPR and PLWE assumptions hold. Then, NTRU-DPKE is strongly disjoint-simulatable with a simulator* $\mathcal{S}$ *that outputs a random polynomial chosen from* R'*.*

Proof. The proof for ciphertext-indistinguishability is obtained by modifying the proof in [SXY18]. We want to show that $(h, c = hr + \mathsf{Lift}(m) \bmod (q, \Phi_1\Phi_n)) \approx_c (h, c')$, where $h = 3gf_q \bmod (q, \Phi_1\Phi_n)$ and $f_q = (1/f) \bmod (q, \Phi_n)$ with $(f, g) \leftarrow \mathsf{Sample\_fg}()$, $(r, m) \leftarrow \mathsf{Sample\_rm}()$, and $c' \leftarrow R'$.
- We first replace h with $h' \leftarrow R'$, which is justified by the modified DSPR assumption.
- We next replace $c = h'r + \mathsf{Lift}(m) \bmod (q, \Phi_1\Phi_n)$ with $c' \leftarrow R'$, which is justified by the modified PLWE assumption.
- We then go backward by replacing random h' with h, which is is justified by the modified DSPR assumption again.

Statistical disjointness follows from the fact that $|R'| = q^{n-1} \gg 3^{2n} = |\mathcal{T} \times \mathcal{T}| \geq |\mathcal{L}_m \times \mathcal{L}_r| \geq |\mathsf{Enc}(h, \mathcal{L}_m \times \mathcal{L}_r)|$. Since R' is independent of an encryption key h, NTRU-DPKE is strong disjoint-simulatability. □

We next show the XCFR security of NTRU-DPKE.

Lemma 5.3. *NTRU-DPKE is* XCFR*-secure.*

Proof. Suppose that the adversary wins with its output c on input ek_0, dk_0, ek_1, and dk_1, where $ek_i = h_i$ for $i \in \{0, 1\}$. Let us define $\mu_0 = \mathsf{Dec}(dk_0, c)$ and $\mu_1 = \mathsf{Dec}(dk_1, c)$.
If the adversary wins, we can assume $\mu_0 = \mu_1 = (r, m, 0) \in \mathcal{L}_r \times \mathcal{L}_m \times \{0, 1\}$. Otherwise, that is, if $\mu_0 = \mu_1 = (0, 0, 1)$, then the output is treated as $\perp$ and the adversary loses.
Moreover, because of the check in the decryption, we have $c \equiv h_0 \cdot r + \mathsf{Lift}(m) \equiv h_1 \cdot r + \mathsf{Lift}(m) \pmod{q, \Phi_1\Phi_n}$, which implies $r(h_0 - h_1) \equiv 0 \pmod{(q, \Phi_n)}$. On the other hand, according to Lemma 5.1, for any $r \in \mathcal{L}_r = \mathcal{T}$, we have $r \neq 0 \in S/q$ In addition, we have $h_0 \equiv h_1 \in S/q$ with negligible probability. Thus, all but negligible choices of h_0 and h_1, any $r \in \mathcal{L}_r = \mathcal{T}$ results in $r(h_0 - h_1) \not\equiv 0 \pmod{(q, \Phi_n)}$ and $h_0 \cdot r + \mathsf{Lift}(m) \not\equiv h_1 \cdot r + \mathsf{Lift}(m) \pmod{q, \Phi_1\Phi_n}$. Hence, the probability that the adversary wins is negligible, concluding the proof. □

5.3 Properties of NTRU

Combining NTRU-DPKE's strong disjoint-simulatability and XCFR security with previous theorems on SXY, we obtain the following theorems.

Theorem 5.1. *Suppose that the modified DSPR and PLWE assumptions hold. Then, NTRU is* SPR-CCA*-secure and* SSMT-CCA*-secure in the QROM.*

Proof. Under the modified DSPR and PLWE assumptions, NTRU-DPKE is strongly disjoint-simulatable (Lemma 5.2). In addition, NTRU-DPKE is perfectly correct. Applying Theorem 4.2 and Theorem 4.3, we obtain the theorem. □

Theorem 5.2. *NTRU is* SCFR-CCA*-secure in the QROM.*

Proof. NTRU-DPKE is XCFR-secure (Lemma 5.3). Applying Theorem 4.4, we have that NTRU is SCFR-CCA-secure in the QROM. □

Theorem 5.3. *Under the modified DSPR and PLWE assumptions, NTRU is* ANON-CCA*-secure in the QROM.*

Proof. Due to Theorem 5.1, under the modified DSPR and PLWE assumptions, NTRU is SPR-CCA-secure in the QROM. Thus, applying Theorem 2.5, we have that, under those assumptions, NTRU is ANON-CCA-secure in the QROM. □

Theorem 5.4. *Under the modified DSPR and PLWE assumptions, NTRU leads to* ANON-CCA*-secure and* SROB-CCA*-secure hybrid PKE in the QROM, combined with* SPR-OTCCA*-secure and* FROB*-secure DEM.*

Proof. Due to Theorem 5.1, under the modified DSPR and PLWE assumptions, NTRU is SPR-CCA-secure and SSMT-CCA-secure in the QROM. Moreover, NTRU is perfectly correct. Thus, combining NTRU with SPR-OTCCA-secure DEM, we obtain a SPR-CCA-secure hybrid PKE in the QROM (Theorem 3.2). Moreover, NTRU is SCFR-CCA-secure in the QROM (Theorem 5.2). Thus, if DEM is FROB-secure, then the hybrid PKE is SROB-CCA-secure (Theorem 2.2). □

Acknowledgement

The author is grateful to John Schanck for insightful comments and suggestions on NTRU, Akinori Hosoyamada and Takashi Yamakawa for insightful comments and discussion on quantum random oracles, and Kohei Nakagawa for discussion on the collision problem in SIKE. The author would like to thank Daniel J. Bernstein for insightful comments and discussion on the indifferentiability of the quantum random oracles. The author would like to thank anonymous reviewers for their valuable comments and suggestions on this paper.

References

AAB+20. Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Edoardo Persichetti, Gilles Zémor, and Jurjen Bos. HQC. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 58, 59

ABB+20. Nicolas Aragon, Paulo Barreto, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Phillipe Gaborit, Shay Gueron, Tim Guneysu, Carlos Aguilar Melchor, Rafael Misoczki, Edoardo Persichetti, Nicolas Sendrier, Jean-Pierre Tillich, Gilles Zémor, Valentin Vasseur, and Santosh Ghosh. BIKE. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 53, 54, 55

ABC+05. Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 205–222. Springer, Heidelberg, August 2005. 1

ABC+20. Martin R. Albrecht, Daniel J. Bernstein, Tung Chou, Carlos Cid, Jan Gilcher, Tanja Lange, Varun Maram, Ingo von Maurich, Rafael Misoczki, Ruben Niederhagen, Kenneth G. Paterson, Edoardo Persichetti, Christiane Peters, Peter Schwabe, Nicolas Sendrier, Jakub Szefer, Cen Jung Tjhai, Martin Tomlinson, and Wen Wang. Classic McEliece. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 49, 50

Abe10. Masayuki Abe, editor. *ASIACRYPT 2010*, volume 6477 of *LNCS*. Springer, Heidelberg, December 2010. 23

ABN10. Michel Abdalla, Mihir Bellare, and Gregory Neven. Robust encryption. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 480–497. Springer, Heidelberg, February 2010. 1

AHU19. Andris Ambainis, Mike Hamburg, and Dominique Unruh. Quantum security proofs using semi-classical oracles. In Boldyreva and Micciancio [BM19], pages 269–295. 24, 33

BBC+20. Daniel J. Bernstein, Billy Bob Brumley, Ming-Shing Chen, Chitchanok Chuengsatiansup, Tanja Lange, Adrian Marotzke, Bo-Yuan Peng, Nicola Tuveri, Christine van Vredendaal, and Bo-Yin Yang. NTRU Prime. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 61, 62

BBDP01. Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In Colin Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 566–582. Springer, Heidelberg, December 2001. 1, 5

BCGNP09. Colin Boyd, Yvonne Cliff, Juan Manuel González Nieto, and Kenneth G. Paterson. One-round key exchange in the standard model. *Int. J. Appl. Cryptogr.*, 1(3):181–199, 2009. 1

BDF+11. Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Heidelberg, December 2011. 4

BDPR98. Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations among notions of security for public-key encryption schemes. In Hugo Krawczyk, editor, *CRYPTO'98*, volume 1462 of *LNCS*, pages 26–45. Springer, Heidelberg, August 1998. 5, 8

Ber21. Daniel J. Bernstein. personal communication, October 2021. 3, 62

BHH+19. Nina Bindel, Mike Hamburg, Kathrin Hövelmanns, Andreas Hülsing, and Edoardo Persichetti. Tighter proofs of CCA security in the quantum random oracle model. In Dennis Hofheinz and Alon Rosen, editors, *TCC 2019, Part II*, volume 11892 of *LNCS*, pages 61–90. Springer, Heidelberg, December 2019. 30, 31, 33, 42, 43, 46, 47

BM19. Alexandra Boldyreva and Daniele Micciancio, editors. *CRYPTO 2019, Part II*, volume 11693 of *LNCS*. Springer, Heidelberg, August 2019. 21, 24

CDH+20. Cong Chen, Oussama Danba, Jeffrey Hoffstein, Andreas Hulsing, Joost Rijneveld, John M. Schanck, Peter Schwabe, William Whyte, Zhenfei Zhang, Tsunekazu Saito, Takashi Yamakawa, and Keita Xagawa. NTRU. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 18, 19

CFA+05. Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Tanja Lange, Kim Nguyen, and Frederik Vercauteren. *Handbook of Elliptic and Hyperelliptic Curve Cryptography*. 2005. 18

CL01. Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In Birgit Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 93–118. Springer, Heidelberg, May 2001. 1

CS02. Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 45–64. Springer, Heidelberg, April / May 2002. 3, 8

CS03. Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33(1):167–226, 2003. 12, 13, 25, 26

DKR+20. Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, Frederik Vercauteren, Jose Maria Bermudo Mera, Michiel Van Beirendonck, and Andrea Basso. SABER. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 53

FNP14. Nelly Fazio, Antonio Nicolosi, and Irippuge Milinda Perera. Broadcast steganography. In Josh Benaloh, editor, *CT-RSA 2014*, volume 8366 of *LNCS*, pages 64–84. Springer, Heidelberg, February 2014. 66

FO99. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In Michael J. Wiener, editor, *CRYPTO'99*, volume 1666 of *LNCS*, pages 537–554. Springer, Heidelberg, August 1999. 1

FO13. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. *Journal of Cryptology*, 26(1):80–101, January 2013. 1

FOR17. Pooya Farshim, Claudio Orlandi, and Răzvan Roşie. Security of symmetric primitives under incorrect usage of keys. *IACR Trans. Symm. Cryptol.*, 2017(1):449–473, 2017. 8

FSXY13. Atsushi Fujioka, Koutarou Suzuki, Keita Xagawa, and Kazuki Yoneyama. Practical and post-quantum authenticated key exchange from one-way secure key encapsulation mechanism. In Kefei Chen, Qi Xie, Weidong Qiu, Ninghui Li, and Wen-Guey Tzeng, editors, *ASIACCS 13*, pages 83–94. ACM Press, May 2013. 1

FSXY15. Atsushi Fujioka, Koutarou Suzuki, Keita Xagawa, and Kazuki Yoneyama. Strongly secure authenticated key exchange from factoring, codes, and lattices. *Des. Codes Cryptogr.*, 76(3):469–504, 2015. 1

GMP21a. Paul Grubbs, Varun Maram, and Kenneth G. Paterson. Anonymous, robust post-quantum public key encryption. Cryptology ePrint Archive, Report 2021/708, 2021. https://eprint.iacr.org/2021/708. To appear in EUROCRYPT 2022. 1, 2, 3, 8, 11, 49, 51, 53, 54, 57, 58, 62, 66

GMP21b. Paul Grubbs, Varun Maram, and Kenneth G. Paterson. Anonymous, robust post-quantum public key encryption (presentation slides). The third NIST PQC Standardization Conference, 2021. https://csrc.nist.gov/Presentations/2021/anonymous-robust-post-quantum-public-key-encryptio. 2

HHK17. Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part I*, volume 10677 of *LNCS*, pages 341–371. Springer, Heidelberg, November 2017. 3, 5, 14, 24, 27, 29, 30, 53

HKSU20. Kathrin Hövelmanns, Eike Kiltz, Sven Schäge, and Dominique Unruh. Generic authenticated key exchange in the quantum random oracle model. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *PKC 2020, Part II*, volume 12111 of *LNCS*, pages 389–422. Springer, Heidelberg, May 2020. 4, 6, 27

Hop05. Nicholas Hopper. On steganographic chosen covertext security. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *ICALP 2005*, volume 3580 of *LNCS*, pages 311–323. Springer, Heidelberg, July 2005. 5, 8

Hos21. Akinori Hosoyamada. personal communication, June 2021. 5

IZ89. Russell Impagliazzo and David Zuckerman. How to recycle random bits. In *30th FOCS*, pages 248–253. IEEE Computer Society Press, October / November 1989. 66

JAC+20. David Jao, Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, Brian Koziel, Brian LaMacchia, Patrick Longa, Michael Naehrig, Joost Renes, Vladimir Soukharev, David Urbanik, Geovandro Pereira, Koray Karabina, and Aaron Hutchinson. SIKE. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 65

JD11. David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Bo-Yin Yang, editor, *Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011*, pages 19–34. Springer, Heidelberg, November / December 2011. 65, 66

JZC+18. Haodong Jiang, Zhenfeng Zhang, Long Chen, Hong Wang, and Zhi Ma. IND-CCA-secure key encapsulation mechanism in the quantum random oracle model, revisited. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 96–125. Springer, Heidelberg, August 2018. 4, 11, 27, 48

JZM19. Haodong Jiang, Zhenfeng Zhang, and Zhi Ma. Key encapsulation mechanism with explicit rejection in the quantum random oracle model. In Dongdai Lin and Kazue Sako, editors, *PKC 2019, Part II*, volume 11443 of *LNCS*, pages 618–645. Springer, Heidelberg, April 2019. 3, 30, 38, 39, 40

KSS+20. Veronika Kuchta, Amin Sakzad, Damien Stehlé, Ron Steinfeld, and Shifeng Sun. Measure-rewind-measure: Tighter quantum random oracle model proofs for one-way to hiding and CCA security. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 703–728. Springer, Heidelberg, May 2020. 33

LPR10. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer, Heidelberg, May / June 2010. 19

LTV12. Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In Howard J. Karloff and Toniann Pitassi, editors, *44th ACM STOC*, pages 1219–1234. ACM Press, May 2012. 19

LW21. Xu Liu and Mingqiang Wang. QCCA-secure generic key encapsulation mechanism with tighter security in the quantum random oracle model. In Juan Garay, editor, *PKC 2021, Part I*, volume 12710 of *LNCS*, pages 3–26. Springer, Heidelberg, May 2021. 6, 14, 27, 28, 34, 39

Moh10. Payman Mohassel. A closer look at anonymity and robustness in encryption schemes. In Abe [Abe10], pages 501–518. 1, 5, 11

MTSB13. Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *Proceedings of the 2013 IEEE International Symposium on Information Theory (ISIT), Istanbul, Turkey, July 7-12, 2013*, pages 2069–2073. IEEE, 2013. 54

NAB+20. Michael Naehrig, Erdem Alkim, Joppe Bos, Léo Ducas, Karen Easterbrook, Brian LaMacchia, Patrick Longa, Ilya Mironov, Valeria Nikolaenko, Christopher Peikert, Ananth Raghunathan, and Douglas Stebila. FrodoKEM. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 57

Per13. Edoardo Persichetti. Secure and anonymous hybrid encryption from coding theory. In Philippe Gaborit, editor, *Post-Quantum Cryptography - 5th International Workshop, PQCrypto 2013*, pages 174–187. Springer, Heidelberg, June 2013. 51

RS92. Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 433–444. Springer, Heidelberg, August 1992. 5, 8

SAB+20. Peter Schwabe, Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Gregor Seiler, and Damien Stehlé. CRYSTALS-KYBER. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 2, 52

Sak00. Kazue Sako. An auction protocol which hides bids of losers. In Hideki Imai and Yuliang Zheng, editors, *PKC 2000*, volume 1751 of *LNCS*, pages 422–432. Springer, Heidelberg, January 2000. 1

Sch21. John Schanck. personal communication, June 2021. 18

SS10. Damien Stehlé and Ron Steinfeld. Faster fully homomorphic encryption. In Abe [Abe10], pages 377–394. 19

SSTX09. Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient public key encryption based on ideal lattices. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 617–635. Springer, Heidelberg, December 2009. 19

SSW20. Peter Schwabe, Douglas Stebila, and Thom Wiggers. Post-quantum TLS without handshake signatures. In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *ACM CCS 2020*, pages 1461–1480. ACM Press, November 2020. 1

SXY18. Tsunekazu Saito, Keita Xagawa, and Takashi Yamakawa. Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part III*, volume 10822 of *LNCS*, pages 520–551. Springer, Heidelberg, April / May 2018. 3, 4, 6, 14, 19, 20, 27, 30, 55

TU16. Ehsan Ebrahimi Targhi and Dominique Unruh. Post-quantum security of the Fujisaki-Okamoto and OAEP transforms. In Martin Hirt and Adam D. Smith, editors, *TCC 2016-B, Part II*, volume 9986 of *LNCS*, pages 192–216. Springer, Heidelberg, October / November 2016. 30, 53

Unr15. Dominique Unruh. Revocable quantum timed-release encryption. *Journal of the ACM*, 62(6):No.49, 2015. The preliminary version appeared in *EUROCRYPT 2014*. See also https://eprint.iacr.org/2013/606. 24

vH04. Luis von Ahn and Nicholas J. Hopper. Public-key steganography. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 323–341. Springer, Heidelberg, May 2004. 5, 8

XY19. Keita Xagawa and Takashi Yamakawa. (Tightly) QCCA-secure key-encapsulation mechanism in the quantum random oracle model. In Jintai Ding and Rainer Steinwandt, editors, *Post-Quantum Cryptography - 10th International Conference, PQCrypto 2019*, pages 249–268. Springer, Heidelberg, 2019. 5, 14, 15, 16, 17, 27, 28, 29, 34, 35, 37, 39, 44, 48

Zha15. Mark Zhandry. A note on the quantum collision and set equality problems. *Quantum Info. Comput.*, 15(7–8):557–567, May 2015. 4

Zha19. Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Boldyreva and Micciancio [BM19], pages 239–268. 3, 62

A Missing Lemma

Lemma A.1. *Let* A *and* B *denote events. Suppose that we have* $\Pr[\mathsf{A}] \le \delta$. *For any* $p \ge 0$, *we have*

$$|\Pr[\mathsf{B}] - p| \le |\Pr[\mathsf{B} \wedge \neg\mathsf{A}] - p| + \delta \quad \textit{and} \quad |\Pr[\mathsf{B} \wedge \neg\mathsf{A}] - p| \le |\Pr[\mathsf{B}] - p| + \delta.$$

Proof. Those bounds are obtained by using the triangle inequality. We have

$$\begin{aligned}|\Pr[\mathsf{B}] - p| &= |\Pr[\mathsf{B} \wedge \mathsf{A}] + \Pr[\mathsf{B} \wedge \neg\mathsf{A}] - p| \le \Pr[\mathsf{B} \wedge \mathsf{A}] + |\Pr[\mathsf{B} \wedge \neg\mathsf{A}] - p| \\ &\le \Pr[\mathsf{A}] + |\Pr[\mathsf{B} \wedge \neg\mathsf{A}] - p| \le |\Pr[\mathsf{B} \wedge \neg\mathsf{A}] - p| + \delta\end{aligned}$$

and

$$\begin{aligned}|\Pr[\mathsf{B} \wedge \neg\mathsf{A}] - p| &= |\Pr[\mathsf{B} \wedge \neg\mathsf{A}] + \Pr[\mathsf{B} \wedge \mathsf{A}] - \Pr[\mathsf{B} \wedge \mathsf{A}] - p| \\ &= |\Pr[\mathsf{B}] - p - \Pr[\mathsf{B} \wedge \mathsf{A}]| \le |\Pr[\mathsf{B}] - p| + \Pr[\mathsf{B} \wedge \mathsf{A}] \\ &\le |\Pr[\mathsf{B}] - p| + \Pr[\mathsf{A}] \le |\Pr[\mathsf{B}] - p| + \delta\end{aligned}$$

as we wanted. □

The following lemma is called the oneway-to-hiding (O2H) lemma, which is proven by Unruh [Unr15, Lemma 6.2]. Roughly speaking, the lemma states that if any quantum adversary issuing at most q queries to a quantum random oracle H can distinguish $(x, \mathsf{H}(x))$ from (x, y), where y is chosen uniformly at random, then we can find x by measuring one of the adversary's queries. The following lemma is a generalized version of the O2H lemma taken from [AHU19].

Lemma A.2 (Oneway to Hiding [AHU19, Theorem 3]). *Let* $\mathcal{S} \subseteq \mathcal{X}$ *be random. Let* $\mathsf{G}, \mathsf{H} \colon \mathcal{X} \to \mathcal{Y}$ *be random functions satisfying* $\mathsf{G}(x) = \mathsf{H}(x)$ *for every* $x \notin \mathcal{S}$. *Let* z *be a random bit string. (*$\mathcal{S}, \mathsf{G}, \mathsf{H}, z$ *may have arbitrary joint distribution.)*
Let $\mathcal{A}$ *be a quantum oracle algorithm with query depth* d *(not necessarily unitary).*
Let $\mathcal{B}^{\mathsf{H}}$ *be an oracle algorithm that on input* z *does the following: pick* $i \leftarrow \{1, \dots, d\}$, *run* $\mathcal{A}^{\mathsf{H}}(z)$ *until (just before) the* i*-th query, measure all query input registers in the computational basis, output the set* $\mathcal{T} = \{t_1, \dots, t_{|\mathcal{T}|}\}$ *of measurement outcomes.*
Let

$$\begin{aligned}P_{\text{left}} &:= \Pr_{\mathsf{H},z}[b \leftarrow \mathcal{A}^{\mathsf{H}}(z) : b = 1], \\ P_{\text{right}} &:= \Pr_{\mathsf{G},z}[b \leftarrow \mathcal{A}^{\mathsf{G}}(z) : b = 1], \\ P_{\text{guess}} &:= \Pr_{\mathsf{H},\mathsf{G},\mathcal{S},z}[\mathcal{T} \leftarrow \mathcal{B}^{\mathsf{H}}(z) : \mathcal{S} \cap \mathcal{T} \neq \emptyset].\end{aligned}$$

Then,

$$\left|P_{\text{left}} - P_{\text{right}}\right| \le 2d\sqrt{P_{\text{guess}}} \quad \textit{and} \quad \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \le 2d\sqrt{P_{\text{guess}}}.$$

The same result holds with $\mathcal{B}^{\mathsf{G}}$ *instead of* $\mathcal{B}^{\mathsf{H}}$ *in the definition of* P_{guess}.

In this paper, we use lemma in [Unr15, HHK17] stated as follows:

Table 5. Summary of Games for the Proof of Theorem 3.1

Game	c^* and K^*	d^*	Deryption oracle	Justification
Game_0	$\overline{\mathsf{Enc}}(ek)$	$\mathsf{E}(K^*, \mu^*)$		
Game_1	$\overline{\mathsf{Enc}}(ek)$ at first	$\mathsf{E}(K^*, \mu^*)$		conceptual change
Game_2	$\overline{\mathsf{Enc}}(ek)$ at first	$\mathsf{E}(K^*, \mu^*)$	use K^* if $c = c^*$	δ-correctness of $\mathsf{KEM}^{\perp}$
Game_3	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$ at first	$\mathsf{E}(K^*, \mu^*)$	use K^* if $c = c^*$	SPR-CCA security of $\mathsf{KEM}^{\perp}$
Game_4	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$ at first	$U(C_{\lvert\mu^*\rvert})$	use K^* if $c = c^*$	SPR-OTCCA security of DEM
Game_5	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$ at first	$U(C_{\lvert\mu^*\rvert})$	use $\perp^*$ if $c = c^*$	INT-CTXT security of DEM
Game_6	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$ at first	$U(C_{\lvert\mu^*\rvert})$		ϵ-sparseness of $\mathsf{KEM}^{\perp}$
Game_7	$\mathcal{S}(1^\kappa) \times U(\mathcal{K})$	$U(C_{\lvert\mu^*\rvert})$		conceptual change

Corollary A.1 (Algorithmic Oneway-to-Hiding lemma). *Let $\mathsf{H} : \mathcal{X} \to \mathcal{Y}$ be a quantum random oracle, and let $\mathcal{A}$ be an adversary issuing at most q queries to H that on input $(x, y) \in \mathcal{X} \times \mathcal{Y}$ outputs either $0/1$. Let $\mathcal{D}_\mathcal{X}$ be a some distribution over $\mathcal{X}$.*
For all (probabilistic) algorithms F whose input space is $\mathcal{X} \times \mathcal{Y}$ and which do not make any hash queries to H, we have

$$\left|\begin{array}{l}\Pr[x \leftarrow \mathcal{D}_\mathcal{X}; y \leftarrow \mathsf{H}(x); \mathsf{inp} \leftarrow \mathsf{F}(x, y); b \leftarrow \mathcal{A}^{\mathsf{H}}(\mathsf{inp}) : b = 1] \\ \quad - \Pr[x \leftarrow \mathcal{D}_\mathcal{X}; y \leftarrow \mathcal{Y}; \mathsf{inp} \leftarrow \mathsf{F}(x, y); b \leftarrow \mathcal{A}^{\mathsf{H}}(\mathsf{inp}) : b = 1]\end{array}\right| \\ \leq 2q \cdot \sqrt{\Pr[x \leftarrow \mathcal{D}_\mathcal{X}; y \leftarrow \mathcal{Y}; \mathsf{inp} \leftarrow \mathsf{F}(x, y); x' \leftarrow \mathsf{EXT}^{\mathcal{A},\mathsf{H}}(\mathsf{inp}) : x' = x]},$$

where EXT picks $i \leftarrow \{1, \ldots, q\}$, runs $\mathcal{A}^{\mathsf{H}}(\mathsf{inp})$ until i-th query $\lvert\hat{x}\rangle$ to H, and returns $x' := \mathsf{Measure}(\lvert\hat{x}\rangle)$ (when $\mathcal{A}$ makes fewer than i queries, EXT outputs $\perp \notin \mathcal{X}$).

We can obtain the corollary by picking H uniformly at random, pick $x \leftarrow \mathcal{D}_\mathcal{X}$, pick y uniformly at random, set $\mathcal{S} := \{x\}$, $\mathsf{G}(x) := y$, and $z := \mathsf{inp} \leftarrow \mathsf{F}(x, \mathsf{H}(x))$. We then have

$$\begin{aligned} P_{\mathrm{right}} &= \Pr[b \leftarrow \mathcal{A}^{\mathsf{G}}(z) : b = 1] \\ &= \Pr[x \leftarrow \mathcal{D}_\mathcal{X}, y \leftarrow \mathcal{Y}, \mathsf{inp} \leftarrow \mathsf{F}(x, \mathsf{H}(x)), b \leftarrow \mathcal{A}^{\mathsf{G}}(\mathsf{inp}) : b = 1] \\ &= \Pr[x \leftarrow \mathcal{D}_\mathcal{X}, y \leftarrow \mathcal{Y}, \mathsf{inp} \leftarrow \mathsf{F}(x, y), b \leftarrow \mathcal{A}^{\mathsf{H}}(\mathsf{inp}) : b = 1]. \end{aligned}$$

The last equality follows from the fact that the distribution of $\mathsf{H}(x)$ and y are equivalent and we can switch them.

B Missing Proofs

B.1 Proof of Theorem 3.1

We consider Game_i for $i = 0, \ldots, 7$ defined later. We summarize the games in Table 5. Let S_i denote the event that the adversary outputs $b' = 1$ in Game_i.
Let $\mathcal{S}$ be the simulator for the SPR-CCA security of $\mathsf{KEM}^{\perp}$. We define $\mathcal{S}_{\mathsf{hy}}(1^\kappa, \lvert\mu^*\rvert) := \mathcal{S}(1^\kappa) \times U(C_{\lvert\mu^*\rvert})$ as the simulator for the SPR-CCA security of $\mathsf{PKE}_{\mathsf{hy}}$.
The security proof is similar to the security proof of the IND-CCA security of KEM/DEM [CS03] for Game_0, $\ldots$, Game_4. We need to take care of pseudorandom ciphertexts when moving from Game_4 to Game_7 and require the INT-CTXT security of DEM and the ϵ-sparseness of $\mathsf{KEM}^{\perp}$.

Game_0: This is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}},\mathcal{S}_{\mathsf{hy}},\mathcal{A}}(\kappa)$ with $b = 0$. Given μ^*, the target ciphertext is computed as follows:

$$(c^*, K^*) \leftarrow \overline{\mathsf{Enc}}(ek); d^* \leftarrow \mathsf{E}(K^*, \mu^*);\ \text{return } ct^* = (c^*, d^*).$$

We have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}},\mathcal{S}_{\mathsf{hy}},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: In this game, c^* and K^* are generated before invoking $\mathcal{A}$ with ek. This change is just conceptual, and we have

$$\Pr[S_0] = \Pr[S_1].$$

Game_2: In this game, the decryption oracle uses K^* if $c = c^*$ instead of $K = \overline{\mathsf{Dec}}(dk, c^*)$. Game_1 and Game_2 differ if correctly generated ciphertext c^* with K^* is decapsulated into different $K \neq K^*$ or $\perp$, which violates the correctness and occurs with probability at most δ. Hence, the difference of Game_1 and Game_2 is bounded by δ, and we have

$$|\Pr[S_1] - \Pr[S_2]| \leq \delta.$$

This bound is corresponding to the event BadKeyPair in [CS03].

Game_3: In this game, the challenger uses random (c^*, K^*) generated by the simulator and uses K^* in DEM. The challenge ciphertext is generated as follows:

$$(c^*, K^*) \leftarrow \mathcal{S}(1^\kappa) \times U(\mathcal{K}); d^+ \leftarrow \mathsf{E}(K^*, \mu^*);\ \text{return } ct^* = (c^*, d^+).$$

The difference between Game_2 and Game_3 is bounded by the SPR-CCA security of $\mathsf{KEM}^{\not\perp}$: There is an adversary $\mathcal{A}_{23}$ whose running time is approximately the same as that of $\mathcal{A}$ satisfying

$$|\Pr[S_2] - \Pr[S_3]| \leq \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM}^{\not\perp}, \mathcal{S}, \mathcal{A}_{23}}(\kappa).$$

We omit the detail of $\mathcal{A}_{23}$ since it is straightforward.

Game_4: In this game, the challenger uses random d^*. The challenge ciphertext is generated as follows:

$$(c^*, K^*) \leftarrow \mathcal{S}(1^\kappa) \times \mathcal{K}; d^* \leftarrow U(C_{|\mu^*|});\ \text{return } ct^* = (c^*, d^*).$$

The difference between Game_3 and Game_4 is bounded by the SPR-OTCCA security of DEM: There is an adversary $\mathcal{A}_{34}$ whose running time is approximately the same as that of $\mathcal{A}$ satisfying

$$|\Pr[S_3] - \Pr[S_4]| \leq \mathsf{Adv}^{\text{spr-otcca}}_{\mathsf{DEM}, \mathcal{A}_{34}}(\kappa).$$

We omit the detail of $\mathcal{A}_{34}$ since it is straightforward.

Game_5: We replace the decryption oracle. If given $ct = (c^*, d)$, the decryption oracle always return $\perp$.
Let Forge be an event that the adversary queries $d \neq d^*$ decrypted into some $\mu \neq \perp$ by using K^*. Game_4 and Game_5 are equal to each other until the event Forge occurs in Game_4. Hence, the difference between Game_4 and Game_5 is bounded by the INT-CTXT security of DEM: There is an adversary $\mathcal{A}_{45}$ whose running time is approximately the same as that of $\mathcal{A}$ satisfying

$$|\Pr[S_4] - \Pr[S_5]| \leq \Pr[\mathsf{Forge}] \leq \mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{DEM}, \mathcal{A}_{45}}(\kappa).$$

We omit the detail of $\mathcal{A}_{45}$ since it is straightforward. (We note that $\mathcal{A}_{45}$ makes no queries to Enc2.)

Game_6: We replace the decryption oracle in Game_5 with the original one.
Let Bad be the event that a randomly chosen $c^* \leftarrow \mathcal{S}(1^\kappa)$ is decapsulated into a key $K \neq \perp$. Game_5 and Game_6 are equivalent unless the event Bad occurs. Since $\mathsf{KEM}^{\perp}$ is ϵ-sparse, we have

$$|\Pr[S_5] - \Pr[S_6]| \leq \Pr[\mathsf{Bad}] \leq \epsilon.$$

Game_7: We change the timing of the generation of (c^*, K^*) as the original. This change is just conceptual, and we have

$$\Pr[S_6] = \Pr[S_7].$$

Notice that this is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa)$ with $b = 1$, thus, we have

$$\Pr[S_7] = \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}, \mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing the (in)equalities, we obtain the bound in the statement as follows:

$$\begin{aligned}\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{PKE}_{\mathsf{hy}}, \mathcal{S}_{\mathsf{hy}}\mathcal{A}}(\kappa) = |\Pr[S_0] - \Pr[S_7]| &\leq \sum_{i=0}^{6} |\Pr[S_i] - \Pr[S_{i+1}]| \\ &\leq \mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM}^{\perp}, \mathcal{S}, \mathcal{A}_{23}}(\kappa) + \mathsf{Adv}^{\text{spr-otcca}}_{\mathsf{DEM}, \mathcal{A}_{34}}(\kappa) + \mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{DEM}, \mathcal{A}_{45}}(\kappa) + \delta + \epsilon.\end{aligned}$$

□

Table 6. Summary of Games for the Proof of Theorem 4.1. We define $g(\mu) := \mathsf{Enc}(ek,\mu) = \mathsf{Enc}_0(ek,\mu;\mathsf{G}(\mu))$.

Game	H	G	c^*	K^*	Decapsulation: valid c	Decapsulation: invalid c	Justification
Game_0	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s,c)$	
Game_1	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	Lemma 2.2
$\mathsf{Game}_{1.1}$	H	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
$\mathsf{Game}_{1.2}$	$\mathsf{H}'_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	if a key pair is good
Game_2	$\mathsf{H}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	if a key pair is good
Game_3	$\mathsf{H}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	conceptual
$\mathsf{Game}_{3.1}$	$\mathsf{H}_q \circ g$	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
Game_4	$\mathsf{H}_q \circ g$	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	ciphertext indistinguishability
Game_5	$\mathsf{H}_q \circ g$	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	statistical disjointness
$\mathsf{Game}_{5.1}$	$\mathsf{H}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
Game_6	$\mathsf{H}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	conceptual
$\mathsf{Game}_{6.1}$	$\mathsf{H}'_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	if a key pair is good
$\mathsf{Game}_{6.2}$	H	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	if a key pair is good
Game_7	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
Game_8	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s,c)$	Lemma 2.2

B.2 Proof of Theorem 4.1:

We use the game-hopping proof. We consider Game_i for $i = 0,\dots,8$. We summarize the games in Table 6. Let S_i denote the event that the adversary outputs $b' = 1$ in game Game_i. We extend the security proof for SXY in [LW21], which extends the security proof for SXY [SXY18, XY19] to the case that the underlying PKE is derandomized by $\mathsf{KC} \circ \mathsf{T}$.

Game_0: This game is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 0$. Thus, we have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: This game is the same as Game_0 except that $\mathsf{H}_{\mathsf{prf}}(s,c)$ in the decapsulation oracle is replaced by $\mathsf{H}_q(c)$ where $\mathsf{H}_q : \mathcal{C} \to \mathcal{K}$ is another random oracle. We remark that $\mathcal{A}$ cannot access H_q directly.
As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_0] - \Pr[S_1]| \le 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}}) \cdot 2^{-\ell/2},$$

where $q_{\mathsf{H}_{\mathsf{prf}}}$ and q_{Dec} denote the number of queries to $\mathsf{H}_{\mathsf{prf}}$ and Dec the adversary makes, respectively.

Definition of $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$: We consider a set of *good* random oracles G, $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$. The following definition is taken from [HHK17, JZC+18, HKSU20, LW21]: For $(ek,dk) \in \mathsf{Gen}_0()$ and $\mu \in \mathcal{M}$, we define a set of good randomness as $\mathcal{R}^{\mathrm{good}}_{ek,dk,\mu} := \{r \in \mathcal{R} : \mathsf{Dec}(dk, \mathsf{Enc}_0(ek,\mu;r)) = \mu\}$, which could be empty. Let $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$ be a set of functions $\mathsf{G}: \mathcal{M} \to \mathcal{R}$ satisfying $\mathsf{G}(\mu) \in \mathcal{R}^{\mathrm{good}}_{ek,dk,\mu}$ for all $\mu \in \mathcal{M}$. Define $\delta_{ek,dk,\mu} := |\mathcal{R} \setminus \mathcal{R}^{\mathrm{good}}_{ek,dk,\mu}|/|\mathcal{R}|$, which is the fraction of the bad randomness for μ. Define $\delta_{ek,dk} := \max_{\mu\in\mathcal{M}} \delta_{ek,dk,\mu}$. We note that $\delta = \mathrm{Exp}_{(ek,dk)\leftarrow\mathsf{Gen}_0(1^\kappa)}[\delta_{ek,dk}]$.

$\mathsf{Game}_{1.1}$: This game is the same as Game_1 except that the random oracle G is chosen from $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$ instead of $\mathcal{F}(\mathcal{M},\mathcal{R})$.
If we fix (ek,dk), then we have $|\Pr[S_1 \mid (ek,dk)] - \Pr[S_{1.1} \mid (ek,dk)]| \le 8(q_{\mathsf{G}}+q_{\mathrm{Dec}}+2)^2 \cdot \delta_{ek,dk}$. (See [HKSU20, Theorem 3.2] and [LW21, Claim 1] for the analysis using Lemma 2.1. We note that the generation of the challenge ciphertext also queries to G and thus, the number of queries is $q_{\mathsf{G}} + q_{\mathrm{Dec}} + 1$.) Taking the average over $(ek,dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$, we obtain

$$|\Pr[S_1] - \Pr[S_{1.1}]| \le 8(q_{\mathsf{G}} + q_{\mathrm{Dec}} + 2)^2 \cdot \mathrm{Exp}_{(ek,dk)\leftarrow\mathsf{Gen}_0(1^\kappa)}[\delta_{ek,dk}] = 8(q_{\mathsf{G}} + q_{\mathrm{Dec}} + 2)^2\delta,$$

where q_{G} denotes the number of queries to G the adversary makes.

Definition of Bad *and* Good*:* We next define a bad event for key pairs. This definition is taken from [LW21]. Let us define an event Bad that there exists $\mu \in \mathcal{M}$ such that any $r \in \mathcal{R}$ is bad randomness, that is,

$$\mathsf{Bad} := \mathsf{boole}\left(\exists \mu \in \mathcal{M} : \mathcal{R}^{\mathrm{good}}_{ek,dk,\mu} = \emptyset\right),$$

where randomness is taken over $(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$. We define Good $= \neg$Bad. We have $\Pr[\neg\mathsf{Good}] = \Pr[\mathsf{Bad}] \le \delta$ ([LW21, Claim 3]). [6]
According to Lemma A.1, for any $p \ge 0$, we also have

$$|\Pr[S_{1.1}] - p| \le |\Pr[S_{1.1} \wedge \mathsf{Good}] - p| + \delta.$$

$\mathsf{Game}_{1.2}$*:* This game is the same as $\mathsf{Game}_{1.1}$ except that the random oracle $\mathsf{H}(\cdot)$ is simulated by $\mathsf{H}'_q(\mathsf{Enc}(ek, \cdot))$ where $\mathsf{H}'_q : \mathcal{C} \to \mathcal{K}$ is yet another random oracle. We remark that the decapsulation oracle and the generation of K^* also use $\mathsf{H}'_q(\mathsf{Enc}(ek, \cdot))$ as $\mathsf{H}(\cdot)$.
If Good occurs, then $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$ is perfectly correct from the definition of G and $g(\mu) := \mathsf{Enc}(ek, \mu; \mathsf{G}(\mu))$ is *injective*. Thus, if Good occurs, then $\mathsf{H}'_q \circ g : \mathcal{M} \to \mathcal{K}$ is a random function and the two games $\mathsf{Game}_{1.1}$ and $\mathsf{Game}_{1.2}$ are equivalent. We have

$$\Pr[S_{1.1} \wedge \mathsf{Good}] = \Pr[S_{1.2} \wedge \mathsf{Good}].$$

Game_2*:* This game is the same as $\mathsf{Game}_{1.2}$ except that the random oracle H is simulated by $\mathsf{H}_q \circ g$ instead of $\mathsf{H}'_q \circ g$.
As in the discussion of Theorem 4.2 on the difference between $\mathsf{Game}_{1.5}$ and Game_2, using the fact that, if Good occurs, $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$ is perfectly correct, we can show that the two games $\mathsf{Game}_{1.2}$ and Game_2 are equivalent and we have

$$\Pr[S_{1.2} \wedge \mathsf{Good}] = \Pr[S_2 \wedge \mathsf{Good}].$$

Game_3*:* This game is the same as Game_2 except that K^* is set as $\mathsf{H}_q(c^*)$ and the decapsulation oracle always returns $\mathsf{H}_q(c)$ as long as $c \neq c^*$. This modified decapsulation oracle is denoted by $\textsc{Dec}'$.
If Good occurs, then $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$ is perfectly correct from the definition of G. Thus, the two games Game_2 and Game_3 are equivalent and we have

$$\Pr[S_2 \wedge \mathsf{Good}] = \Pr[S_3 \wedge \mathsf{Good}].$$

In addition, according to Lemma A.1, for any $p \ge 0$, we have

$$|\Pr[S_3 \wedge \mathsf{Good}] - p| \le |\Pr[S_3] - p| + \delta.$$

$\mathsf{Game}_{3.1}$*:* This game is the same as Game_3 except that G is chosen from $\mathcal{F}(\mathcal{M}, \mathcal{R})$ instead of $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$.
As the difference between Game_1 and $\mathsf{Game}_{1.1}$, we have

$$|\Pr[S_3] - \Pr[S_{3.1}]| \le 8(q_\mathsf{G} + q_\mathsf{H} + 2)^2 \cdot \mathrm{Exp}_{(ek,dk)\leftarrow\mathsf{Gen}_0(1^\kappa)}[\delta_{ek,dk}] = 8(q_\mathsf{G} + q_\mathsf{H} + 2)^2\delta,$$

where q_H is the number of queries to H the adversary makes. We note that H queries to G internally.

Game_4*:* This game is the same as $\mathsf{Game}_{3.1}$ except that c^* is generated by $\mathcal{S}(1^\kappa)$.
The difference between two games $\mathsf{Game}_{3.1}$ and Game_4 is bounded by the advantage of ciphertext indistinguishability in disjoint simulatability as in [XY19, Lemma 4.7]. We have

$$|\Pr[S_{3.1}] - \Pr[S_4]| \le \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_\mathcal{M},\mathcal{S},\mathcal{A}_{34}}(\kappa).$$

The reduction algorithm is obtained straightforwardly.

Game_5*:* This game is the same as Game_4 except that $K^* \leftarrow \mathcal{K}$ instead of $K^* \leftarrow \mathsf{H}_q(c^*)$.
In Game_4, if $c^* \leftarrow \mathcal{S}(1^\kappa)$ is not in $\mathsf{Enc}(ek, \mathcal{M})$, then the adversary has no information about $K^* = \mathsf{H}_q(c^*)$ and thus, K^* looks uniformly at random. Hence, the difference between two games Game_4 and Game_5 is bounded by the statistical disjointness in disjoint simulatability as in [XY19, Lemma 4.8]. We have

$$|\Pr[S_4] - \Pr[S_5]| \le \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

[6] $\Pr[\mathsf{Bad}] = \Pr_{(ek,dk)\leftarrow\mathsf{Gen}_0(1^\kappa)}[\exists \mu \in \mathcal{M} \text{ s.t. } \mathcal{R}^{\mathrm{good}}_{ek,dk,\mu} = \emptyset] = \Pr_{(ek,dk)\leftarrow\mathsf{Gen}_0(1^\kappa)}[\exists \mu \in \mathcal{M} \text{ s.t. } \delta_{ek,dk,\mu} = 1] = \Pr_{(ek,dk)\leftarrow\mathsf{Gen}_0(1^\kappa)}[\delta_{ek,dk} = 1] \le \mathrm{Expt}_{(ek,dk)\leftarrow\mathsf{Gen}_0(1^\kappa)}[\delta_{ek,dk}] = \delta$

$\mathsf{Game}_{5.1}$: This game is the same as Game_5 except that G is chosen from $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$ instead of $\mathcal{F}(\mathcal{M},\mathcal{R})$. As the difference between Game_3 and $\mathsf{Game}_{3.1}$, we have

$$|\Pr[S_5] - \Pr[S_{5.1}]| \le 8(q_G + q_H + 1)^2 \cdot \mathrm{Exp}_{(ek,dk)\leftarrow \mathsf{Gen}_0(1^\kappa)}[\delta_{ek,dk}] \le 8(q_G + q_H + 2)^2\delta.$$

We note that H queries to G internally.
In addition, according to Lemma A.1, for any $p \ge 0$, we have

$$|\Pr[S_{5.1}] - p| \le |\Pr[S_{5.1} \wedge \mathrm{Good}] - p| + \delta.$$

Game_6: This game is the same as Game_5 except that the decapsulation oracle is reset as Dec. Similar to the case for Game_2 and Game_3, if Good occurs, then the two games Game_5 and Game_6 are equivalent. We have

$$\Pr[S_{5.1} \wedge \mathrm{Good}] = \Pr[S_6 \wedge \mathrm{Good}].$$

$\mathsf{Game}_{6.1}$: This game is the same as Game_6 except that the random oracle H is simulated by $\mathsf{H}'_q \circ g$ where $\mathsf{H}'_q : \mathcal{C} \to \mathcal{K}$ is yet another random oracle as in $\mathsf{Game}_{1.2}$. Similar to the case for $\mathsf{Game}_{1.2}$ and Game_2, if Good occurs, then the two games Game_6 and $\mathsf{Game}_{6.1}$ are equivalent. We have

$$\Pr[S_6 \wedge \mathrm{Good}] = \Pr[S_{6.1} \wedge \mathrm{Good}].$$

$\mathsf{Game}_{6.2}$: This game is the same as $\mathsf{Game}_{6.1}$ except that the random oracle $\mathsf{H}(\cdot)$ is set as the original. Similar to the case for $\mathsf{Game}_{1.1}$ and $\mathsf{Game}_{1.2}$, if Good occurs, then the two games $\mathsf{Game}_{6.1}$ and $\mathsf{Game}_{6.2}$ are equivalent. We have

$$\Pr[S_{6.1} \wedge \mathrm{Good}] = \Pr[S_{6.2} \wedge \mathrm{Good}].$$

In addition, according to Lemma A.1, we have, for any $p \ge$,

$$|\Pr[S_{6.2} \wedge \mathrm{Good}] - p| \le |\Pr[S_{6.2}] - p| + \delta.$$

Game_7: This game is the same as $\mathsf{Game}_{6.2}$ except that the random oracle G is chosen from $\mathcal{F}(\mathcal{M},\mathcal{R})$ instead of $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$. Similar to the case for Game_1 and $\mathsf{Game}_{1.1}$, we have

$$|\Pr[S_{6.2}] - \Pr[S_7]| \le 8(q_G + q_{\mathrm{Dec}} + 1)^2\delta. \le 8(q_G + q_{\mathrm{Dec}} + 2)^2\delta.$$

Game_8: This game is the same as Game_7 except that $\mathsf{H}_q(c)$ in the decapsulation is replaced by $\mathsf{H}_{\mathsf{prf}}(s,c)$. As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_7] - \Pr[S_8]| \le 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}}) \cdot 2^{-\ell/2}.$$

We note that this game is the original game $\mathrm{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 1$. Thus, we have

$$\Pr[S_8] = \Pr[\mathrm{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing those (in)equalities, we obtain the following bound:

$$\begin{aligned}\mathrm{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_8]| \\ &\le \mathrm{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + \mathrm{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) \\ &\quad + 4\delta + 16(q_G + q_{\mathrm{Dec}} + 2)^2\delta + 16(q_G + q_H + 2)^2\delta + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}}) \cdot 2^{-\ell/2}.\end{aligned}$$

C Variants of the Fujisaki-Okamoto Transform

In this section we review the variants of the FO transforms. The Fujisaki-Okamoto (FO) transform FO converts weakly-secure probabilistic PKE scheme PKE_0 into IND-CCA-secure KEM scheme. Hofheinz et al. [HHK17] decomposed the FO transform FO into two transforms T and U. In this section we review the variants of the FO transforms, we define variants of U and then define the variants of FO by combining with T.

C.1 Transform T

In the original T in [HHK17, Section 3.1], the decryption algorithm checks the validity of c by re-encryption check. We omit this re-encryption check.
Let $\mathsf{PKE}_0 = (\mathsf{Gen}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ be a probabilistic PKE scheme, whose ciphertext space is $\mathcal{C}_{\mathsf{PKE}}$, message space is $\mathcal{M}$, and randomness space is $\mathcal{R}_{\mathsf{Enc}_0}$. Let $\mathsf{G} : \mathcal{M} \to \mathcal{R}_{\mathsf{Enc}_0}$ be a hash function modeled by the random oracle. $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}) = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$ is defined as in Figure 8.

$\mathsf{Gen}(1^\kappa)$	$\mathsf{Enc}(ek, \mu)$	$\mathsf{Dec}(dk, c)$
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$c := \mathsf{Enc}_0(ek, \mu; \mathsf{G}(\mu))$	$\mu' \leftarrow \mathsf{Dec}_0(dk, c)$
return (ek, dk)	**return** c	**return** μ'

Fig. 8. $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$

C.2 Variants of U

Hofheinz et al. defined U's variants, $\mathsf{U}^{\not\perp}$, $\mathsf{U}^{\perp}$, $\mathsf{U}^{\not\perp}_m$, and $\mathsf{U}^{\perp}_m$ [HHK17], where the superscript "$^{\not\perp}$" and "$^{\perp}$" implies *implicit rejection* and *explicit rejection*, respectively, and the subscript "$_m$" implies the computation of key K involves a plaintext μ only, while if there is no subscript, then it involves μ and ciphertext c.
Saito et al. defined SXY, which is essentially the same as $\mathsf{U}^{\not\perp}_m$ [SXY18]. Bindel et al. discussed the relations of the IND-CCA security of KEM schemes obtained by those transforms via indifferentiable reductions [BHH+19]. In their discussion, they modify $\mathsf{U}^{\not\perp}$, which we write $\mathsf{U}^{\not\perp,\mathsf{prf}}$ here. In their $\mathsf{U}^{\not\perp}$, they use $K := \mathsf{H}_{\mathsf{prf}}(s, c)$ for invalid ciphertext c instead of $K := \mathsf{H}(s, c)$ as in [HHK17].
Let us review the definitions of the transforms. Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme, whose ciphertext space is $\mathcal{C}$ and message space is $\mathcal{M}$. Let $\mathsf{H} \colon \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ be a hash function modeled by the random oracle. Let $\mathsf{H}_{\mathsf{prf}} \colon \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ be another hash function modeled by the random oracle.

- $\mathsf{U}^{\not\perp}[\mathsf{PKE}, \mathsf{H}]$: $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{U}^{\not\perp}[\mathsf{PKE}, \mathsf{H}]$ is defined as in Figure 9.
- $\mathsf{U}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$: This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 3 of $\overline{\mathsf{Dec}}$ is replaced by "then return $K := \mathsf{H}_{\mathsf{prf}}(s, c)$."
- $\mathsf{U}^{\perp}[\mathsf{PKE}, \mathsf{H}]$: This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 3 of $\overline{\mathsf{Dec}}$ is replaced by "then return $K := \perp$." This variant does not require s in $\overline{dk}$.
- $\mathsf{U}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$: Let $\mathsf{H} \colon \mathcal{M} \to \mathcal{K}$ be a hash function modeled by the random oracle. This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 3 of $\overline{\mathsf{Enc}}$ is replace by "$K := \mathsf{H}(\mu)$" and line 4 of $\overline{\mathsf{Dec}}$ is replaced by "else return $K := \mathsf{H}(\mu)$."
- $\mathsf{U}^{\perp}_m[\mathsf{PKE}, \mathsf{H}]$: Let $\mathsf{H} \colon \mathcal{M} \to \mathcal{K}$ be a hash function modeled by the random oracle. This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 3 of $\overline{\mathsf{Enc}}$ is replace by "$K := \mathsf{H}(\mu)$," line 3 of $\overline{\mathsf{Dec}}$ is replaced by "then return $K := \perp$," and line 4 of $\overline{\mathsf{Dec}}$ is replaced by "else return $K := \mathsf{H}(\mu)$." This variants does not require s in $\overline{dk}$.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, c)$, where $\overline{dk} = (dk, ek, s)$
1 : $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	1 : $\mu \leftarrow \mathcal{D}_{\mathcal{M}}$	1 : $\mu' \leftarrow \mathsf{Dec}(dk, c)$
2 : $s \leftarrow \mathcal{M}$	2 : $c := \mathsf{Enc}(ek, \mu)$	2 : **if** $\mu' = \perp$ or $c \neq \mathsf{Enc}(ek, \mu')$
3 : $\overline{dk} := (dk, ek, s)$	3 : $K := \mathsf{H}(\mu, c)$	3 : **then return** $K := \mathsf{H}(s, c)$
4 : **return** $(ek, \overline{dk})$	4 : **return** (c, K)	4 : **else return** $K := \mathsf{H}(\mu', c)$

Fig. 9. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{U}^{\not\perp}[\mathsf{PKE}, \mathsf{H}]$

We adapt the discussions of Bindel et al. to SPR-CCA-security of KEM schemes obtained by the variants of U. See the left hand side of Figure 10.

C.3 Variants of HU

Targhi and Unruh [TU16] introduced a variant of FO transform for PKE, whose ciphertext has an additional hash value of a random message μ. Hofheinz et al. called this variant QFO and they decomposed it into T and QU, [HHK17]. Hofheinz et al. defined QU's variants, $\mathsf{QU}^{\not\perp}_m$ and $\mathsf{QU}^{\perp}_m$. In those variants a ciphertext includes an additional hash $d := \mathsf{F}(\mu)$, where $\mathsf{F} \colon \mathcal{M} \to \mathcal{M}$. (They require $\mathcal{M}$ to be a subset of a finite field.) Jiang et al. [JZM19] defined $\mathsf{HU}^{\perp}_m$ as a variant of $\mathsf{QU}^{\perp}_m$, where $\mathsf{F} \colon \mathcal{M} \to \mathcal{H}$ with arbitrary $\mathcal{M}$ and $\mathcal{H}$. This allows us to make a ciphertext shorter. We define its variants $\mathsf{HU}^{\perp}_m$, $\mathsf{HU}^{\not\perp}_m$, $\mathsf{HU}^{\perp}$, $\mathsf{HU}^{\not\perp}_m$, and $\mathsf{HU}^{\not\perp,\mathsf{prf}}$ as the variants of U. In the definition, we allow F to take ek optional.
Let us review the definitions of the variants: Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme, whose ciphertext space is $\mathcal{C}$ and message space is $\mathcal{M}$. Let $\mathsf{H} \colon \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ be a hash function modeled by

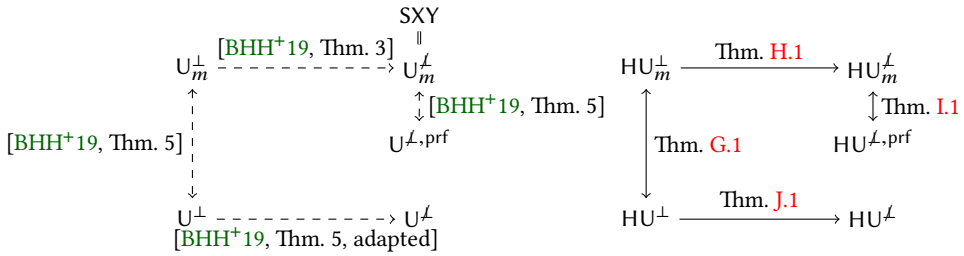

Fig. 10. The relation between IND-CCA and SPR-CCA security of KEMs using the variants of U and HU. Dashed arrow implies the implications in [BHH+19].

the random oracle. Let $\mathsf{H}_{\mathsf{prf}} \colon \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ be another hash function modeled by the random oracle. Let $\mathsf{F} \colon \mathcal{M} \to \mathcal{H}$ be yet another hash function modeled by the random oracle.

- $\mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$: $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ is defined as in Figure 11.
- $\mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$: This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 3 of $\overline{\mathsf{Dec}}$ is replaced by "then return $K := \mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$."
- $\mathsf{HU}^{\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$: This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 3 of $\overline{\mathsf{Dec}}$ is replaced by "then return $K := \perp$." This variants does not require s in $\overline{dk}$.
- $\mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$: Let $\mathsf{H} \colon \mathcal{M} \to \mathcal{K}$ be a hash function modeled by the random oracle. This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 4 of $\overline{\mathsf{Enc}}$ is replaced by "$K := \mathsf{H}(\mu)$," line 3 of $\overline{\mathsf{Dec}}$ is replaced by "then return $K := \mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$," and line 4 of $\overline{\mathsf{Dec}}$ is replaced by "else return $K := \mathsf{H}(\mu)$."
- $\mathsf{HU}^{\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$: Let $\mathsf{H} \colon \mathcal{M} \to \mathcal{K}$ be a hash function modeled by the random oracle. This transform is the same as $\mathsf{U}^{\not\perp}$ except that line 4 of $\overline{\mathsf{Enc}}$ is replaced by "$K := \mathsf{H}(\mu)$," line 3 of $\overline{\mathsf{Dec}}$ is replaced by "then return $K := \perp$," and line 4 of $\overline{\mathsf{Dec}}$ is replaced by "else return $K := \mathsf{H}(\mu)$." This variants does not require s in $\overline{dk}$.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek, s)$
1 : $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	1 : $\mu \leftarrow \mathcal{D}_{\mathcal{M}}$	1 : $\mu' \leftarrow \mathsf{Dec}(dk, c_0)$
2 : $s \leftarrow \mathcal{M}$	2 : $c_0 := \mathsf{Enc}(ek, \mu)$	2 : **if** $\mu' = \perp$ or $c_0 \neq \mathsf{Enc}(ek, \mu')$ or $c_1 \neq \mathsf{F}(\mu'[, ek])$
3 : $\overline{dk} := (dk, ek, s)$	3 : $c_1 := \mathsf{F}(\mu[, ek])$	3 : **then return** $K := \mathsf{H}(s, c_0, c_1)$
4 : **return** $(ek, \overline{dk})$	4 : $K := \mathsf{H}(\mu, c_0, c_1)$	4 : **else return** $K := \mathsf{H}(\mu', c_0, c_1)$
	5 : **return** $((c_0, c_1), K)$	

Fig. 11. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$

We will adapt the discussions of Bindel et al.to SPR-CCA-security of KEM schemes obtained by the variants of U. See the right hand side of Figure 10.

C.4 Variants of FO

Combining T and the variants of U or HU, we obtain several variants of FO as follows: Let $\mathsf{PKE}_0 = (\mathsf{Gen}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ be a probabilistic PKE scheme: If we combine T and U^y_x, then we obtain FO^y_x. If we combine T and HU^y_x, then we obtain HFO^y_x.

D Property of T

In this section, we show that T preserves ciphertext indistinguishability of disjoint simulatability.

Theorem D.1. *Suppose that a probabilistic PKE scheme* PKE_0 *is ciphertext indistinguishable and* OW-CPA*-secure. Then,* $\mathsf{PKE} := \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$ *is also ciphertext indistinguishable in the QROM.*

$\mathsf{F}(\mu^*, r^*)$	$\mathcal{A}_{01}^{\mathsf{G}}(ek, c^*)$:
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$\mathsf{inp} := (ek, c^*)$
$c^* := \mathsf{Enc}_0(ek, \mu^*; r^*)$	$i \leftarrow [q_{\mathsf{H}}]$
$\mathsf{inp} := (ek, c^*)$	Run $\mathcal{A}^{\mathsf{G}}(\mathsf{inp})$ until i-th query $\|\hat{x}\rangle$ to G
return inp	**if** $i >$ number of queries to G, **return** $\perp$
	else return $x' := \mathsf{Measure}(\|\hat{x}\rangle)$

Fig. 12. Algorithm F and adversary $\mathcal{A}_{01}$

Precisely speaking, for any quantum adversary $\mathcal{A}$ against PKE *issuing at most q_{G} quantum queries to* G*, there exist quantum adversaries $\mathcal{A}_{01}$ against* OW-CPA *security of* PKE_0 *and $\mathcal{A}_{12}$ against ciphertext indistinguishability of* PKE_0 *such that*

$$\mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}}(\kappa) \le 2q_{\mathsf{G}}\sqrt{\mathsf{Adv}^{\text{ow-cpa}}_{\mathsf{PKE}_0,\mathcal{D}_{\mathcal{M}},\mathcal{A}_{01}}(\kappa)} + \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}_0,\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{12}}(\kappa).$$

Proof: Let us consider the following three games, Game_0, Game_1, and Game_2. Let S_i denote the event that the adversary outputs $b' = 1$ in Game_i.

Game_0: This game is defined as follows:

$$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa); \mu^* \leftarrow \mathcal{D}_{\mathcal{M}}; r^* \leftarrow \mathsf{G}(\mu^*); c^* := \mathsf{Enc}_0(ek, \mu^*; r^*); b' \leftarrow \mathcal{A}^{\mathsf{G}(\cdot)}(ek, c^*); \textbf{return } b'.$$

Game_1: This game is the same as Game_0 except that a randomness to generate a challenge ciphertext is freshly generated:

$$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa); \mu^* \leftarrow \mathcal{D}_{\mathcal{M}}; r^* \leftarrow \mathcal{R}; c^* := \mathsf{Enc}_0(ek, \mu^*; r^*); b' \leftarrow \mathcal{A}^{\mathsf{G}(\cdot)}(ek, c^*); \textbf{return } b'.$$

Game_2: This game is the same as Game_1 except that a challenge ciphertext is generated by the simulator $\mathcal{S}(1^\kappa, ek)$:

$$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa); c^* \leftarrow \mathcal{S}(1^\kappa, ek); b' \leftarrow \mathcal{A}^{\mathsf{G}(\cdot)}(ek, c^*); \textbf{return } b'.$$

This completes the descriptions of games. It is easy to see that we have

$$\mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}}(\kappa) = |\Pr[S_0] - \Pr[S_2]|.$$

We give an upperbound for this advantage by the following lemmas.

Lemma D.1. *There exists a quantum adversary $\mathcal{A}_{01}$ such that*

$$|\Pr[S_0] - \Pr[S_1]| \le 2q_{\mathsf{G}}\sqrt{\mathsf{Adv}^{\text{ow-cpa}}_{\mathsf{PKE}_0,\mathcal{D}_{\mathcal{M}},\mathcal{A}_{01}}(\kappa)}.$$

Proof (Proof of Lemma D.1). Let F be an algorithm described in Figure 12. It is easy to see that Game_0 can be restated as

$$\mu^* \leftarrow \mathcal{D}_{\mathcal{M}}; r^* \leftarrow \mathsf{G}(\mu^*); \mathsf{inp} := \mathsf{F}(\mu^*, r^*); b' \leftarrow \mathcal{A}^{\mathsf{G}(\cdot)}(\mathsf{inp}); \textbf{return } b'.$$

and Game_1 can be restated as

$$\mu^* \leftarrow \mathcal{D}_{\mathcal{M}}; r^* \leftarrow \mathcal{R}; \mathsf{inp} := \mathsf{F}(\mu^*, r^*); b' \leftarrow \mathcal{A}^{\mathsf{G}(\cdot)}(\mathsf{inp}); \textbf{return } b'.$$

Applying the O2H lemma (Corollary A.1) with $\mathcal{X} = \mathcal{M}$, $\mathcal{Y} = \mathcal{R}$, $\mathcal{D}_{\mathcal{X}} = \mathcal{D}_{\mathcal{M}}$, $x = \mu^*$, $y = r^*$, and algorithms $\mathcal{A}$ and F, we have

$$|\Pr[S_0] - \Pr[S_1]| \le 2q_{\mathsf{G}}\sqrt{\Pr[\mu^* \leftarrow \mathcal{A}_{01}^{\mathsf{G}}(ek, c^*)]}.$$

where $\mathcal{A}_{01}^{\mathsf{G}}$ is an algorithm described in Figure 12, $(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$, $\mu^* \leftarrow \mathcal{D}_{\mathcal{M}}$, $r^* \leftarrow \mathcal{R}$, and $c^* := \mathsf{Enc}_0(ek, \mu^*, r^*)$.
We have $\Pr[\mu^* \leftarrow \mathcal{A}_{01}^{\mathsf{G}}(ek, c^*)] \le \mathsf{Adv}^{\text{ow-cpa}}_{\mathsf{PKE}_0,\mathcal{D}_{\mathcal{M}},\mathcal{A}_{01}}(\kappa)$. By combining these inequalities, the lemma is proven. □

Lemma D.2. *There exists an adversary $\mathcal{A}_{12}$ such that*

$$|\Pr[S_1] - \Pr[S_2]| \le \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}_0,\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{12}}(\kappa).$$

Since the proof is obtained straightforwardly, we omit it.
Combining the above two lemmas, we obtain the wanted result. □

Open problem: One might wonder whether we could make the above lemma tighter by using the semi-classical O2H lemma [AHU19], the double-sided O2H lemma [BHH+19], or the MRM O2H lemma [KSS+20]. Essentially speaking, in some game transition, we need to replace $c^* = \mathsf{Enc}(ek, \mu^*; \mathsf{G}(\mu^*))$ with $c^* = \mathsf{Enc}(ek, \mu^*; r^*)$ with fresh randomness $r^* \leftarrow \mathcal{R}$. This change is an obstacle for tight security.
The existing tight security proof for transform T in [BHH+19] strongly depends on the fact that the goal is onewayness and the adversary finally outputs μ in the game. The condition allows us to use $\left|\sqrt{P_{\mathrm{left}}} - \sqrt{P_{\mathrm{right}}}\right| \le 2d\sqrt{P_{\mathrm{right}}}$ with $P_{\mathrm{guess}} = 0$ in Lemma A.2, which yields $P_{\mathrm{left}} \le 4d^2 P_{\mathrm{guess}}$.
If we invoke the MRM O2H lemma [KSS+20], we will consider an algorithm Ext such that

$$\Pr\left[\begin{array}{c} \mathsf{G} = \mathsf{G}' \leftarrow \mathcal{F}(\mathcal{M}, \mathcal{R}); (ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa); \mu^* \leftarrow \mathcal{D}_{\mathcal{M}}; S := \{\mu^*\}; \\ r^* \leftarrow \mathcal{R}; \mathsf{G}'(\mu^*) := r^*; c^* \leftarrow \mathsf{Enc}_0(ek, \mu^*; r^*); T \leftarrow \mathsf{Ext}^{\mathsf{G},\mathsf{G}'}(ek, c^*) : T \cap S \neq \emptyset \end{array}\right].$$

Notice that, on input μ^*, G' is overwritten by r^*. We want to connect this probability with the advantage of the OW-CPA/IND-CPA/DS security of PKE_0, but this seems impossible. The reduction algorithm on input ek and $c^* = \mathsf{Enc}_0(ek, \mu^*; r^*)$ is given an access to a random oracle G. In order to implement G' on input μ^*, it should know μ^* and r^*, which is already the solution of the challenge ciphertext of the security game in the OW-CPA/IND-CPA/DS security. Thus, we cannot use them in the context of T unfortunately.

E Properties of $\mathsf{U}^{\not\perp}$

As we seen in Figure 10, $\mathsf{U}^{\not\perp}$ and $\mathsf{SXY} = \mathsf{U}^{\not\perp}_m$ are not connected. Indeed, we face a subtle problem to apply the indifferentiable reduction in Bindel et al. [BHH+19]: Suppose that we have $\mathcal{A}$ against the SPR-CCA security of KEM obtained by $\mathsf{U}^{\not\perp}$. In their indifferentiable reduction, they construct $\mathcal{A}_m$ against the SPR-CCA security of KEM obtained by $\mathsf{U}^{\not\perp}_m$. $\mathcal{A}_m$ given $\mathsf{H}_m \colon \mathcal{M} \to \mathcal{K}$ simulates $\mathsf{H} \colon \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ by

$$\mathsf{H}(\mu, c) = \begin{cases} \mathsf{H}_m(\mu) & \text{if } c = \mathsf{Enc}(ek, \mu) \\ \mathsf{H}'(\mu, c) & \text{otherwise.} \end{cases}$$

Unfortunately, this simulation makes $\mathsf{H}(s, c)$ different from $\mathsf{H}_{\mathsf{prf}}(s, c)$ at the point (s, c) with $c = \mathsf{Enc}(ek, s)$. We here *directly* prove the security properties of $\mathsf{U}^{\not\perp}$. We give proof sketches, because the proofs are very similar to those of SXY in Section 4.

E.1 SPR-CCA Security

We can use the proof of the SPR-CCA security of $\mathsf{SXY} = \mathsf{U}^{\not\perp}_m$ (subsection B.2) with slight modifications. Roughly speaking, we replace $\mathsf{H}(s, c)$ with $\mathsf{H}_q(c)$ and, then, apply the above indifferentiable reduction. Doing so, we can find the situation is essentially equivalent to Game_1 (or Game_7) of Table 6.

Theorem E.1 (Case for derandomized PKE). *Let PKE_0 be a probabilistic PKE scheme. Let us consider a derandomized PKE scheme $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$. Suppose that a ciphertext space $\mathcal{C}$ of PKE depends on the public parameter only. If PKE is strongly disjoint-simulatable and δ-correct with negligible δ, then $\mathsf{KEM} = \mathsf{U}^{\not\perp}[\mathsf{PKE}, \mathsf{H}]$ is SPR-CCA-secure.*
Formally speaking, for any $\mathcal{A}$ against the SPR-CCA security of KEM issuing at most q_{Dec} queries to the decapsulation oracle and q_{G} and q_{H} queries to G and H respectively, there exist $\mathcal{A}_{34}$ against ciphertext-indistinguishability of PKE such that

$$\begin{aligned}\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) \le \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4\delta \\ + 16(q_{\mathsf{G}} + q_{\mathrm{Dec}} + 2)^2\delta + 16(q_{\mathsf{G}} + q_{\mathsf{H}} + 2)^2\delta + 4(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|}.\end{aligned}$$

Theorem E.2 (Case for non-derandomized PKE). *Suppose that a ciphertext space $\mathcal{C}$ of PKE depends on the public parameter only. If PKE is strongly disjoint-simulatable and δ-correct with negligible δ, then $\mathsf{KEM} = \mathsf{U}^{\not\perp}[\mathsf{PKE}, \mathsf{H}]$ is SPR-CCA-secure.*
Formally speaking, for any $\mathcal{A}$ against the SPR-CCA security of KEM issuing at most q_{Dec} queries to the decapsulation oracle and q_{G} and q_{H} queries to G and H, respectively, there exist $\mathcal{A}_{34}$ against ciphertext-indistinguishability of PKE such that

$$\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A},\mathcal{S}}(\kappa) \le \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|} + 4\delta.$$

Proof of Theorem E.1: We use the game-hopping proof. We consider Game_i for $i = 0, \ldots, 8$. We summarize the games in Table 7. Let S_i denote the event that the adversary outputs $b' = 1$ in game Game_i. Let Acc and $\overline{\mathsf{Acc}}$ denote the event that the key pair (ek, dk) is accurate and inaccurate, respectively.

Table 7. Summary of Games for the Proof of Theorem E.1. We define $g(\mu) = \mathsf{Enc}(ek, \mu) = \mathsf{Enc}_0(ek, \mu; \mathsf{G}(\mu))$.

Game	H	G	c^*	K^*	Decryption valid c	Decryption invalid c	justification
Game_0	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*,c^*)$	$\mathsf{H}(\mu,c)$	$\mathsf{H}(s,c)$	
Game_1	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*,c^*)$	$\mathsf{H}(\mu,c)$	$\mathsf{H}_q(c)$	Lemma 2.2
$\mathsf{Game}_{1.1}$	H	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*,c^*)$	$\mathsf{H}(\mu,c)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
$\mathsf{Game}_{1.2}$	$\mathsf{H}'_q \circ g$ / H′	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*,c^*)$	$\mathsf{H}(\mu,c)$	$\mathsf{H}_q(c)$	if a key pais is good
Game_2	$\mathsf{H}_q \circ g$ / H′	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}(\mu^*,c^*)$	$\mathsf{H}(\mu,c)$	$\mathsf{H}_q(c)$	if a key pais is good
Game_3	$\mathsf{H}_q \circ g$ / H′	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	conceptual
$\mathsf{Game}_{3.1}$	$\mathsf{H}_q \circ g$ / H′	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathsf{Enc}(ek,\mu^*)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
Game_4	$\mathsf{H}_q \circ g$ / H′	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$\mathsf{H}_q(c^*)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	ciphertext indistinguishability
Game_5	$\mathsf{H}_q \circ g$ / H′	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	statistical disjointness
$\mathsf{Game}_{5.1}$	$\mathsf{H}_q \circ g$ / H′	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
Game_6	$\mathsf{H}_q \circ g$ / H′	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c)$	conceptual change
$\mathsf{Game}_{6.1}$	$\mathsf{H}'_q \circ g$ / H′	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu,c)$	$\mathsf{H}_q(c)$	if a key pais is good
$\mathsf{Game}_{6.2}$	H	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu,c)$	$\mathsf{H}_q(c)$	if a key pais is good
Game_7	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu,c)$	$\mathsf{H}_q(c)$	Lemma 2.1 + correctness
Game_8	H	$\mathcal{F}(\mathcal{M},\mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{K})$	$\mathsf{H}(\mu,c)$	$\mathsf{H}(s,c)$	Lemma 2.2

Game_0: This game is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 0$. Thus, we have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: This game is the same as Game_0 except that $\mathsf{H}(s,c)$ in the decapsulation oracle is replace with $\mathsf{H}_q(c)$ where $\mathsf{H}_q : \mathcal{C} \to \mathcal{K}$ is another random oracle. We remark that $\mathcal{A}$ is not given direct access to H_q. As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_0] - \Pr[S_1]| \le 2(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|},$$

where q_{H} and q_{Dec} denote the number of queries to H and Dec the adversary makes, respectively.

$\mathsf{Game}_{1.1}$: This game is the same as Game_1 except that the random oracle $\mathsf{G}(\cdot)$ is chosen from $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$ instead of $\mathcal{F}(\mathcal{M},\mathcal{R})$. See subsection B.2 for the definitions of $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M},\mathcal{R})$, Bad, and Good. As in the argument in subsection B.2, we obtain

$$|\Pr[S_1] - \Pr[S_{1.1}]| \le 8(q_{\mathsf{G}} + q_{\mathrm{Dec}} + 2)^2\delta.$$

In addition, We have $\Pr[\mathsf{Bad}] \le \delta$ ([LW21, Claim 3]). According to Lemma A.1, for any $p \ge 0$, we also have

$$|\Pr[S_{1.1}] - p| \le |\Pr[S_{1.1} \wedge \mathsf{Good}] - p| + \delta.$$

$\mathsf{Game}_{1.2}$: This game is the same as $\mathsf{Game}_{1.1}$ except that the random oracle $\mathsf{H}(\cdot,\cdot)$ is simulated as follows: Let $\mathsf{H}'_q : \mathcal{C} \to \mathcal{K}$ and $\mathsf{H}' : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ be random oracles. Define

$$\mathsf{H}(\mu,c) = \begin{cases} \mathsf{H}'_q(\mathsf{Enc}(ek,\mu)) & \text{if } c = \mathsf{Enc}(ek,\mu), \\ \mathsf{H}'(\mu,c) & \text{otherwise.} \end{cases}$$

We remark that the decapsulation oracle and the generation of K^* also use this simulation.
If Good occurs, then $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$ is perfectly correct from the definition of G and $g(\mu) := \mathsf{Enc}(ek,\mu;\mathsf{G}(\mu))$ is *injective*. Thus, $\mathsf{H}'_q \circ g : \mathcal{M} \to \mathcal{K}$ is a random function and the two games $\mathsf{Game}_{1.1}$ and $\mathsf{Game}_{1.2}$ are equivalent if Bad does not occurs. We have

$$\Pr[S_{1.1} \wedge \mathsf{Good}] = \Pr[S_{1.2} \wedge \mathsf{Good}].$$

Game_2: This game is the same as $\mathsf{Game}_{1.2}$ except that the random oracle H is simulated by $\mathsf{H}_q \circ g$ and H' instead of $\mathsf{H}'_q \circ g$ and H'.
If Good occurs, then $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}, \mathsf{G}]$ is perfectly correct from the definition of G. Hence, the two games $\mathsf{Game}_{1.2}$ and Game_2 are equivalent, because a value of $\mathsf{H}'_q(c)$ for an invalid c is not used in $\mathsf{Game}_{1.2}$. We have

$$\Pr[S_{1.2} \wedge \mathrm{Good}] = \Pr[S_2 \wedge \mathrm{Good}].$$

Game_3: This game is the same as Game_2 except that K^* is set as $\mathsf{H}_q(c^*)$ and the decapsulation oracle always returns $\mathsf{H}_q(c)$ as long as $c \neq c^*$. This decapsulation oracle will be denoted by Dec'.
If Good occurs, then $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}, \mathsf{G}]$ is perfectly correct from the definition of G. If so, the two games Game_2 and Game_3 are equivalent, and we have

$$\Pr[S_2 \wedge \mathrm{Good}] = \Pr[S_3 \wedge \mathrm{Good}].$$

According to Lemma A.1, for any $p \geq 0$, we have

$$|\Pr[S_3 \wedge \mathrm{Good}] - p| \leq |\Pr[S_3] - p| + \delta.$$

$\mathsf{Game}_{3.1}$: This game is the same as Game_3 except that G is chosen from $\mathcal{F}(\mathcal{M}, \mathcal{R})$ instead of $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$.
As in the argument in subsection B.2, we obtain

$$|\Pr[S_3] - \Pr[S_{3.1}]| \leq 8(q_{\mathsf{G}} + q_{\mathsf{H}} + 2)^2 \delta.$$

(We note that H and the challenge ciphertext also query to G internally.)

Game_4: This game is the same as Game_3 except that c^* is generated by $\mathcal{S}(1^\kappa)$.
The difference between two games Game_3 and Game_4 is bounded by the advantage of ciphertext indistinguishability in disjoint simulatability. We have

$$|\Pr[S_3] - \Pr[S_4]| \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa).$$

Game_5: This game is the same as Game_4 except that $K^* \leftarrow \mathcal{K}$ instead of $K^* \leftarrow \mathsf{H}_q(c^*)$.
In Game_4, if $c^* \leftarrow \mathcal{S}(1^\kappa)$ is not in $\mathsf{Enc}(ek, \mathcal{M})$, then the adversary has no information about $K^* = \mathsf{H}_q(c^*)$ and thus, K^* looks uniformly at random. Hence, the difference between two games Game_4 and Game_5 is bounded by the statistical disjointness in disjoint simulatability as in [XY19, Lemma 4.8]. We have

$$|\Pr[S_4] - \Pr[S_5]| \leq \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

$\mathsf{Game}_{5.1}$: This game is the same as Game_5 except that G is chosen from $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$ instead of $\mathcal{F}(\mathcal{M}, \mathcal{R})$.
As in the argument in subsection B.2, we obtain

$$|\Pr[S_5] - \Pr[S_{5.1}]| \leq 8(q_{\mathsf{G}} + q_{\mathsf{H}} + 2)^2 \delta.$$

(We note that H and the challenge ciphertext also query to G internally.)
According to Lemma A.1, for any $p \geq 0$, we have

$$|\Pr[S_{5.1} \wedge \mathrm{Good}] - p| \leq |\Pr[S_{5.1}] - p| + \delta.$$

Game_6: This game is the same as Game_5 except that the decapsulation oracle is reset as Dec. Similar to the case for Game_2 and Game_3, if a key pair is good, the two games Game_5 and Game_6 are equivalent as in the proof of [XY19, Lemma 4.5]. We have

$$\Pr[S_{5.1} \wedge \mathrm{Good}] = \Pr[S_6 \wedge \mathrm{Good}].$$

$\mathsf{Game}_{6.1}$: This game is the same as Game_6 except that the random oracle H is simulated by $\mathsf{H}'_q \circ g$ and H' as in $\mathsf{Game}_{1.2}$. If Good occurs, the two games Game_6 and $\mathsf{Game}_{6.1}$ are equivalent. We have

$$\Pr[S_6 \wedge \mathrm{Good}] = \Pr[S_{6.1} \wedge \mathrm{Good}].$$

$\mathsf{Game}_{6.2}$: This game is the same as $\mathsf{Game}_{6.1}$ except that the random oracle $\mathsf{H}(\cdot)$ is set as the original. If Good occurs, the two games $\mathsf{Game}_{6.1}$ and $\mathsf{Game}_{6.2}$ are equivalent. We have

$$\Pr[S_{6.1} \wedge \mathrm{Good}] = \Pr[S_{6.2} \wedge \mathrm{Good}].$$

We also have, for any $p \geq 0$,

$$|\Pr[S_{6.2} \wedge \mathrm{Good}] - p| \leq |\Pr[S_{6.2}] - p| + \delta$$

from Lemma A.1.

Table 8. Summary of Games for the Proof of Theorem E.3: '$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$' implies that the challenger generates $c^* \leftarrow \mathcal{S}(1^\kappa)$ and returns $\perp$ if $c^* \in \mathsf{Enc}(ek, \mathcal{M})$.

Game	H	c^*	K^*	Decryption valid c	Decryption invalid c	justification
Game_0	H	$\mathcal{S}(1^\kappa)$	random	$\mathsf{H}(\mu, c)$	$\mathsf{H}(s, c)$	
Game_1	H	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	random	$\mathsf{H}(\mu, c)$	$\mathsf{H}(s, c)$	statistical disjointness
Game_2	H	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	random	$\mathsf{H}(\mu, c)$	$\mathsf{H}_q(c)$	Lemma 2.2
Game_3	H	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$\mathsf{H}_q(c^*)$	$\mathsf{H}(\mu, c)$	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c^*)$ is hidden
Game_4	H	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$\mathsf{H}(s, c^*)$	$\mathsf{H}(\mu, c)$	$\mathsf{H}(s, c)$	Lemma 2.2
Game_5	H	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$\overline{\mathsf{Dec}}(dk, c^*)$	$\mathsf{H}(\mu, c)$	$\mathsf{H}(s, c)$	re-encryption check
Game_6	H	$\mathcal{S}(1^\kappa)$	$\overline{\mathsf{Dec}}(dk, c^*)$	$\mathsf{H}(\mu, c)$	$\mathsf{H}(s, c)$	statistical disjointness

Game_7: This game is the same as $\mathsf{Game}_{6.2}$ except that the random oracle G is chosen from $\mathcal{F}(\mathcal{M}, \mathcal{R})$ instead of $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$. As in the argument in subsection B.2, we have,

$$|\Pr[S_{6.2}] - \Pr[S_7]| \le 8(q_G + q_{\mathrm{DEC}} + 1)^2\delta. \le 8(q_G + q_{\mathrm{DEC}} + 2)^2\delta.$$

Game_8: This game is the same as Game_7 except that $\mathsf{H}_q(c)$ in the decapsulation is replaced by $\mathsf{H}(s, c)$. According to Lemma 2.2 we have the bound

$$|\Pr[S_7] - \Pr[S_8]| \le 2(q_H + q_{\mathrm{DEC}})/\sqrt{|\mathcal{M}|}.$$

We note that this game is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 1$. Thus, we have

$$\Pr[S_8] = \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing those (in)equalities, we obtain the following bound:

$$\begin{aligned}\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_8]| \\ &\le \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4\delta \\ &\quad + 16(q_G + q_{\mathrm{DEC}} + 2)^2\delta + 16(q_G + q_H + 2)^2\delta + 4(q_H + q_{\mathrm{DEC}})/\sqrt{|\mathcal{M}|}.\end{aligned}$$

Proof of Theorem E.2: The proof of Theorem E.2 is a simplified version of that of Theorem E.1, since it does not require to consider G. Ignoring the transition between real G with good G, we obtain the bound as follows:

$$\begin{aligned}\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_8]| \\ &\le 4(q_H + q_{\mathrm{DEC}})/\sqrt{|\mathcal{M}|} + 4\delta + \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{A}_{34},\mathcal{S}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).\end{aligned}$$

E.2 SSMT-CCA Security

We can show the SSMT-CCA security of $\mathsf{U}^{\not\perp}$ by using the essentially same proof of that for SXY.

Theorem E.3. *Suppose that a ciphertext space $\mathcal{C}$ of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{U}^{\not\perp}[\mathsf{PKE}, \mathsf{H}]$ *is* SSMT-CCA*-secure.*
Formally speaking, for any adversary $\mathcal{A}$ against SSMT-CCA *security of* KEM*, we have*

$$\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) \le 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_H + q_{\mathrm{DEC}})/\sqrt{|\mathcal{M}|}.$$

Note that this security proof is independent of that PKE is deterministic PKE or one derandomized by T.

Proof Sketch: We use the game-hopping proof. We consider Game_i for $i = 0, \ldots, 6$. We summarize the games in Table 8. Let S_i denote the event that the adversary outputs $b' = 1$ in game Game_i. Let Acc and $\overline{\mathsf{Acc}}$ denote the event that the key pair (ek, dk) is accurate and inaccurate, respectively.

Game_0: This game is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa)$ with $b = 0$. The challenge is generated as

$$(c^*, K_0^*) \leftarrow \mathcal{S}(1^\kappa) \times \mathcal{K}.$$

We have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: In this game, the ciphertext is set as $\perp$ if c^* is in $\mathsf{Enc}(ek, \mathcal{M})$. The difference between two games Game_0 and Game_1 is bounded by statistical disjointness.

$$|\Pr[S_0] - \Pr[S_1]| \leq \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Game_2: This game is the same as Game_1 except that $\mathsf{H}(s, c)$ in the decapsulation oracle is replace with $\mathsf{H}_q(c)$ where $\mathsf{H}_q : C \to \mathcal{K}$ is another random oracle.
As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_1] - \Pr[S_2]| \leq 2(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|},$$

where q_{H} denote the number of queries to $\mathsf{H}_{\mathsf{prf}}$ the adversary makes.

Game_3: This game is the same as Game_2 except that $K^* := \mathsf{H}_q(c^*)$ instead of chosen random. Since c^* is always outside of $\mathsf{Enc}(ek, \mathcal{M})$, $\mathcal{A}$ cannot obtain any information about $\mathsf{H}_q(c^*)$. Hence, the two games Game_2 and Game_3 are equivalent and we have

$$\Pr[S_2] = \Pr[S_3].$$

Game_4: This game is the same as Game_3 except that $\mathsf{H}_q(\cdot)$ is replaced by $\mathsf{H}(s, \cdot)$. As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_3] - \Pr[S_4]| \leq 2(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|}.$$

Game_5: This game is the same as Game_4 except that $K^* := \overline{\mathsf{Dec}}(dk, c^*)$ instead of $\mathsf{H}(s, c^*)$. Recall that c^* is always in *outside* of $\mathsf{Enc}(ek, \mathcal{M})$. Thus, we always have $\mathsf{Dec}(c^*) = \perp$ or $\mathsf{Enc}(ek, \mathsf{Dec}(c^*)) \neq c^*$ and, thus, $K^* = \mathsf{H}(s, c^*)$. Hence, the two games are equivalent and we have

$$\Pr[S_4] = \Pr[S_5].$$

Game_6: We finally replace how to compute c^*. In this game, the ciphertext is chosen by $\mathcal{S}(1^\kappa)$ as in Game_0. The difference between two games Game_5 and Game_6 is bounded by statistical disjointness.

$$|\Pr[S_5] - \Pr[S_6]| \leq \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Moreover, this game Game_6 is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa)$ with $b = 1$.

$$\Pr[S_6] = \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing the (in)equalities, we obtain Theorem E.3:

$$\begin{aligned}\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{S},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_6]| \\ &\leq 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|}.\end{aligned}$$

E.3 SCFR-CCA Security

Theorem E.4. *If* PKE *is* SCFR-CCA*-secure (or* XCFR*-secure), then* $\mathsf{KEM} = \mathsf{U}^{\not\perp}[\mathsf{PKE}, \mathsf{H}]$ *is* SCFR-CCA*-secure in the QROM.*

Note that this security proof is irrelevant to PKE is deterministic PKE or one derandomized by T.

Proof. Suppose that an adversary outputs a ciphertext c which is decapsulated into $K \neq \perp$ by both $\overline{dk}_0$ and $\overline{dk}_1$, that is, $\overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c)$. Let us define $\mu'_i = \mathsf{Dec}(dk_i, c)$ for $i \in \{0, 1\}$. We also define $\mu_i := \mu'_i$ if $c = \mathsf{Enc}(ek_i, \mu'_i)$ and $\perp$ otherwise.
We have five cases defined as follows:

1. Case 1 ($\mu_0 = \mu_1 \neq \perp$): This violates the SCFR-CCA security (or the XCFR security) of the underlying PKE and it is easy to make a reduction.

2. Case 2 ($\bot \neq \mu_0 \neq \mu_1 \neq \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0, c) = \mathsf{H}(\mu_1, c)$. Thus, we succeed to find a collision for H, which is negligible for any QPT adversary (Lemma 2.3).
3. Case 3 ($\mu_0 = \bot$ and $\mu_1 \neq \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(s_0, c) = \mathsf{H}(\mu_1, c)$. Notice that we can replace $\mathsf{H}(s_0, \cdot)$ with $\mathsf{H}_q(\cdot)$ by introducing negligible error (Lemma 2.2). After that, we find a claw $(c, (\mu_1, c))$ between H_q and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
4. Case 4 ($\mu_0 \neq \bot$ and $\mu_1 = \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0, c) = \mathsf{H}(s_1, c)$. Again, we can replace $\mathsf{H}(s_1, \cdot)$ with $\mathsf{H}_q(\cdot)$ by introducing negligible error (Lemma 2.2). After that, we find a claw $((\mu_0, c), c)$ between H and H_q. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
5. Case 5 (The other cases): In this case, we find a collision $((s_0, c), (s_1, c))$ of H, which is indeed collision if $s_0 \neq s_1$ which occurs with probability at lease $1 - 1/2^\ell$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

We conclude that the advantage of the adversary is negligible in any cases. □

F Properties of $\mathsf{HU}^{\bot}_m$

In this section, we review $\mathsf{HU}^{\bot}_m$ [JZM19], which allows explicit rejection by using the additional 'key-confirmation' hash. Since $\mathsf{HU}^{\bot}_m$ is KEM with explicit rejection, we only consider the SPR-CCA security and smoothness.
Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme whose plaintext space is $\mathcal{M}$. Let $\mathcal{C}$ and $\mathcal{K}$ be a ciphertext and key space. Let $\mathcal{H}$ be a some finite space. Let $\mathsf{H}\colon \mathcal{M} \to \mathcal{K}$ and $\mathsf{F}\colon \mathcal{M} \to \mathcal{H}$ be hash functions modeled by random oracles. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{HU}^{\bot}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ obtained by using $\mathsf{HU}^{\bot}_m$ is defined as in Figure 13.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek)$
1 : $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	1 : $m \leftarrow \mathcal{D}_{\mathcal{M}}$	1 : $\mu' \leftarrow \mathsf{Dec}(dk, c_0)$
2 : $\overline{dk} := (dk, ek)$	2 : $c_0 := \mathsf{Enc}(ek, \mu)$	2 : **if** $\mu' = \bot$ or $c_0 \neq \mathsf{Enc}(ek, \mu')$ or $c_1 \neq \mathsf{F}(\mu'[, ek])$
3 : **return** $(ek, \overline{dk})$	3 : $c_1 := \mathsf{F}(\mu[, ek])$	3 : **then return** $K := \bot$
	4 : $K := \mathsf{H}(\mu)$	4 : **else return** $K := \mathsf{H}(\mu')$
	5 : **return** $((c_0, c_1), K)$	

Fig. 13. $\mathsf{KEM} = \mathsf{HU}^{\bot}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$

F.1 SPR-CCA Security

Theorem F.1 (Case of derandomized PKE). *Let* $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable with simulator* $\mathcal{S}$ *and* δ*-correct with negligible* δ*, then* $\mathsf{KEM} = \mathsf{HU}^{\bot}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SPR-CCA*-secure, where we use a new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$. *Formally speaking, for any* $\mathcal{A}$ *against the* SPR-CCA *security of* KEM *issuing at most* q_{Dec} *queries to the decapsulation oracle and* q_{F}, q_{G}, *and* q_{H} *queries to* F, G, *and* H, *respectively, there exists* $\mathcal{A}_{34}$ *against ciphertext-indistinguishability of* PKE *such that*

$$\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{S}',\mathcal{A}}(\kappa) \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 16(q_{\mathsf{G}} + q_{\mathrm{Dec}} + 2)^2\delta + 4\delta$$
$$+ 8(q_{\mathsf{G}} + q_{\mathsf{H}} + q_{\mathsf{F}} + 2)^2\delta + 8(q_{\mathsf{G}} + q_{\mathsf{H}} + q_{\mathsf{F}} + q_{\mathrm{Dec}} + 2)^2\delta + (4q_{\mathrm{Dec}} + 1)/|\mathcal{H}|.$$

Theorem F.2 (Case of non-derandomized PKE). *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable with simulator* $\mathcal{S}$ *and* δ*-correct with negligible* δ*, then* $\mathsf{KEM} = \mathsf{HU}^{\bot}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SPR-CCA*-secure, where we use a new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$. *Formally speaking, for any* $\mathcal{A}$ *against the* SPR-CCA *security of* KEM *issuing at most* q_{Dec} *queries to the decapsulation oracle and* q_{F} *and* q_{H} *queries to* F *and* H, *respectively, there exists* $\mathcal{A}_{34}$ *against ciphertext-indistinguishability of* PKE *such that*

$$\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{S}',\mathcal{A}}(\kappa) \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4\delta + (4q_{\mathrm{Dec}} + 1)/|\mathcal{H}|.$$

Table 9. Summary of Games for the Proof of Theorem F.1. We define $g(\mu) = \mathsf{Enc}(ek, \mu) = \mathsf{Enc}_0(ek, \mu; \mathsf{G}(\mu))$.

Game	H	F	G	c_0^*	c_1^*	K^*	Decapsulation K	Decapsulation condition	justification
Game_0	H	F	$\mathcal{F}(\mathcal{M}, \mathcal{R})$	$\mathsf{Enc}(ek, \mu^*)$	$\mathsf{F}(\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}(\mu)$	
$\mathsf{Game}_{0.1}$	H	F	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathsf{Enc}(ek, \mu^*)$	$\mathsf{F}(\mu^*)$	$\mathsf{H}(\mu^*)$	$\mathsf{H}(\mu)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}(\mu)$	Lemma 2.1 + correctness
Game_1	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathsf{Enc}(ek, \mu^*)$	$\mathsf{F}_q(c_0^*)$	$\mathsf{H}_q(c_0^*)$	$\mathsf{H}(\mu)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}(\mu)$	if key is not bad
Game_2	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathsf{Enc}(ek, \mu^*)$	$\mathsf{F}_q(c_0^*)$	$\mathsf{H}_q(c_0^*)$	$\mathsf{H}_q(c_0)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}_q(c_0)$	conceptual change
Game_3	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathsf{Enc}(ek, \mu^*)$	$\mathsf{F}_q(c_0^*)$	$\mathsf{H}_q(c_0^*)$	$\mathsf{H}_q(c_0)$	if $c_1 = \mathsf{F}_q(c_0)$	statistical
$\mathsf{Game}_{3.1}$	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}(\mathcal{M}, \mathcal{R})$	$\mathsf{Enc}(ek, \mu^*)$	$\mathsf{F}_q(c_0^*)$	$\mathsf{H}_q(c_0^*)$	$\mathsf{H}_q(c_0)$	if $c_1 = \mathsf{F}_q(c_0)$	Lemma 2.1 + correctness
Game_4	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$\mathsf{F}_q(c_0^*)$	$\mathsf{H}_q(c_0^*)$	$\mathsf{H}_q(c_0)$	if $c_1 = \mathsf{F}_q(c_0)$	DS-IND
Game_5	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$\mathsf{F}_q(c_0 *)$	$U(\mathcal{K})$	$\mathsf{H}_q(c_0)$	if $c_1 = \mathsf{F}_q(c_0)$	statistical disjointness
$\mathsf{Game}_{5.1}$	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}_q(c_0)$	if $c_1 = \mathsf{F}_q(c_0)$	statistical disjointness
$\mathsf{Game}_{5.2}$	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}_q(c_0)$	if $c_1 = \mathsf{F}_q(c_0)$	Lemma 2.1 + correctness
Game_6	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}_q(c_0)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}_q(c_0)$	statistical
Game_7	$\mathsf{H}_q \circ g$	$\mathsf{F}_q \circ g$	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}(\mu)$	conceptual change
$\mathsf{Game}_{7.1}$	H	F	$\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}(\mu)$	if key is not bad
Game_8	H	F	$\mathcal{F}(\mathcal{M}, \mathcal{R})$	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}(\mu)$	Lemma 2.1 + correctness

Proof of Theorem F.1: We use the game-hopping proof. We consider Game_i for $i = 0, \ldots, 8$. We summarize the games in Table 9. Let S_i denote the event that the adversary outputs $b' = 1$ in game Game_i.
We mainly follow the security proof in [JZM19, XY19, LW21], while we use a new simulator $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$ instead of $\mathcal{S}' = \mathsf{Enc}(ek, \mathcal{M}) \times U(\mathcal{H})$.

Game_0: This game is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 0$. By the definition, we have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

$\mathsf{Game}_{0.1}$: This game is the same as Game_0 except that the random oracle G is chosen from $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$ instead of $\mathcal{F}(\mathcal{M}, \mathcal{R})$, where $\mathcal{F}^{\mathrm{good}}_{ek,dk}(\mathcal{M}, \mathcal{R})$ is a set of functions $\mathsf{G} \colon \mathcal{M} \to \mathcal{R}$ satisfying $\mathsf{G}(\mu) \in \mathcal{R}^{\mathrm{good}}_{ek,dk,\mu}$ for all $\mu \in \mathcal{M}$ with $\mathcal{R}^{\mathrm{good}}_{ek,dk,\mu} := \{r \in \mathcal{R} : \mathsf{Dec}_0(dk, \mathsf{Enc}_0(ek, \mu; r)) = \mu\}$. We define $\mathsf{Bad} := \mathsf{boole}\left(\exists \mu \in \mathcal{M} : \mathcal{R}^{\mathrm{good}}_{ek,dk,\mu} = \emptyset\right)$ and $\mathsf{Good} := \neg\mathsf{Bad}$.
As in the proof of Theorem 4.1 in subsection B.2, we have

$$|\Pr[S_0] - \Pr[S_{0.1}]| \le 8(q_{\mathsf{G}} + q_{\mathrm{DEC}} + 2)^2 \delta.$$

In addition, we have $\Pr[\mathsf{Bad}] \le \delta$ and $|\Pr[S_{0.1}] - p| \le |\Pr[S_{0.1} \wedge \mathsf{Good}] - p| + \delta$ for any $p \in [0, 1]$.

Game_1: This game is the same as $\mathsf{Game}_{0.1}$ except that the random oracles H and F are simulated by $\mathsf{H}_q \circ g$ and $\mathsf{F}_q \circ g$, respectively, where $\mathsf{H}_q \colon \mathcal{C} \to \mathcal{K}$ and $\mathsf{F}_q \colon \mathcal{C} \to \mathcal{H}$ are random oracles and $g(\mu) := \mathsf{Enc}(ek, \mu)$. If Good occurs, then $\mathsf{H}_q \circ g$ and $\mathsf{F}_q \circ g$ are random functions and those two games are equal to each other. We have

$$\Pr[S_{0.1} \wedge \mathsf{Good}] = \Pr[S_1 \wedge \mathsf{Good}].$$

Game_2: This game is the same as Game_1 except that the decapsulation oracle internally computes c_1 as $\mathsf{F}_q(c_0)$ instead of $\mathsf{F}(\mu')$ and K as $\mathsf{H}_q(c_0)$ instead of $\mathsf{H}(\mu')$, where $\mu' = \mathsf{Dec}(dk, c_0)$, that is, we rewrite the line 2 of $\overline{\mathsf{Dec}}$ with "if $\mu' = \perp$ or $c_0 \neq \mathsf{Enc}(ek, \mu')$ or $c_1 \neq \mathsf{F}(\mu')$" and the line 4 of $\overline{\mathsf{Dec}}$ with "else return $K := \mathsf{H}_q(c_0)$."
If the two conditions $\mu' \neq \perp$ and $c_0 = \mathsf{Enc}(ek, \mu')$ are satisfied, then the former change is just conceptual since we set $\mathsf{F} = \mathsf{F}_q \circ g$ in the previous game and we have $\mathsf{F}_q(c_0) = \mathsf{F}_q(\mathsf{Enc}(ek, \mu')) = (\mathsf{F}_q \circ g)(\mu')$. The latter change is also conceptual since we set $\mathsf{H} = \mathsf{H}_q \circ g$ in the previous game and we have $\mathsf{H}_q(c_0) = \mathsf{H}_q(\mathsf{Enc}(ek, \mu')) = (\mathsf{H}_q \circ g)(\mu')$. Thus, we have

$$\Pr[S_1 \wedge \mathsf{Good}] = \Pr[S_2 \wedge \mathsf{Good}].$$

Game_3: In this game the decapsulation oracle ignores the condition "$\mu' = \perp$ or $c_0 \neq \mathsf{Enc}(ek, \mu')$," that is, we rewrite the line 2 of $\overline{\mathsf{Dec}}$ with "if $c_1 \neq \mathsf{F}(\mu')$." By this modification, when $(c_0, c_1) \neq (c_0^*, c_1^*)$, the oracle returns $K = \mathsf{H}_q(c_0)$ if $c_1 = \mathsf{F}_q(c_0)$.
Let us consider the following three cases:

- Case 1 ($c_0 = \mathsf{Enc}(ek, \mu')$ for some μ'): In this case, the answers of the decapsulation oracles in both games are equal to each other.
- Case 2 ($c_0 \notin \mathsf{Enc}(ek, \mathcal{M})$ and $c_1 \neq \mathsf{F}_q(c_0)$): In this case, the answers of the decapsulation oracles in both games are $\perp$.
- Case 3 ($c_0 \notin \mathsf{Enc}(ek, \mathcal{M})$ and $c_1 = \mathsf{F}_q(c_0)$): In this case, the answer in Game_2 is $\perp$, but the answer in Game_3 is $K = \mathsf{H}_q(c)$.

Thus, the difference occurs when c_0 is outside of $\mathsf{Enc}(ek, \mathcal{M})$ and $c_1 = \mathsf{F}_q(c_0)$. Notice that the adversary cannot access such hash values $\mathsf{F}_q(\mathcal{C} \setminus \mathsf{Enc}(ek, \mathcal{M}))$ directly, since it is given F instead of F_q. Therefore, any c_1 hits the value $\mathsf{F}_q(c_0)$ with probability at most $1/|\mathcal{H}|$ and we obtain the bound $q_{\mathrm{Dec}}/|\mathcal{H}|$. (If a decapsulation query is quantum, we will get another bound $2q_{\mathrm{Dec}}/\sqrt{|\mathcal{H}|}$.) We have

$$|\Pr[S_2 \wedge \mathrm{Good}] - \Pr[S_3 \wedge \mathrm{Good}]| \leq q_{\mathrm{Dec}}/|\mathcal{H}|.$$

We also have for any $p \geq 0$,

$$|\Pr[S_3 \wedge \mathrm{Good}] - p| \leq |\Pr[S_3] - p| + \delta.$$

$\mathsf{Game}_{3.1}$: This game is the same as Game_3 except that G is chosen from $\mathcal{F}(\mathcal{M}, \mathcal{R})$. We have

$$|\Pr[S_3] - \Pr[S_{3.1}]| \leq 8(q_{\mathsf{G}} + q_{\mathsf{H}} + q_{\mathsf{F}} + q_{\mathrm{Dec}} + 2)^2\delta$$

as in the proof of Theorem 4.1 in subsection B.2. (We note that H, F, Dec, and the challenge ciphertext also query to G internally.)

Game_4: We replace $c_0^* := \mathsf{Enc}(ek, \mu^*; \mathsf{G}(\mu^*))$ with $c_0^* \leftarrow \mathcal{S}(1^\kappa)$. The difference is bounded by the advantage of ciphertext indistinguishability and we have an quantum adversary $\mathcal{A}_{34}$ satisfying

$$|\Pr[S_{3.1}] - \Pr[S_4]| \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}, \mathcal{D}_M, \mathcal{S}, \mathcal{A}_{34}}(\kappa).$$

We omit the detail of the reduction algorithm since it is easy to construct.

Game_5: This game is the same as Game_4 except that $K^* \leftarrow \mathcal{K}$ instead of $K^* \leftarrow \mathsf{H}_q(c_0^*)$.
We note that the adversary cannot access to $K^* = \mathsf{H}_q(c_0^*)$ via H if c_0^* is outside of $\mathsf{Enc}(ek, \mathcal{M})$ in both games: Let (c_0, c_1) be a query to Dec the adversary makes. If $c_0 = c_0^*$ and $c_1 = c_1^*$, then the adversary receives $\perp$ in both games. If $c_0 = c_0^*$ and $c_1 \neq c_1^*$, then $c_1 \neq \mathsf{F}_q(c_0^*) = c_1^*$ holds and the adversary receives $\perp$ in both games. Thus, if c_0^* is outside of $\mathsf{Enc}(ek, \mathcal{M})$, the two games are equal to each other. Hence, the difference is bounded by the statistical disjointness in disjoint simulatability. We have

$$|\Pr[S_4] - \Pr[S_5]| \leq \mathsf{Disj}_{\mathsf{PKE}, \mathcal{S}}(\kappa).$$

$\mathsf{Game}_{5.1}$: This game is the same as $\mathsf{Game}_{5.1}$ except that $c_1^* \leftarrow U(\mathcal{H})$ instead of $c_1^* := \mathsf{F}_q(c_0^*)$, in which our proof is different from that of Jiang et al. [JZM19].
When the adversary queries (c_0, c_1) for $c_0 \neq c_0^*$, there is no leak on $\mathsf{F}_q(c_0^*)$. In addition, when c_0^* is the outside of $\mathsf{Enc}(ek, \mathcal{M})$, the adversary cannot obtain the real hash value $c_1^* = \mathsf{F}_q(c_0^*)$ directly.
Suppose that c_0^* is the outside of $\mathsf{Enc}(ek, \mathcal{M})$. We consider the case that the adversary queries (c_0^*, c_1) for Dec.

- In Game_5, we have $c_1^* = \mathsf{F}_q(c_0^*)$. If $c_1 = c_1^*$, then the adversary receives $\perp$; otherwise, that is, if $c_1 \neq c_1^*$, it also receives $\perp$.
- In $\mathsf{Game}_{5.1}$, we have $c_1^* \leftarrow U(\mathcal{H})$.
  - If $c_1^* = \mathsf{F}_q(c_0^*)$, then this game is the same as Game_5.
  - Suppose that $c_1^* \neq \mathsf{F}_q(c_0^*)$. If $c_1 = c_1^*$, then the adversary receives $\perp$; otherwise, it receives $\perp$ if and only if $c_1 \neq \mathsf{F}_q(c_0^*)$; it receives $K = \mathsf{H}_q(c_0^*)$ if $c_1 = \mathsf{F}_q(c_0^*)$.

Assuming that c_0^* is the outside of $\mathsf{Enc}(ek, \mathcal{M})$ and $c_1^* \neq \mathsf{F}_q(c_0^*)$, a value c_1 hits $\mathsf{F}_q(c_0^*)$ with probability at most $1/(|\mathcal{H}| - 1)$. Thus, w have

$$|\Pr[S_5] - \Pr[S_{5.1}]| \leq \mathsf{Disj}_{\mathsf{PKE}, \mathcal{S}}(\kappa) + 1/|\mathcal{H}| + q_{\mathrm{Dec}}/(|\mathcal{H}| - 1).$$

$\mathsf{Game}_{5.2}$: This game is the same as $\mathsf{Game}_{5.1}$ except that G is chosen from $\mathcal{F}^{\mathrm{good}}_{ek, dk}(\mathcal{M}, \mathcal{R})$.
As in the proof of Theorem 4.1 in subsection B.2, we have

$$|\Pr[S_{5.1}] - \Pr[S_{5.2}]| \leq 8(q_{\mathsf{G}} + q_{\mathsf{H}} + q_{\mathsf{F}} + 2)^2\delta.$$

We also have, for any $p \geq 0$,

$$|\Pr[S_{5.2}] - p| \leq |\Pr[S_{5.2} \wedge \mathrm{Good}] - p| + \delta.$$

Game_6: This game is the same as $\mathsf{Game}_{5.2}$ except that the decapsulation algorithm checks if $c_0 = \mathsf{Enc}(ek, \mu)$ and $c_1 = \mathsf{F}_q(c_0)$ or not.
As in the argument for the difference between Game_2 and Game_3, we consider the following three cases for a decapsulation query (c_0, c_1):

- Case 1 ($c_0 = \mathsf{Enc}(ek, \mu)$ for some μ): In this case, the answers of the decapsulation oracles in both games are equal to each other.
- Case 2 ($c_0 \notin \mathsf{Enc}(ek, \mathcal{M})$ and $c_1 \neq \mathsf{F}_q(c_0)$): In this case, the answers of the decapsulation oracles in both games are $\perp$.
- Case 3 ($c_0 \notin \mathsf{Enc}(ek, \mathcal{M})$ and $c_1 = \mathsf{F}_q(c_0)$): In this case, the answer in $\mathsf{Game}_{5.2}$ is $K = \mathsf{H}_q(c)$, but the answer in Game_6 is $\perp$.

Thus, the difference occurs when c_0 is outside of $\mathsf{Enc}(ek, \mathcal{M})$ and $c_1 = \mathsf{F}_q(c_0)$. Notice that the adversary cannot access such hash values directly, since it is given F instead of F_q. Therefore, any c_1 hits the value $\mathsf{F}_q(c_0)$ with probability at most $1/|\mathcal{H}|$ and we obtain the bound $q_{\mathrm{Dec}}/|\mathcal{H}|$. (If the query is quantum, we will get another bound $2q_{\mathrm{Dec}}(|\mathcal{H}|)^{-1/2}$.) We have

$$|\Pr[S_{5.2} \wedge \mathrm{Good}] - \Pr[S_6 \wedge \mathrm{Good}]| \leq q_{\mathrm{Dec}}/|\mathcal{H}|.$$

Game_7: This game is the same as Game_6 except that the decapsulation oracle use H and F instead of H_q and F_q, respectively. As in the argument for Game_1 and Game_2, if the key pair is good, then this is the conceptual change and we have

$$\Pr[S_6 \wedge \mathrm{Good}] = \Pr[S_7 \wedge \mathrm{Good}].$$

$\mathsf{Game}_{7.1}$: This game is the same as Game_7 except that H and F are modified as the original. As in the argument for $\mathsf{Game}_{0.1}$ and Game_1, if the key pair is good, then this is the conceptual change and we have

$$\Pr[S_7 \wedge \mathrm{Good}] = \Pr[S_{7.1} \wedge \mathrm{Good}].$$

We also have, for any $p \geq 0$,

$$|\Pr[S_{7.1} \wedge \mathrm{Good}] - p| \leq |\Pr[S_{7.1}] - p| + \delta.$$

Game_8: This game is the same as $\mathsf{Game}_{7.1}$ except that the random oracle G is chosen from $\mathcal{F}(\mathcal{M}, \mathcal{R})$. As in the argument for Game_0 and $\mathsf{Game}_{0.1}$, we have

$$|\Pr[S_{7.1}] - \Pr[S_8]| \leq 8(q_{\mathsf{G}} + q_{\mathrm{Dec}} + 1)^2\delta.$$

We note that this game is the original game $\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 1$. We have

$$\Pr[S_8] = \Pr[\mathsf{Expt}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summary: Summing those (in)equalities, we obtain the following bound:

$$\begin{aligned}
\mathsf{Adv}^{\text{spr-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_8]| \\
&\leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{A}_{34}}(\kappa) + 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 16(q_{\mathsf{G}} + q_{\mathrm{Dec}} + 2)^2\delta + 4\delta \\
&\quad + 8(q_{\mathsf{G}} + q_{\mathsf{H}} + q_{\mathsf{F}} + 2)^2\delta + 8(q_{\mathsf{G}} + q_{\mathsf{H}} + q_{\mathsf{F}} + q_{\mathrm{Dec}} + 2)^2\delta \\
&\quad + (2q_{\mathrm{Dec}} + 1)/|\mathcal{H}| + q_{\mathrm{Dec}}/(|\mathcal{H}| - 1)
\end{aligned}$$

and we replace $(2q_{\mathrm{Dec}} + 1)/|\mathcal{H}| + q_{\mathrm{Dec}}/(|\mathcal{H}| - 1)$ with $(4q_{\mathrm{Dec}} + 1)/|\mathcal{H}|$.

F.2 Sparseness

Theorem F.3. *Suppose that a ciphertext space C of* PKE *depends on the public parameter only. Let* $\mathsf{KEM} = \mathsf{HU}^{\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$. *Let $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$ be the simulator for* SPR-CCA *security of* KEM. *Then,* KEM *is $1/|\mathcal{H}|$-sparse.*

Proof. Let us consider $(c_0, c_1) \leftarrow \mathcal{S}(1^\kappa) \times U(\mathcal{H})$. If c_0 is decrypted into $\mu' \neq \perp$, then $c_1 = \mathsf{F}(\mu')$ with probability at most $1/|\mathcal{H}|$. Thus, KEM is $1/|\mathcal{H}|$-sparse. □

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek)$
1 : $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	1 : $m \leftarrow \mathcal{M}$	1 : $\mu' \leftarrow \mathsf{Dec}(dk, c_0)$
2 : $\overline{dk} := (dk, ek)$	2 : $c_0 := \mathsf{Enc}(ek, \mu)$	2 : **if** $\mu' = \bot$ or $c_0 \neq \mathsf{Enc}(ek, \mu')$ or $c_1 \neq \mathsf{F}(\mu'[, ek])$
3 : **return** $(ek, \overline{dk})$	3 : $c_1 := \mathsf{F}(\mu[, ek])$	3 : **then return** $K := \bot$
	4 : $K := \mathsf{H}(\mu, c_0, c_1)$	4 : **else return** $K := \mathsf{H}(\mu', c_0, c_1)$
	5 : **return** $((c_0, c_1), K)$	

Fig. 14. $\mathsf{KEM} = \mathsf{HU}^\bot[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$

G Properties of $\mathsf{HU}^\bot$

In this section, we consider a variant of HU with explicit rejection, $\mathsf{HU}^\bot$. Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme whose plaintext space is $\mathcal{M}$. Let $\mathcal{C}$ and $\mathcal{K}$ be a ciphertext and key space. Let $\mathcal{H}$ be a some finite space. Let $\mathsf{H} \colon \mathcal{M} \times \mathcal{C} \times \mathcal{H} \to \mathcal{K}$ and $\mathsf{F} \colon \mathcal{M} \to \mathcal{H}$ be hash functions modeled by random oracles. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{HU}^\bot[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ is defined as follows:

G.1 SPR-CCA security:

In order to show the SPR-CCA security of $\mathsf{HU}^\bot$, we consider the following theorem on indifferentiable reduction, which is obtained by mimicking that for $\mathsf{U}^x_m \leftrightarrow \mathsf{U}^x$ in [BHH+19, Theorem 5].

Theorem G.1 ($\mathsf{HU}^\bot_m \leftrightarrow \mathsf{HU}^\bot$). *Let* PKE *be a deterministic PKE. Let* $\mathsf{KEM}_m = \mathsf{HU}^\bot_m[\mathsf{PKE}, \mathsf{H}_m, \mathsf{F}]$ *and* $\mathsf{KEM} = \mathsf{HU}^\bot[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$.

1. *If* KEM_m *is* SPR-CCA*-secure, then* KEM *is* SPR-CCA*-secure also.*
2. *If* KEM *is* SPR-CCA*-secure, then* KEM_m *is* SPR-CCA*-secure also.*

Proof (The first part). Suppose that we have an adversary $\mathcal{A}$ against the SPR-CCA security of KEM. We construct an adversary $\mathcal{A}_m$ against the SPR-CCA security of KEM_m with random oracle $\mathsf{H}_m \colon \mathcal{M} \to \mathcal{K}$ as follows: $\mathcal{A}_m$ samples a fresh random oracle $\mathsf{H}' \leftarrow \mathsf{Func}(\mathcal{M} \times \mathcal{C} \times \mathcal{H}, \mathcal{K})$ and set

$$\mathsf{H}(\mu, c_0, c_1) = \begin{cases} \mathsf{H}_m(\mu) & \text{if } c_0 = \mathsf{Enc}(ek, \mu) \text{ and } c_1 = \mathsf{F}(\mu) \\ \mathsf{H}'(\mu, c_0, c_1) & \text{otherwise.} \end{cases}$$

This simulation is perfect and we conclude the proof. □

Proof (The second part). Suppose that we have an adversary $\mathcal{A}_m$ against the SPR-CCA security of KEM_m. We construct an adversary $\mathcal{A}$ against the SPR-CCA security of KEM with random oracle $\mathsf{H} \colon \mathcal{M} \times (\mathcal{C} \times \mathcal{H}) \to \mathcal{K}$ as follows: $\mathcal{A}$ define

$$\mathsf{H}_m(\mu) := \mathsf{H}\big(\mu, \mathsf{Enc}(ek, \mu), \mathsf{F}(\mu)\big).$$

This simulation is perfect and we conclude the proof. □

We obtain the following theorems by combining the above theorem with Theorem F.1 and Theorem F.2:

Theorem G.2 (Case of derandomized PKE). *Let* $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable with simulator* $\mathcal{S}$*, then* $\mathsf{KEM} = \mathsf{HU}^\bot[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SPR-CCA*-secure, where we use the new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$.

Theorem G.3 (Case of non-derandomized PKE). *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{HU}^\bot[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SPR-CCA*-secure.*

G.2 Sparseness

$\mathsf{KEM} = \mathsf{HU}^\bot[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ is $1/|\mathcal{H}|$-sparse as $\mathsf{HU}^\bot_m$.

Theorem G.4. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. Let* $\mathsf{KEM} = \mathsf{HU}^\bot[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$. *Let* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$ *be the simulator for* SPR-CCA *security of* KEM. *Then,* KEM *is* $1/|\mathcal{H}|$*-sparse.*

Proof. Let us consider $(c_0, c_1) \leftarrow \mathcal{S}(1^\kappa) \times U(\mathcal{H})$. If c_0 is decrypted into $\mu' \neq \bot$, then $c_1 = \mathsf{F}(\mu')$ with probability at most $1/|\mathcal{H}|$. Thus, KEM is $1/|\mathcal{H}|$-sparse. □

H Properties of $\mathsf{HU}^{\not\perp}_m$

Let us review $\mathsf{HU}^{\not\perp}_m$. Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme whose plaintext space is $\mathcal{M}$. Let $\mathcal{C}$ and $\mathcal{K}$ be a ciphertext and key space. Let $\mathcal{H}$ be a some finite space. Let $\mathsf{H}\colon \mathcal{M} \to \mathcal{K}$, $\mathsf{H}_{\mathsf{prf}}\colon \{0,1\}^\ell \times \mathcal{C} \times \mathcal{H} \to \mathcal{K}$, and $\mathsf{F}\colon \mathcal{M} \to \mathcal{H}$ be hash functions modeled by random oracles. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ is defined as in Figure 15.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek, s)$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	$m \leftarrow \mathcal{D}_{\mathcal{M}}$	$\mu' \leftarrow \mathsf{Dec}(dk, c_0)$
$s \leftarrow \{0,1\}^\ell$	$c_0 := \mathsf{Enc}(ek, m)$	**if** $\mu' = \perp$ or $c_0 \neq \mathsf{Enc}(ek, \mu')$ or $c_1 \neq \mathsf{F}(\mu'[, ek])$
$\overline{dk} := (dk, ek, s)$	$c_1 := \mathsf{F}(\mu[, ek])$	**then return** $K := \mathsf{H}_{\mathsf{prf}}(s, (c_0, c_1))$
return $(ek, \overline{dk})$	$K := \mathsf{H}(\mu)$	**else return** $K := \mathsf{H}(\mu')$
	return $((c_0, c_1), K)$	

Fig. 15. $\mathsf{KEM} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$

H.1 SPR-CCA Security

Bindel et al. showed that if $\mathsf{KEM}^\perp = \mathsf{U}^\perp_m[\mathsf{PKE}, \mathsf{H}]$ is IND-CCA-secure then $\mathsf{KEM}^{\not\perp} = \mathsf{U}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{H}_{\mathsf{prf}}]$ is also IND-CCA-secure [BHH+19, Theorem 3] by overwriting $\perp$ from the decapsulation query c with the PRF value $\mathsf{H}_{\mathsf{prf}}(s, c)$. The same indifferentiable reduction can be applied to the SPR-CCA security of $\mathsf{HU}^\perp_m$ and $\mathsf{HU}^{\not\perp}_m$, and we obtain the following theorem.

Theorem H.1 ($\mathsf{HU}^\perp_m \to \mathsf{HU}^{\not\perp}_m$). *Let* PKE *be a deterministic PKE. Let* $\mathsf{KEM}^\perp = \mathsf{HU}^\perp_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *and* $\mathsf{KEM}^{\not\perp} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$. *If* $\mathsf{KEM}^\perp$ *is* SPR-CCA*-secure, then* $\mathsf{KEM}^{\not\perp}$ *is also* SPR-CCA*-secure.*

Proof. Suppose that we have an adversary $\mathcal{A}$ against the SPR-CCA security of $\mathsf{KEM}^{\not\perp}$. We construct an adversary $\mathcal{A}'$ against the SPR-CCA security of $\mathsf{KEM}^\perp$ as follows: Given an encapsulation key ek, a target ciphertext (c_0^*, c_1^*), and a key K_b^*, $\mathcal{A}'$ samples a fresh seed $s \leftarrow \mathcal{M}$. It runs $\mathcal{A}$ on input ek, (c_0^*, c_1^*), and K_b^*. If $\mathcal{A}$ queries a ciphertext (c_0, c_1) to the decapsulation oracle, then $\mathcal{A}'$ queries the ciphertext (c_0, c_1) and receives K. If $K \neq \perp$, then it returns K to $\mathcal{A}$; Otherwise, it queries $(s, (c_0, c_1))$ to the random oracle $\mathsf{H}_{\mathsf{prf}}$, receives $\tilde{K}$, and returns $\tilde{K}$ to $\mathcal{A}$. If $\mathcal{A}$ outputs b' and halts, then $\mathcal{A}'$ also outputs b' and halts.
This simulation is clearly perfect and the theorem follows. □

Apply the above indifferentiable reduction with Theorem F.1 and Theorem F.2, we obtain the following theorems:

Theorem H.2 (Case of derandomized PKE). *Let* $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable with simulator* $\mathcal{S}$*, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SPR-CCA*-secure, where we use the new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$.

Theorem H.3 (Case of non-derandomized PKE). *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SPR-CCA*-secure, where we use the new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$.

H.2 SSMT-CCA Security

Theorem H.4. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SSMT-CCA*-secure.*
Formally speaking, for any $\mathcal{A}$*, we have*

$$\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) \leq 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}}) \cdot 2^{-\ell/2}.$$

The security proof is essentially same as that for SXY (Theorem 4.3). Note that this security proof is irrelevant to PKE is deterministic PKE or one derandomized by T.

Table 10. Summary of Games for the Proof of Theorem H.4: $\mathsf{Enc}'(ek, \mathcal{M}) = \{(c_0, c_1) = (\mathsf{Enc}(ek, m), \mathsf{F}(\mu) \mid m \in \mathcal{M}\}$. '$\mathcal{S}(1^\kappa) \times U(\mathcal{H}) \setminus \mathsf{Enc}'(ek, \mathcal{M})$' implies that the challenger generates $c_0^* \leftarrow \mathcal{S}(1^\kappa)$, $c_1^* \leftarrow \mathcal{H}$ and returns $\perp$ if $(c_0^*, c_1^*) \in \mathsf{Enc}'(ek, \mathcal{M})$.

Game	H	F	c_0^*	c_1^*	K^*	Decryption valid (c_0, c_1)	Decryption invalid (c_0, c_1)	justification
Game_0	H	F	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$	
Game_1	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$	statistical disjointness
Game_2	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c_0, c_1)$	Lemma 2.2
Game_3	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$\mathsf{H}_q(c_0^*, c_1^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_q(c_0, c_1)$	$\mathsf{H}_q(c_0^*, c_1^*)$ is hidden
Game_4	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$\mathsf{H}_{\mathsf{prf}}(s, c_0^*, c_1^*)$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$	Lemma 2.2
Game_5	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0^*, c_1^*))$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$	re-encryption check
Game_6	H	F	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0^*, c_1^*))$	$\mathsf{H}(\mu)$	$\mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$	statistical disjointness

Game_0: This game is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 0$. The challenge is generated as

$$(c_0^*, c_1^*, K_0^*) \leftarrow \mathcal{S}(1^\kappa) \times U(\mathcal{H}) \times \mathcal{K}.$$

We have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: In this game, the ciphertext is set as $\perp$ if c_0^* is in $\mathsf{Enc}(ek, \mathcal{M})$.
The difference between two games Game_0 and Game_1 is bounded by statistical disjointness.

$$|\Pr[S_0] - \Pr[S_1]| \le \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Game_2: This game is the same as Game_1 except that $\mathsf{H}_{\mathsf{prf}}(s, c, d)$ in the decapsulation oracle is replace with $\mathsf{H}_q(c_0, c_1)$ where $\mathsf{H}_q : \mathcal{C} \times \mathcal{H} \to \mathcal{K}$ is another random oracle.
As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_1] - \Pr[S_2]| \le 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}}) \cdot 2^{-\ell/2},$$

where $q_{\mathsf{H}_{\mathsf{prf}}}$ denote the number of queries to $\mathsf{H}_{\mathsf{prf}}$ the adversary makes.

Game_3: This game is the same as Game_2 except that $K^* := \mathsf{H}_q(c_0^*, c_1^*)$ instead of chosen random. Since c_0^* is always outside of $\mathsf{Enc}(ek, \mathcal{M})$, $\mathcal{A}$ cannot obtain any information about $\mathsf{H}_q(c_0^*, c_1^*)$ via the decapsulation oracle. Hence, the two games Game_2 and Game_3 are equivalent and we have

$$\Pr[S_2] = \Pr[S_3].$$

Game_4: This game is the same as Game_3 except that $\mathsf{H}_q(\cdot, \cdot)$ is replaced by $\mathsf{H}_{\mathsf{prf}}(s, \cdot, \cdot)$. As in [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_3] - \Pr[S_4]| \le 2(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}}) \cdot 2^{-\ell/2}.$$

Game_5: This game is the same as Game_4 except that $K^* := \overline{\mathsf{Dec}}(\overline{dk}, (c_0^*, c_1^*))$ instead of $\mathsf{H}_{\mathsf{prf}}(s, c_0^*, c_1^*)$. Recall that c_0^* is always in *outside* of $\mathsf{Enc}(ek, \mathcal{M})$. Thus, we always have $\mathsf{Dec}(c_0^*) = \perp$ or $\mathsf{Enc}(ek, \mathsf{Dec}(c_0^*)) \neq c_0^*$ and, thus, $K^* = \mathsf{H}_{\mathsf{prf}}(s, c_0^*, c_1^*)$. Hence, the two games are equivalent. We have

$$\Pr[S_4] = \Pr[S_5].$$

Game_6: We finally replace how to compute (c_0^*, c_1^*). In this game, the ciphertext is chosen by $\mathcal{S}(1^\kappa) \times U(\mathcal{H})$ as in Game_0.
The difference between two games Game_5 and Game_6 is bounded by statistical disjointness.

$$|\Pr[S_5] - \Pr[S_6]| \le \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Moreover, this game Game_6 is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 1$.

$$\Pr[S_6] = \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summarizing the (in)equalities, we obtain Theorem H.4:

$$\begin{aligned}\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_6]| \\ &\le 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{Dec}}) \cdot 2^{-\ell/2}.\end{aligned}$$

H.3 SCFR-CCA Security

Theorem H.5. *If* PKE *is* SCFR-CCA*-secure (or* XCFR*-secure), then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SCFR-CCA*-secure in the quantum random oracle model.*

Note that this security proof is independent of that PKE is deterministic PKE or one derandomized by T.

Proof. Suppose that an adversary outputs a ciphertext $c = (c_0, c_1)$ which is decapsulated into $K \neq \perp$ by $\overline{dk}_0$ and $\overline{dk}_1$, that is, $\overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c)$. Let us define $\mu'_i = \mathsf{Dec}(dk_i, c_0)$ for $i \in \{0,1\}$. We also define $\mu_i = \mu'_i$ if $c_0 = \mathsf{Enc}(ek_i, \mu'_i)$ and $c_1 = \mathsf{F}(\mu'_i)$, and $\perp$ otherwise.
We have five cases defined as follows:

1. Case 1 ($\mu_0 = \mu_1 \neq \perp$): This violates the SCFR-CCA security (or the XCFR security) of the underlying PKE.
2. Case 2 ($\perp \neq \mu_0 \neq \mu_1 \neq \perp$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0) = \mathsf{H}(\mu_1)$ and we succeed to find a collision for H and F, which is negligible for any QPT adversary (Lemma 2.3).
3. Case 3 ($\mu_0 = \perp$ and $\mu_1 \neq \perp$): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1)$ and $\mathsf{H}(\mu_1)$ and we find a claw $((s_0, c_0, c_1), \mu_1)$ of $\mathsf{H}_{\mathsf{prf}}$ and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
4. Case 4 ($\mu_0 \neq \perp$ and $\mu_1 = \perp$): In this case, the decapsulation algorithms output $K = \mathsf{H}(\mu_0) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a claw $(\mu_0, (s_1, c_0, c_1))$ of H and $\mathsf{H}_{\mathsf{prf}}$. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
5. Case 5 (The other cases): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a collision $((s_0, c_0, c_1), (s_1, c_0, c_1))$ of $\mathsf{H}_{\mathsf{prf}}$ if $s_0 \neq s_1$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

We conclude that the advantage of the adversary is negligible in any cases. □

If we add *ek* to F's input, we can reduce the assumption on PKE.

Theorem H.6. *Let* $\mathsf{Col}_{\mathsf{Gen}}$ *be the event that when generating two keys* $(ek_i, dk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ *for* $i \in \{0,1\}$*, they collide, that is,* $ek_0 = ek_1$*. If* $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ *is negligible, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *with* $c_1 = \mathsf{F}(\mu, ek)$ *is* SCFR-CCA*-secure in the quantum random oracle model.*

Note that this security proof is irrelevant to PKE is deterministic PKE or one derandomized by T.

Proof. Suppose that an adversary outputs a ciphertext $c = (c_0, c_1)$ which is decapsulated into $K \neq \perp$ by $\overline{dk}_0$ and $\overline{dk}_1$, that is, $\overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c)$. Let us define $\mu'_i = \mathsf{Dec}(dk_i, c_0)$ for $i \in \{0,1\}$. We also define $\mu_i = \mu'_i$ if $c_0 = \mathsf{Enc}(ek_i, \mu'_i)$ and $c_1 = \mathsf{F}(\mu'_i, ek_i)$, and $\perp$ otherwise.
We consider six cases defined as follows:

1. Case 1-1 ($\mu_0 = \mu_1 \neq \perp$ and $ek_0 = ek_1$): This case rarely occurs since $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ is negligible.
2. Case 1-2 ($\mu_0 = \mu_1 \neq \perp$ and $ek_0 \neq ek_1$): In this case, we have $d = \mathsf{F}(\mu'_0, ek_0) = \mathsf{F}(\mu'_1, ek_1)$ with $(\mu'_0, ek_0) \neq (\mu'_1, ek_1)$ and we succeed to find a collision for F, which is negligible for any QPT adversary (Lemma 2.3).
3. Case 2 ($\perp \neq \mu_0 \neq \mu_1 \neq \perp$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0) = \mathsf{H}(\mu_1)$ and we succeed to find a collision for H and F, which is negligible for any QPT adversary (Lemma 2.3).
4. Case 3 ($\mu_0 = \perp$ and $\mu_1 \neq \perp$): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1)$ and $\mathsf{H}(\mu_1)$ and we find a claw $((s_0, c_0, c_1), \mu_1)$ of $\mathsf{H}_{\mathsf{prf}}$ and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
5. Case 4 ($\mu_0 \neq \perp$ and $\mu_1 = \perp$): In this case, the decapsulation algorithms output $K = \mathsf{H}(\mu_0) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a claw $(\mu_0, (s_1, c_0, c_1))$ of H and $\mathsf{H}_{\mathsf{prf}}$. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
6. Case 5 (The other cases): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a collision $((s_0, c_0, c_1), (s_1, c_0, c_1))$ of $\mathsf{H}_{\mathsf{prf}}$ if $s_0 \neq s_1$, which occurs with probability at least $1 - 1/2^\ell$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

We conclude that the advantage of the adversary is negligible in any cases. □

I Properties of $\mathsf{HU}^{\not\perp,\mathsf{prf}}$

Next, we consider a variant of HU with implicit rejection, $\mathsf{HU}^{\not\perp,\mathsf{prf}}$, which is used in Classic McEliece. Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme whose plaintext space is $\mathcal{M}$. Let $\mathcal{C}$ and $\mathcal{K}$ be a ciphertext and key space. Let $\mathcal{H}$ be a some finite space. Let $\mathsf{H}, \mathsf{H}_{\mathsf{prf}} \colon \mathcal{M} \times \mathcal{C} \times \mathcal{H} \to \mathcal{K}$ and $\mathsf{F} \colon \mathcal{M} \to \mathcal{H}$ be hash functions modeled by random oracles. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ is defined as in Figure 16.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek, s)$
1 : $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	1 : $\mu \leftarrow \mathcal{M}$	1 : $\mu' \leftarrow \mathsf{Dec}(dk, c_0)$
2 : $s \leftarrow \mathcal{M}$	2 : $c_0 := \mathsf{Enc}(ek, \mu)$	2 : **if** $\mu' = \bot$ or $c \neq \mathsf{Enc}(ek, \mu')$ or $c_1 \neq \mathsf{F}(\mu'[, ek])$
3 : $\overline{dk} := (dk, ek, s)$	3 : $c_1 := \mathsf{F}(\mu[, ek])$	3 : **then return** $K := \mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$
4 : **return** $(ek, \overline{dk})$	4 : $K := \mathsf{H}(\mu, c_0, c_1)$	4 : **else return** $K := \mathsf{H}(\mu', c_0, c_1)$
	5 : **return** $((c_0, c_1), K)$	

Fig. 16. $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$

I.1 SPR-CCA Security

In order to show the SPR-CCA security of $\mathsf{HU}^{\not\perp,\mathsf{prf}}$, we first show the following theorem for indifferentiable reduction, which is obtained by mimicking that for $\mathsf{U}^{\not\perp}_m \hookrightarrow \mathsf{U}^{\not\perp,\mathsf{prf}}$ in [BHH+19, Theorem 5].

Theorem I.1 ($\mathsf{HU}^{\not\perp}_m \hookrightarrow \mathsf{HU}^{\not\perp,\mathsf{prf}}$:). *Let* PKE *be a deterministic PKE. Let* $\mathsf{KEM}_m = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}_m, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *and* $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$.

1. *If* KEM_m *is* SPR-CCA*-secure, then* KEM *is* SPR-CCA*-secure also.*
2. *If* KEM *is* SPR-CCA*-secure, then* KEM_m *is* SPR-CCA*-secure also.*

Since the proof is the same as that of Theorem G.1, we omit it.
We then apply the above theorem to Theorem H.2 and Theorem H.3 and obtain the following theorems:

Theorem I.2 (Case of derandomized PKE). *Let* $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable with simulator* $\mathcal{S}$*, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SPR-CCA*-secure, where we use the new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$.

Theorem I.3 (Case of non-derandomized PKE). *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SPR-CCA*-secure, where we use the new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$.

I.2 SSMT-CCA Security

Theorem I.4. *Suppose that a ciphertext space* $\mathcal{C}$ *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SSMT-CCA*-secure.*
Formally speaking, for any $\mathcal{A}$*, we have*

$$\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) \leq 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}_{\mathsf{prf}}} + q_{\mathrm{DEC}}) \cdot 2^{-\ell/2}.$$

Since the security proof is the same as that for $\mathsf{HU}^{\not\perp}_m$ (Theorem H.4), we omit it.

I.3 SCFR-CCA Security

Theorem I.5. *If* PKE *is* SCFR-CCA*-secure (or* XCFR*-secure), then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *is* SCFR-CCA*-secure in the quantum random oracle model.*

Theorem I.6. *Let* $\mathsf{Col}_{\mathsf{Gen}}$ *be the event that when generating two keys* $(ek_i, dk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ *for* $i \in \{0, 1\}$*, they collide, that is,* $ek_0 = ek_1$. *If* $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ *is negligible, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *with* $c_1 = \mathsf{F}(\mu, ek)$ *is* SCFR-CCA*-secure in the quantum random oracle model.*

The security proofs are the same as those for $\mathsf{HU}^{\not\perp}_m$ (Theorem H.5 and Theorem H.6) and we omit them.

J Properties of $\mathsf{HU}^{\not\perp}$

Finally, we consider another variant of HU with implicit rejection, $\mathsf{HU}^{\not\perp}$. Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a deterministic PKE scheme whose plaintext space is $\mathcal{M}$. Let $\mathcal{C}$ and $\mathcal{K}$ be a ciphertext and key space. Let $\mathcal{H}$ be a some finite space. Let $\mathsf{H} \colon \mathcal{M} \times \mathcal{C} \times \mathcal{H} \to \mathcal{K}$ and $\mathsf{F} \colon \mathcal{M} \to \mathcal{H}$ be hash functions modeled by random oracles. $\mathsf{KEM} = (\overline{\mathsf{Gen}}, \overline{\mathsf{Enc}}, \overline{\mathsf{Dec}}) = \mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ is defined as in Figure 11.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek, s)$
1 : $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	1 : $\mu \leftarrow \mathcal{M}$	1 : $\mu' \leftarrow \mathsf{Dec}(dk, c)$
2 : $s \leftarrow \mathcal{M}$	2 : $c_0 := \mathsf{Enc}(ek, \mu)$	2 : **if** $\mu' = \perp$ or $c_0 \neq \mathsf{Enc}(ek, \mu')$ or $c_1 \neq \mathsf{F}(\mu'[, ek])$
3 : $\overline{dk} := (dk, ek, s)$	3 : $c_1 := \mathsf{F}(\mu[, ek])$	3 : **then return** $K := \mathsf{H}(s, c_0, c_1)$
4 : **return** $(ek, \overline{dk})$	4 : $K := \mathsf{H}(\mu, c_0, c_1)$	4 : **else return** $K := \mathsf{H}(\mu', c_0, c_1)$
	5 : **return** $((c_0, c_1), K)$	

Fig. 17. $\mathsf{KEM} = \mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$

Table 11. Summary of Games for the Proof of Theorem J.4: '$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$' implies that the challenger generates $c_0^* \leftarrow \mathcal{S}(1^\kappa)$, $c_1^* \leftarrow \mathcal{H}$ and returns $\perp$ if $c_0^* \in \mathsf{Enc}(ek, \mathcal{M})$.

Game	H	F	c_0^*	c_1^*	K^*	Decryption valid (c_0, c_1)	Decryption invalid (c_0, c_1)	justification
Game_0	H	F	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu, c_0, c_1)$	$\mathsf{H}(s, c_0, c_1)$	
Game_1	H	F	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu, c_0, c_1)$	$\mathsf{H}_q(c_0, c_1)$	Lemma 2.2
Game_2	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$U(\mathcal{K})$	$\mathsf{H}(\mu, c_0, c_1)$	$\mathsf{H}_q(c_0, c_1)$	statistical disjointness
Game_3	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$\mathsf{H}_q(c_0^*, c_1^*)$	$\mathsf{H}(\mu, c_0, c_1)$	$\mathsf{H}_q(c_0, c_1)$	$\mathsf{H}_q(c_0^*, c_1^*)$ is hidden
Game_4	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$\mathsf{H}(s, c_0^*, c_1^*)$	$\mathsf{H}(\mu, c_0, c_1)$	$\mathsf{H}(s, c_0, c_1)$	Lemma 2.2
Game_5	H	F	$\mathcal{S}(1^\kappa) \setminus \mathsf{Enc}(ek, \mathcal{M})$	$U(\mathcal{H})$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0^*, c_1^*))$	$\mathsf{H}(\mu, c_0, c_1)$	$\mathsf{H}(s, c_0, c_1)$	re-encryption check
Game_6	H	F	$\mathcal{S}(1^\kappa)$	$U(\mathcal{H})$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0^*, c_1^*))$	$\mathsf{H}(\mu, c_0, c_1)$	$\mathsf{H}(s, c_0, c_1)$	statistical disjointness

J.1 SPR-CCA security

IN order to show the SPR-CCA security of $\mathsf{HU}^{\not\perp}$, we use the following theorem, an adapted version of [BHH+19, Theorem 3].

Theorem J.1 ($\mathsf{HU}^{\perp} \to \mathsf{HU}^{\not\perp}$). *Let* PKE *be a deterministic PKE. Let* $\mathsf{KEM}^{\perp} = \mathsf{HU}^{\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *and* $\mathsf{KEM}^{\not\perp} = \mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$. *If* $\mathsf{KEM}^{\perp}$ *is* SPR-CCA*-secure, then* $\mathsf{KEM}^{\not\perp}$ *is also* SPR-CCA*-secure.*

Proof. Suppose that we have an adversary $\mathcal{A}$ against the SPR-CCA security of $\mathsf{KEM}^{\not\perp}$. We construct an adversary $\mathcal{A}'$ against the SPR-CCA security of $\mathsf{KEM}^{\perp}$ as follows: Given an encapsulation key ek, a target ciphertext (c_0^*, c_1^*), and a key K_b^*, $\mathcal{A}'$ samples a fresh seed $s \leftarrow \mathcal{M}$. It runs $\mathcal{A}$ on input ek, (c_0^*, c_1^*), and K_b^*. If $\mathcal{A}$ queries a ciphertext (c_0, c_1) to the decapsulation oracle, then $\mathcal{A}'$ queries the ciphertext (c_0, c_1) and receives K. If $K \neq \perp$, then it returns K to $\mathcal{A}$; Otherwise, it queries (s, c_0, c_1) to the random oracle H, receives $\tilde{K}$, and returns $\tilde{K}$ to $\mathcal{A}$. If $\mathcal{A}$ outputs b' and halts, then $\mathcal{A}'$ also outputs b' and halts.
This simulation is clearly perfect and the theorem follows. □

Applying the above theorem to Theorem G.2 and Theorem G.3, we obtain the following theorems:

Theorem J.2 (Case of derandomized PKE). *Let* $\mathsf{PKE} = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$. *Suppose that a ciphertext space* C *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable with simulator* $\mathcal{S}$*, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SPR-CCA*-secure, where we use the new simulator* $\mathcal{S}' = \mathcal{S} \times U(\mathcal{H})$.

Theorem J.3 (Case of non-derandomized PKE). *Suppose that a ciphertext space* C *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{HU}^{\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SPR-CCA*-secure.*

J.2 SSMT-CCA Security

Theorem J.4. *Suppose that a ciphertext space* C *of* PKE *depends on the public parameter only. If* PKE *is strongly disjoint-simulatable, then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SSMT-CCA*-secure.*
Formally speaking, for any $\mathcal{A}$*, we have*

$$\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM}, \mathcal{A}}(\kappa) \leq 2\mathsf{Disj}_{\mathsf{PKE}, \mathcal{S}}(\kappa) + 4(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|}.$$

The security proof is essentially same as that for SXY (Theorem 4.3).

Game_0: This game is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 0$. The challenge is generated as

$$(c_0^*, c_1^*, K_0^*) \leftarrow \mathcal{S}(1^\kappa) \times U(\mathcal{H}) \times \mathcal{K}.$$

We have

$$\Pr[S_0] = 1 - \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 0].$$

Game_1: This game is the same as Game_0 except that $\mathsf{H}(s, c_0, c_1)$ in the decapsulation oracle is replace with $\mathsf{H}_q(c_0, c_1)$ where $\mathsf{H}_q : \mathcal{C} \times \mathcal{H} \to \mathcal{K}$ is another random oracle. As in[JZC+18, Theorem 1] and [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_1] - \Pr[S_2]| \leq 2(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|},$$

where q_{H} denote the number of queries to H the adversary makes.

Game_2: In this game, the ciphertext is set as $\perp$ if c_0^* is in $\mathsf{Enc}(ek, \mathcal{M})$.
The difference between two games Game_1 and Game_2 is bounded by statistical disjointness.

$$|\Pr[S_1] - \Pr[S_2]| \leq \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Game_3: This game is the same as Game_2 except that $K^* := \mathsf{H}_q(c_0^*, c_1^*)$ instead of chosen random. Since c_0^* is always outside of $\mathsf{Enc}(ek, \mathcal{M})$, $\mathcal{A}$ cannot obtain any information about $\mathsf{H}_q(c_0^*, c_1^*)$ via the decapsulation oracle. Hence, the two games Game_2 and Game_3 are equivalent and we have

$$\Pr[S_2] = \Pr[S_3].$$

Game_4: This game is the same as Game_3 except that $\mathsf{H}_q(\cdot, \cdot)$ is replaced by $\mathsf{H}_{\mathsf{prf}}(s, \cdot, \cdot)$. As in [JZC+18, Theorem 1] and [XY19, Lemmas 4.1], from Lemma 2.2 we have the bound

$$|\Pr[S_3] - \Pr[S_4]| \leq 2(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|},$$

Game_5: This game is the same as Game_4 except that $K^* := \overline{\mathsf{Dec}}(\overline{dk}, (c_0^*, c_1^*))$ instead of $\mathsf{H}(s, c_0^*, c_1^*)$. Recall that c_0^* is always in *outside* of $\mathsf{Enc}(ek, \mathcal{M})$. Thus, we always have $\mathsf{Dec}(c_0^*) = \perp$ or $\mathsf{Enc}(ek, \mathsf{Dec}(c_0^*)) \neq c_0^*$ and, thus, $K^* = \mathsf{H}(s, c_0^*, c_1^*)$. Hence, the two games are equivalent. We have

$$\Pr[S_4] = \Pr[S_5].$$

Game_6: We finally replace how to compute (c_0^*, c_1^*). In this game, the ciphertext is chosen by $\mathcal{S}(1^\kappa) \times U(\mathcal{H})$ as in Game_0.
The difference between two games Game_5 and Game_6 is bounded by statistical disjointness.

$$|\Pr[S_5] - \Pr[S_6]| \leq \mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa).$$

Moreover, this game Game_6 is the original game $\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ with $b = 1$.

$$\Pr[S_6] = \Pr[\mathsf{Expt}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1 \mid b = 1].$$

Summing the (in)equalities, we obtain Theorem J.4:

$$\begin{aligned}\mathsf{Adv}^{\text{ssmt-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) &= |\Pr[S_0] - \Pr[S_6]| \\ &\leq 2\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) + 4(q_{\mathsf{H}} + q_{\mathrm{Dec}})/\sqrt{|\mathcal{M}|}.\end{aligned}$$

J.3 SCFR-CCA Security

Theorem J.5. *If* PKE *is* SCFR-CCA*-secure (or* XCFR*-secure) then* $\mathsf{KEM} = \mathsf{HU}^{\not\perp}_m[\mathsf{PKE}, \mathsf{H}, \mathsf{F}]$ *is* SCFR-CCA*-secure in the quantum random oracle model.*

Proof. Suppose that an adversary outputs a ciphertext $c = (c_0, c_1)$ which is decapsulated into $K \neq \perp$ by $\overline{dk}_0$ and $\overline{dk}_1$, that is, $\overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c)$. Let us define $\mu_i' = \mathsf{Dec}(dk_i, c_0)$ for $i \in \{0, 1\}$. We also define $\mu_i = \mu_i'$ if $c_0 = \mathsf{Enc}(ek_i, \mu_i')$ and $c_1 = \mathsf{F}(\mu_i')$, and $\perp$ otherwise.
We have five cases defined as follows:

1. Case 1 ($\mu_0 = \mu_1 \neq \perp$): This violates the SCFR-CCA security (or the XCFR security) of the underlying PKE.

2. Case 2 ($\bot \neq \mu_0 \neq \mu_1 \neq \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0, c_0, c_1) = \mathsf{H}(\mu_1, c_0, c_1)$ and we succeed to find a collision for H and F, which is negligible for any QPT adversary (Lemma 2.3).
3. Case 3 ($\mu_0 = \bot$ and $\mu_1 \neq \bot$): In this case, the decapsulation algorithms output $K = \mathsf{H}(s_0, c_0, c_1)$ and $\mathsf{H}(\mu_1, c_0, c_1)$. As in the proof of Theorem E.3, we can replace $\mathsf{H}(s_0, \cdot, \cdot)$ with $\mathsf{H}_q(\cdot, \cdot)$ by introducing negligible error (Lemma 2.2). After that, we find a claw $((c_0, c_1), (\mu_1, c_0, c_1))$ between H_q and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
4. Case 4 ($\mu_0 \neq \bot$ and $\mu_1 = \bot$): In this case, the decapsulation algorithms output $K = \mathsf{H}(\mu_0, c_0, c_1) = \mathsf{H}(s_1, c_0, c_1)$. This follows as Case 3.
5. Case 5 (The other cases): In this case, the decapsulation algorithms output $K = \mathsf{H}(s_0, c_0, c_1) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a collision $((s_0, c_0, c_1), (s_1, c_0, c_1))$ of H if $s_0 \neq s_1$, which occurs with overwhelming probability $1 - 1/|\mathcal{M}|$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

We conclude that the advantage of the adversary is negligible in any cases. □

If we add *ek* to F's input, we can reduce the assumption on PKE.

Theorem J.6. *Let* $\mathsf{Col}_{\mathsf{Gen}}$ *be the event that when generating two keys* $(ek_i, dk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ *for* $i \in \{0, 1\}$*, they collide, that is,* $ek_0 = ek_1$*. If* $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ *is negligible, then* $\mathsf{KEM} = \mathsf{HU}^{\not\bot}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ *with* $c_1 = \mathsf{F}(\mu, ek)$ *is* SCFR-CCA*-secure in the quantum random oracle model.*

Note that this security proof is irrelevant to PKE is deterministic PKE or one derandomized by T.

Proof. Suppose that an adversary outputs a ciphertext $c = (c_0, c_1)$ which is decapsulated into $K \neq \bot$ by $\overline{dk}_0$ and $\overline{dk}_1$, that is, $\overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c)$. Let us define $\mu'_i = \mathsf{Dec}(dk_i, c_0)$ for $i \in \{0, 1\}$. We also define $\mu_i = \mu'_i$ if $c_0 = \mathsf{Enc}(ek_i, \mu'_i)$ and $c_1 = \mathsf{F}(\mu'_i, ek_i)$, and $\bot$ otherwise.
We consider six cases defined as follows:

1. Case 1-1 ($\mu_0 = \mu_1 \neq \bot$ and $ek_0 = ek_1$): This case rarely occurs since $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ is negligible.
2. Case 1-2 ($\mu_0 = \mu_1 \neq \bot$ and $ek_0 \neq ek_1$): In this case, we have $d = \mathsf{F}(\mu'_0, ek_0) = \mathsf{F}(\mu'_1, ek_1)$ with $(\mu'_0, ek_0) \neq (\mu'_1, ek_1)$ and we succeed to find a collision for F, which is negligible for any QPT adversary (Lemma 2.3).
3. Case 2 ($\bot \neq \mu_0 \neq \mu_1 \neq \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0, c_0, c_1) = \mathsf{H}(\mu_1, c_0, c_1)$ and we succeed to find a collision for H and F, which is negligible for any QPT adversary (Lemma 2.3).
4. Case 3 ($\mu_0 = \bot$ and $\mu_1 \neq \bot$): In this case, the decapsulation algorithms output $K = \mathsf{H}(s_0, c_0, c_1)$ and $\mathsf{H}(\mu_1, c_0, c_1)$. As in the proof of Theorem E.3, we can replace $\mathsf{H}(s_0, \cdot, \cdot)$ with $\mathsf{H}_q(\cdot, \cdot)$ by introducing negligible error (Lemma 2.2). After that, we find a claw $((c_0, c_1), (\mu_1, c_0, c_1))$ between H_q and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
5. Case 4 ($\mu_0 \neq \bot$ and $\mu_1 = \bot$): In this case, the decapsulation algorithms output $K = \mathsf{H}(\mu_0, c_0, c_1) = \mathsf{H}(s_1, c_0, c_1)$. This follows as Case 3.
6. Case 5 (The other cases): In this case, the decapsulation algorithms output $K = \mathsf{H}(s_0, c_0, c_1) = \mathsf{H}(s_1, c_0, c_1)$ and we find a collision $((s_0, c_0, c_1), (s_1, c_0, c_1))$ of H if $s_0 \neq s_1$, which occurs with overwhelming probability $1 - 1/|\mathcal{M}|$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

We conclude that the advantage of the adversary is negligible in any cases. □

K Classic McEliece

We briefly review Classic McEliece [ABC+20] in subsection K.1, discuss the security properties of the underlying DPKE, CM-DPKE, in subsection K.2, and discuss the security properties of Classic McEliece in subsection K.3. We want to show that, under appropriate assumptions, Classic McEliece is ANON-CCA-secure in the QROM, and Classic McEliece leads to ANON-CCA-secure hybrid PKE in the QROM. (Unfortunately, Classic McEliece is not collision-free [GMP21a].) In order to do so, we show that the underlying CM-DPKE of Classic McEliece is strongly disjoint-simulatable under appropriate assumptions in subsection K.2. Since Classic McEliece is obtained by applying $\mathsf{HU}^{\not\bot,\mathsf{prf}}$ to CM-DPKE, this strong disjoint-simulatablility implies that Classic McEliece is SPR-CCA-secure and SSMT-CCA-secure in the QROM under those assumptions. Those three properties lead to the anonymity of Classic McEliece and hybrid PKE in the QROM as we wanted. We also discuss a modification of Classic McEliece in order to salvage collision-freeness.

Table 12. Parameter sets of Classic McEliece in Round 3. Note that $q = 2^m$ and $k = n - mt$. (We omit the semi-systematic forms.)

parameter sets	m	n	t	k
kem/mceliece348864	12	3488	64	2720
kem/mceliece460896	13	4608	96	3360
kem/mceliece6688128	13	6688	128	5024
kem/mceliece6960119	13	6960	119	5413
kem/mceliece8192128	13	8192	128	6528

K.1 Review of Classic McEliece

Classic McEliece [ABC+20] is a KEM scheme based on the Niederreiter PKE, in which a public key is a scrambled parity-check matrix, a plaintext is an error vector, and a ciphertext is a syndrome. See Table 12 for concrete parameter values (we omit semi-systematic ones).

Let m, n, t, k, q be positive integers with $q = 2^m$ and $k = n - mt$. Define $\mathcal{S} = \{e \in \mathbb{F}_2^n : \mathsf{HW}(e) = t\}$, which is a plaintext space. Let I_{n-k} be the identity matrix of dimension $n - k$. The underlying deterministic PKE of Classic McEliece, which we call CM-DPKE, is summarized as follows, where we only consider the systematic form and omit the details for the semi-systematic form:

- $\mathsf{Gen}(1^\kappa)$: Choose a monic irreducible polynomial g in $\mathbb{F}_q[x]$ of degree t and distinct $\alpha_1, \ldots, \alpha_n \leftarrow \mathbb{F}_q$. Compute a parity-check matrix $\hat{H} \in \mathbb{F}_2^{n \times k}$ of the Goppa code generated by g and $\alpha_1, \ldots, \alpha_n$. Reduce $\hat{H}$ to systematic form $[I_{n-k} \mid T]$. (If this fails, return $\perp$). Output $ek := T \in \mathbb{F}_2^{(n-k)\times k}$ and $dk := (T, \Gamma)$, where $\Gamma := (g, \alpha_1, \ldots, \alpha_n)$. We note that, using Γ, one can correct an error up to t, because the minimum distance of the Goppa code is at least $2t + 1$ by design.
- $\mathsf{Enc}(ek, e \in \mathcal{S})$: Define $H := [I_{n-k} \mid T] \in \mathbb{F}_2^{(n-k)\times n}$. Compute $c := He \in \mathbb{F}_2^{n-k}$. Output c.
- $\mathsf{Dec}(dk, c)$: Extend c to $v := (c, 0, \ldots, 0) \in \mathbb{F}_2^n$. Find the unique codeword $\tilde{c}$ in the Goppa code defined by Γ that satisfies $\mathsf{HW}(\tilde{c} - v) \le t$. Set $e := v + \tilde{c}$. If $\mathsf{HW}(e) = t$ and $c = He$, then return e. Otherwise, return $\perp$.

Classic McEliece applies $\mathsf{HU}^{\not\perp,\mathsf{prf}}$ to CM-DPKE, where $\mathsf{H}(\mu, c_0, c_1) = \mathsf{SHAKE256}_{256}(\texttt{0x01}, \mu \| c_0 \| c_1)$, $\mathsf{H}_{\mathsf{prf}}(s, c_0, c_1) = \mathsf{SHAKE256}_{256}(\texttt{0x00}, s \| c_0 \| c_1)$, and $\mathsf{F}(e) = \mathsf{SHAKE256}_{256}(\texttt{0x02}, e)$, and is defined in Figure 18.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk} = (dk, s), (c_0, c_1))$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	$e \leftarrow \mathsf{FixedWeight}()$	$e := \mathsf{Dec}(dk, c_0)$
$s \leftarrow \mathbb{F}_2^n$	$c_0 := \mathsf{Enc}(ek, e)$	**if** $e = \perp \vee c_1 \neq \mathsf{F}(e)$,
$ek := T, \overline{dk} := (dk, s)$	$c_1 := \mathsf{F}(e)$	**then return** $K := \mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$
return $(ek, \overline{dk})$	$K := \mathsf{H}(e, c_0, c_1)$	**else return** $K := \mathsf{H}(e, c_0, c_1)$
	return $((c_0, c_1), K)$	

Fig. 18. Classic McEliece

K.2 Properties of CM-DPKE

It is known that the Niederreiter PKE is pseudorandom under appropriate assumptions. In order to adapt the argument, we use the following assumptions:

Definition K.1. *Fix the parameter set. We define a random key-generation algorithm* $\mathsf{RandGen}(pp)$ *as follows: Choose* $\hat{H} \leftarrow U(\mathbb{F}_2^{n \times k})$*, reduce* $\hat{H}$ *to systematic form* $[I_{n-k} \mid \hat{T}]$ *(if this fails, resample), and output* $\hat{T} \in \mathbb{F}_2^{(n-k)\times k}$

- *The modified PR-Key assumption: It is computationally hard to distinguish* T *and* $\hat{T}$*, where* $(T, sk) \leftarrow \mathsf{Gen}(1^\kappa)$ *and* $\hat{T} \leftarrow \mathsf{RandGen}(pp)$.
- *The modified Decisional Syndrome Decoding assumption: It is computationally hard to distinguish* $(\hat{T}, [I_{n-k} \mid \hat{T}] \cdot e)$ *from* $(\hat{T}, u)$ *with* $\hat{T} \leftarrow \mathsf{RandGen}(pp)$, $e \leftarrow \mathsf{FixedWeight}()$, *and* $u \leftarrow U(\mathbb{F}_2^{n-k})$.

Security: Assuming the modified PR-Key assumption and the modified Decisional Syndrome Decoding assumption, it is easy to show that CM-DPKE is ciphertext-indistinguishable in the sense of disjoint simulatability as the case of NTRU-DPKE. Since $2^n = |\mathbb{F}_2^n| \gg \binom{n}{t} = |\mathcal{S}| \geq |\mathsf{Enc}(ek, \mathcal{M})|$, it has statistical disjointness. Thus, CM-DPKE is strongly disjoint-simulatable.

Lemma K.1. *Suppose that the modified PR-key assumption and the modified Decisional Syndrome Decoding assumption hold. Then, CM-DPKE is strongly disjoint-simulatable with a simulator $\mathcal{S}$ that outputs a random vector chosen from $\mathbb{F}_2^{n-k}$.*

CM-DPKE is not collision-free: Let $e_{\mathsf{fixed}} := (1^t, 0^{n-t})$ and $c_{\mathsf{fixed}} := (1^t, 0^{n-k-t})$. We have $t \leq mt = n - k$ for all parameter sets of Classic McEliece. Grubbs et al. observed that for any public key T, c_{fixed} is a valid ciphertext of plaintext e_{fixed} since $H \cdot e_{\mathsf{fixed}} = [I_{n-k} \mid T] \cdot e_{\mathsf{fixed}} = e_{\mathsf{fixed}} = c_{\mathsf{fixed}}$. Hence, CM-DPKE and Classic McEliece is not collision free.

K.3 Properties of Classic McEliece

Combining CM-DPKE's strong disjoint-simulatability with previous theorems on $\mathsf{HU}^{\not\perp,\mathsf{prf}}$, we obtain the following theorems.

Theorem K.1. *Suppose that the modified PR-key assumption and the modified Decisional Syndrome Decoding assumption hold. Then, Classic McEliece is* SPR-CCA*-secure and* SSMT-CCA*-secure in the QROM.*

Proof. Under the modified PR-key assumption and the modified Decisional Syndrome Decoding assumption, NTRU-DPKE is strongly disjoint-simulatable (Lemma K.1). In addition, CM-DPKE is perfectly correct. Applying Theorem I.3 and Theorem I.4, we obtain the theorem. □

Theorem K.2. *Suppose that the modified PR-key assumption and the modified Decisional Syndrome Decoding assumption hold. Classic McEliece is* ANON-CCA*-secure in the QROM.*

Proof. Due to Theorem K.1, under the modified PR-key assumption and the modified Decisional Syndrome Decoding assumption, Classic McEliece is SPR-CCA-secure in the QROM. Thus, applying Theorem 2.5, we have that, under those assumptions, Classic McEliece is ANON-CCA-secure in the QROM. □

Grubbs et al. [GMP21a] discussed the barrier to show anonymity of hybrid encryption based on Classic McEliece since Classic McEliece is not collision free. We avoid this barrier by using SPR-CCA security.
Persichetti [Per13] proposed 'hybrid Niederreiter' and showed its IND-CCA security in the ROM. He also insisted his hybrid Niederreiter is IK-CCA in the ROM if the hybrid Niederreiter is IND-CCA, but his proof is incorrect. (We cannot show $\Pr[\mathsf{G}_3 = 1] - 1/2 = 0$, where G_3 is the game that the adversary get $\psi^* = \mathsf{Enc}(ek_b, \phi')$ with random ϕ' instead of $\psi^* = \mathsf{Enc}(ek_b, \phi)$ with plaintext ϕ chosen by the adversary.)

Salvaging collision-freeness of Classic McEliece: Grubbs et al. [GMP21a, Section 5.1] suggested a variant of HU with implicit rejection, in which F takes as input μ plus ek, but they did not recommend it since $ek = T$ of Classic McEliece is relatively large. (We can show its security as Theorem J.6.) Instead, we can use a variant of HU with implicit rejection, in which F takes as input μ plus $\mathsf{Hash}(ek)$. We can show its strong collision-freeness assuming that the probability that two independent encryption keys collide is negligible.

Theorem K.3. *Let $\mathsf{Col}_{\mathsf{Gen}}$ be the event that when generating two keys $(ek_i, dk_i) \leftarrow \mathsf{Gen}(1^\kappa)$ for $i \in \{0,1\}$, they collide, that is, $ek_0 = ek_1$. If $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ is negligible, then the modified Classic McEliece is* SCFR-CCA*-secure in the QROM.*

Proof. Suppose that an adversary outputs a ciphertext $c = (c_0, c_1)$ which is decapsulated into $K \neq \perp$ by $\overline{dk}_0$ and $\overline{dk}_1$, that is, $\overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c)$. Let us define $e_i' = \mathsf{Dec}(dk_i, c_0)$ for $i \in \{0,1\}$. We also define $e_i = e_i'$ if $c_0 = \mathsf{Enc}(ek_i, e_i')$ and $c_1 = \mathsf{F}(e_i', \mathsf{Hash}(ek_i))$, and $\perp$ otherwise.
We consider seven cases defined as follows:

1. Case 1-1 ($e_0 = e_1 \neq \perp$ and $ek_0 = ek_1$): This case rarely occurs since $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ is negligible.
2. Case 1-2 ($e_0 = e_1 \neq \perp$, $ek_0 \neq ek_1$, and $\mathsf{Hash}(ek_0) = \mathsf{Hash}(ek_1)$): In this case, we have $\mathsf{Hash}(ek_0) = \mathsf{Hash}(ek_1)$ with $ek_0 \neq ek_1$ and we succeed to find a collision for Hash, which is negligible for any QPT adversary (Lemma 2.3).
3. Case 1-3 ($e_0 = e_1 \neq \perp$, $ek_0 \neq ek_1$, and $\mathsf{Hash}(ek_0) \neq \mathsf{Hash}(ek_1)$): In this case, we have $d = \mathsf{F}(e_0, \mathsf{Hash}(ek_0)) = \mathsf{F}(e_1, \mathsf{Hash}(ek_1))$ with $(e_0, \mathsf{Hash}(ek_0)) \neq (e_1, \mathsf{Hash}(ek_1))$ and we succeed to find a collision for F, which is negligible for any QPT adversary (Lemma 2.3).
4. Case 2 ($\perp \neq e_0 \neq e_1 \neq \perp$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(e_0) = \mathsf{H}(e_1)$ and we succeed to find a collision for H, which is negligible for any QPT adversary (Lemma 2.3).

5. Case 3 ($e_0 = \perp$ and $e_1 \neq \perp$): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1)$ and $\mathsf{H}(e_1, c_0, c_1)$ and we find a claw $((s_0, c_0, c_1), (e_1, c_0, c_1))$ of $\mathsf{H}_{\mathsf{prf}}$ and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
6. Case 4 ($e_0 \neq \perp$ and $e_1 = \perp$): In this case, the decapsulation algorithms output $K = \mathsf{H}(e_0, c_0, c_1) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a claw $((e_0, c_0, c_1), (s_1, c_0, c_1))$ of H and $\mathsf{H}_{\mathsf{prf}}$. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
7. Case 5 (The other cases): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a collision $((s_0, c_0, c_1), (s_1, c_0, c_1))$ of $\mathsf{H}_{\mathsf{prf}}$ if $s_0 \neq s_1$, which occurs with probability at least $1 - 1/2^n$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

Thus, we conclude that the advantage of the adversary is negligible. □

Theorem K.4. *Under the modified PR-key assumption and the modified Decisional Syndrome Decoding assumption, the modified Classic McEliece leads to* ANON-CCA*-secure and* SROB-CCA*-secure hybrid PKE in the QROM, combined with* SPR-OTCCA*-secure and* FROB*-secure DEM.*

Proof. By using the similar proof of Theorem K.1, under the modified PR-key assumption and the modified Decisional Syndrome Decoding assumption, the modified Classic McEliece is SPR-CCA-secure and SSMT-CCA-secure in the QROM. In addition, the modified Classic McEliece is perfectly correct. Thus, combining the modified Classic McEliece with SPR-OTCCA-secure DEM, we obtain a SPR-CCA-secure hybrid PKE in the QROM (Theorem 3.2).
Moreover, the modified Classic McEliece is SCFR-CCA-secure in the QROM (Theorem K.3). Thus, if DEM is FROB-secure, then the hybrid PKE is SROB-CCA-secure (Theorem 2.2). □

L Kyber

Review of Kyber in Round 3: Kyber [SAB+20] is a KEM scheme based on the Module LWE problem. We briefly review Kyber.
The underlying PKE scheme of Kyber, which we call Kyber-PKE, is summarized as follows:

- $\mathsf{Gen}(pp)$: The key generation algorithm outputs ek and dk.
- $\mathsf{Enc}(ek, \mu; \rho)$: The encryption algorithm is probabilistic. Taking $\mu \in \{0,1\}^{256}$, it outputs c.
- $\mathsf{Dec}(dk, c)$: The decryption algorithm is deterministic and outputs $\mu' \in \{0,1\}^{256}$.

We next consider an intermediate PKE scheme $\mathsf{PKE}_0 = (\mathsf{Gen}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ where the encryption algorithm uses pseudorandomness, which we call Kyber-PKE-PRG:

- $\mathsf{Gen}_0(pp) = \mathsf{Gen}(pp)$:
- $\mathsf{Enc}_0(ek, \mu; r)$: It uses $\rho_i = \mathsf{SHAKE256}_X(r, i)$ for $i = 0, 1, \ldots$ to sample randomness ρ of $\mathsf{Enc}(ek, \mu)$. It then outputs $c := \mathsf{Enc}(ek, \mu; \rho)$.
- $\mathsf{Dec}_0(dk, c) = \mathsf{Dec}(dk, c)$:

Kyber applies a variant of the FO transform with implicit rejection, denoted by $\mathsf{FO}^{\not\perp\prime}$, to Kyber-PKE-PRG, where $\mathsf{H}' = \mathsf{SHA3\text{-}256}$, $\mathsf{G}(\mu, h) = \mathsf{SHA3\text{-}512}$, and $\mathsf{H} = \mathsf{SHAKE256}_X$ with unspecified output bits X, and is defined as in Figure 19.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, c)$, where $\overline{dk} = (dk, ek, h, s)$
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$\mu \leftarrow \{0,1\}^{256}$	$\mu' := \mathsf{Dec}_0(dk, c)$
$h \leftarrow \mathsf{H}'(ek)$	$\mu := \mathsf{H}'(\mu)$	$(\bar{K}', r') := \mathsf{G}(\mu', h)$
$s \leftarrow \{0,1\}^{256}$	$(\bar{K}, r) := \mathsf{G}(\mu, \mathsf{H}'(ek))$	$c' := \mathsf{Enc}_0(ek, \mu'; r')$
$\overline{dk} := (dk, ek, h, s)$	$c := \mathsf{Enc}_0(ek, \mu; r)$	**if** $c \neq c'$, **then return** $K := \mathsf{H}(s, \mathsf{H}'(c))$
return $(ek, \overline{dk})$	$K := \mathsf{H}(\bar{K}, \mathsf{H}'(c))$	**else return** $K := \mathsf{H}(\bar{K}', \mathsf{H}'(c))$
	return (c, K)	

Fig. 19. Kyber

Security: Grubbs et al. [GMP21a] pointed out there are technical barriers. At first, a pre-key $\bar{K}$ and a randomness r is generated by $\mathsf{G}(\mu, \mathsf{H}'(ek))$. We can treat is as $\bar{K} = \mathsf{G}_0(\mu, \mathsf{H}'(ek))$ and $r = \mathsf{G}_1(\mu, \mathsf{H}'(ek))$, where $\mathsf{G}_0(x)$ and $\mathsf{G}_1(x)$ are defined as the first and last 256-bits of $\mathsf{G} = \text{SHA3-512}$. Using this notion, we compute $K = \mathsf{H}(\mathsf{G}_0(\mu, \mathsf{H}'(ek)), \mathsf{H}'(c))$. Grubbs et al. solved the problem on nested random oracles on μ by letting $\mathsf{G}_r(\mu) := \mathsf{G}_0(\mu, \mathsf{H}'(ek)) : \{0,1\}^{256} \to \{0,1\}^{256}$ and simulating G_r by a random polynomial over $\mathrm{GF}(2^{256})$ of degree $2q_{\mathsf{G}} + 1$ as in [TU16, HHK17]. Grubbs et al. succeeded to show its IND-CCA-security if K was computed as $\mathsf{H}(\mathsf{G}_r(\mu), c)$ as in $\mathsf{FO}^{\not\perp\prime\prime}$. Unfortunately, they left showing $\mathsf{FO}^{\not\perp\prime\prime}$'s IND-CCA-security as open problem. We also left it here.

M Saber

Review of Saber: Saber [DKR+20] is a KEM scheme based on the Module LWR problem. We briefly review Saber.

The underlying PKE scheme of Saber, which we call Saber-PKE, is summarized as follows:

- $\mathsf{Gen}(pp)$: The key-generation algorithm outputs ek and dk.
- $\mathsf{Enc}(ek, \mu; \rho)$: The encryption algorithm is probabilistic. Taking $\mu \in \{0,1\}^{256}$, it outputs c.
- $\mathsf{Dec}(dk, c)$: The decryption algorithm is deterministic and outputs $\mu' \in \{0,1\}^{256}$.

We next consider an intermediate PKE scheme $\mathsf{PKE}_0 = (\mathsf{Gen}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ where the encryption algorithm uses pseudorandomness, which we call Saber-PKE-PRG:

- $\mathsf{Gen}_0(pp) = \mathsf{Gen}(pp)$:
- $\mathsf{Enc}_0(ek, \mu; r)$: It uses $\rho = \mathsf{SHAKE128}_X(r)$ to sample randomness ρ of $\mathsf{Enc}(ek, \mu)$. It then outputs $c := \mathsf{Enc}(ek, \mu; \rho)$.
- $\mathsf{Dec}_0(dk, c) = \mathsf{Dec}(dk, c)$:

Saber applies the same variant of the FO transform with implicit rejection as Kyber to Saber-PKE-PRG, where $\mathsf{H}' = \text{SHA3-256}$, $\mathsf{G}(\mu, h) = \text{SHA3-512}$, and $\mathsf{H} = \text{SHA3-256}$, and is defined as in Figure 20.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, c)$, where $\overline{dk} = (dk, ek, h, s)$
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$\mu \leftarrow \{0,1\}^{256}$	$\mu' := \mathsf{Dec}_0(dk, c)$
$h \leftarrow \mathsf{H}'(ek)$	$\mu := \mathsf{H}'(\mu)$	$(\bar{K}', r') := \mathsf{G}(\mu', h)$
$s \leftarrow \{0,1\}^{256}$	$(\bar{K}, r) := \mathsf{G}(\mu, \mathsf{H}'(ek))$	$c' := \mathsf{Enc}_0(ek, \mu'; r')$
$\overline{dk} := (dk, ek, h, s)$	$c := \mathsf{Enc}_0(ek, \mu; r)$	**if** $c \neq c'$, **then return** $K := \mathsf{H}(s, \mathsf{H}'(c))$
return $(ek, \overline{dk})$	$K := \mathsf{H}(\bar{K}, \mathsf{H}'(c))$	**else return** $K := \mathsf{H}(\bar{K}', \mathsf{H}'(c))$
	return (c, K)	

Fig. 20. Saber

Security: Grubbs et al. [GMP21a] wrote Saber uses $\mathsf{FO}^{\not\perp\prime\prime}$ as defined in [DKR+20, Section 2.5]. However, the specification uses $\mathsf{FO}^{\not\perp\prime}$ [DKR+20, Section 8.5]. Thus, Saber lacks the IND-CCA-security proof in the QROM as Kyber. We also left proving the IND-CCA security of Saber in the QROM as an open problem. It might be interesting to study anonymity and robustness in the ROM.

N BIKE

We briefly review BIKE [ABB+20] in subsection N.1, discuss the security properties of the underlying PKE, BIKE-PKE, and its derandomized version, BIKE-DPKE, in subsection N.2, and discuss the security properties of BIKE in subsection N.3. We want to show that, under appropriate assumptions, BIKE is ANON-CCA-secure in the QROM, and BIKE leads to ANON-CCA-secure and SROB-CCA-secure hybrid PKE in the QROM. In order to do so, we show that the underlying BIKE-DPKE of BIKE is strongly disjoint-simulatable under appropriate assumptions and XCFR-secure in subsection N.2. BIKE is obtained by applying $\mathsf{U}^{\not\perp}$ to BIKE-DPKE, and the former implies that BIKE is SPR-CCA-secure and SSMT-CCA-secure in the QROM under those assumptions and the latter implies that BIKE is SCFR-CCA-secure in the QROM. Those three properties lead to the anonymity of BIKE and hybrid PKE in the QROM as we wanted.

N.1 Review of BIKE

BIKE in round 3 [ABB+20] is a KEM scheme based on QC-MDPC [MTSB13], which is a variant of the McEliece PKE upon a code with quasi-cyclic (QC) moderate density parity-check (MDPC) matrix. BIKE can be considered as the Niederreiter PKE scheme upon a code with the QC-MDPC matrix. Let $\mathcal{R} := \mathbb{F}[x]/(x^r - 1)$. Let $\mathcal{H}_w := \{(h_0, h_1) \in \mathcal{R}^2 : \mathsf{HW}(h_0) = \mathsf{HW}(e_1) = w/2\}$. Let $\mathcal{E}_t := \{(e_0, e_1) \in \mathcal{R}^2 : \mathsf{HW}(e_0, e_1) = t\}$. For concrete values of r, w, and t, see Table 13.

Table 13. Parameter sets of BIKE in Round 3.

parameter sets	r	w	t
BIKE-1	12,323	142	134
BIKE-3	24,659	206	199
BIKE-5	40,973	274	264

The underlying CPA-secure PKE scheme of BIKE, which we call BIKE-PKE, is summarized as follows:

- $\mathsf{Gen}_0(pp)$: $dk := (h_0, h_1) \leftarrow \mathcal{H}_w$. Output $ek = h := h_1 \cdot h_0^{-1} \in \mathcal{R}$ and dk.
- $\mathsf{Enc}_0(ek, \mu \in \{0,1\}^{256}; r)$: Sample $(e_0, e_1) \leftarrow \mathcal{E}_t$ by using the randomness r. Compute $u := e_0 + e_1 h \in \mathcal{R}$ and $v := \mu \oplus \mathsf{L}(e_0, e_1)$ and output $c := (u, v)$.
- $\mathsf{Dec}_0(dk, (u, v))$: Compute $(e_0, e_1) \leftarrow \mathsf{decode}(uh_0, (h_0, h_1))$, where decode is a decoder of the QC-MDPC code with parity check matrix generated by h_0 and h_1. Output $\mu' := v \oplus \mathsf{L}(e_0, e_1)$, where $\mathsf{L} = \mathsf{SHA3\text{-}384}_{256}$.

Notice that $uh_0 = e_0 h_0 + e_1 h_1$, which is the syndrome of (e_0, e_1) with the parity-check matrix generated by h_0 and h_1.

BIKE applies a variant of the FO transform with implicit rejection, $\mathsf{FO}^{\not\perp} = \mathsf{U}^{\not\perp} \circ \mathsf{T}$, to BIKE-PKE PKE, where $\mathsf{G} = \mathsf{SHAKE256}$ and $\mathsf{H} = \mathsf{SHA3\text{-}384}_{256}$, and is defined as in Figure 21.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, c)$, where $\overline{dk} = (dk, ek, s)$
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$\mu \leftarrow \{0,1\}^{256}$	$\mu' := \mathsf{Dec}_0(dk, c)$
$s \leftarrow \{0,1\}^{256}$	$r := \mathsf{G}(\mu)$	$r' := \mathsf{G}(\mu')$
$\overline{dk} := (dk, ek, s)$	$c := \mathsf{Enc}_0(ek, \mu; r)$	$c' := \mathsf{Enc}_0(ek, \mu'; r')$
return $(ek, \overline{dk})$	$K := \mathsf{H}(\mu, c)$	**if** $c \neq c'$
	return (c, K)	**then return** $K := \mathsf{H}(s, c)$
		else return $K := \mathsf{H}(\mu', c)$

Fig. 21. BIKE

Recall that $\mathsf{FO}^{\not\perp}$ is $\mathsf{U}^{\not\perp} \circ \mathsf{T}$. In what follows, we first study BIKE-PKE's properties and then study BIKE-DPKE's properties, where BIKE-DPKE is obtained by derandomizing BIKE-PKE with transform T.

N.2 Properties of BIKE-PKE and BIKE-DPKE

Although we can invoke theorems on $\mathsf{FO}^{\not\perp}$ by Grubbs et al. [GMP21a] to show BIKE's anonymity and collision-freeness, we can show BIKE's anonymity through another pass.

Assumptions: For $b \in \{0,1\}$, define the finite set $\mathcal{F}_b := \{h \in \mathcal{R} : \mathsf{HW}(h) \equiv b \pmod 2\}$, that is, a set of all binary vectors of length r and parity b. We suppose that w is even and $w/2$ is odd, which hold for all parameter sets of BIKE.

Definition N.1 (The 2-Decisional Quasi-Cyclic Code-Finding (2-DQCCF) assumption [ABB+20]). *For any (Q)PPT adversary, it is hard to distinguish the following two distributions:*

- $h := h_1 \cdot h_0^{-1}$, *where* $(h_0, h_1) \leftarrow \mathcal{H}_w$.

– $h \leftarrow \mathcal{F}_1$.

Definition N.2 (The 2-Computational Quasi-Cyclic Syndrome Decoding (2-CQCSD) assumption [ABB+20]). *For any (Q)PPT adversary, given $(h, u := he_1 + e_0)$, where $h \leftarrow \mathcal{F}_1$ and $(e_0, e_1) \leftarrow \mathcal{E}_t$, it is hard to find $(e_0', e_1') \in \mathcal{E}_t$ with $u = he_1' + e_0'$.*

Definition N.3 (The 2-Decisional Quasi-Cyclic Syndrome Decoding (2-DQCSD) assumption [ABB+20]). *For any (Q)PPT adversary, it is hard to distinguish the following two distributions:*
– *$(h, u := he_1 + e_0)$, where $h \leftarrow \mathcal{F}_1$ and $(e_0, e_1) \leftarrow \mathcal{E}_t$.*
– *(h, u), where $h \leftarrow \mathcal{F}_1$ and $u \leftarrow \mathcal{F}_{t \bmod 2}$.*

BIKE-Simple: Before showing the security, we consider the following deterministic PKE scheme, which we call BIKE-Simple:
– $\mathsf{Gen}(pp)$: $dk := (h_0, h_1) \leftarrow \mathcal{H}_w$. Output $ek = h := h_1 \cdot h_0^{-1} \in \mathcal{R}$ and dk.
– $\mathsf{Enc}(ek, (e_0, e_1) \in \mathcal{E}_t)$: Compute $u := e_0 + e_1 h \in \mathcal{R}$ and output u.
– $\mathsf{Dec}(dk, u)$: Output $(e_0, e_1) \leftarrow \mathsf{decode}(uh_0, (h_0, h_1))$.
The proposers showed that this scheme is OW-CPA-secure using appropriate assumptions as follows:

Lemma N.1 ([ABB+20, Theorem 1]). *If the 2-DQCCF and 2-CQCSD assumptions hold, then BIKE-Simple is* OW-CPA*-secure.*

Remark N.1. It is easy to show BIKE-Simple's disjoint simulatability: Let $\mathcal{F}_1$ be a ciphertext space. We define the simulator as sampling $u \leftarrow U(\mathcal{F}_1)$. Statistical disjointness follows from the fact that $|\mathcal{F}_1| \approx 2^r/2 \gg \binom{2r}{t} = |\mathcal{E}_t| \geq |\mathsf{Enc}(ek, \mathcal{E}_t)|$. We can show ciphertext indistinguishability by using the 2-DQCCF and 2-DQCSD assumptions as we showed ciphertext indistinguishability of NTRU-DPKE and CM-DPKE.

Remark N.2. Applying SXY and assuming δ is negligible, we can obtain a *tightly* CCA-secure KEM scheme with shorter ciphertext, which leads to anonymous, robust hybrid PKE.

Security of BIKE-PKE: We next show that BIKE-PKE is ciphertext-indistinguishable in the QROM.

Lemma N.2. *Suppose that the 2-DQCCF and 2-DQCSD assumptions hold. Then, BIKE-PKE is ciphertext-indistinguishable in the QROM with a simulator that outputs $u \leftarrow \mathcal{F}_{t \bmod 2}$ and $v \leftarrow \mathbb{F}_2^{256}$.*

Proof (Proof Sketch). We consider four games Game_i for $i = 0, 1, \ldots, 4$ defined as follows:
– Game_0: In this game, an encryption key and a target ciphertext is computed as follows:
  • Key generation: $(h_0, h_1) \leftarrow \mathcal{H}_w$ and $h := h_1 \cdot h_0^{-1}$.
  • Encryption: $\mu \leftarrow \mathbb{F}_2^{256}$, $(e_0, e_1) \leftarrow \mathcal{E}_t$; compute $u := e_0 + he_1$ and $v := \mu \oplus \mathsf{L}(e_0, e_1)$; return $c = (u, v)$.
– Game_1: In this game, an encryption key and a target ciphertext is computed as follows:
  • Key generation: $(h_0, h_1) \leftarrow \mathcal{H}_w$ and $h := h_1 \cdot h_0^{-1}$.
  • Encryption: $(e_0, e_1) \leftarrow \mathcal{E}_t$; compute $u := e_0 + he_1$; $v \leftarrow \mathbb{F}_2^{256}$; return $c = (u, v)$.
– Game_2: In this game, an encryption key and a target ciphertext is computed as follows:
  • Key generation: $h \leftarrow \mathcal{F}_1$.
  • Encryption: $(e_0, e_1) \leftarrow \mathcal{E}_t$; compute $u := e_0 + he_1$; $v \leftarrow \mathbb{F}_2^{256}$; return $c = (u, v)$.
– Game_3: In this game, an encryption key and a target ciphertext is computed as follows:
  • Key generation: $h \leftarrow \mathcal{F}_1$.
  • Encryption: $u \leftarrow \mathcal{F}_{t \bmod 2}$; $v \leftarrow \mathbb{F}_2^{256}$; return $c = (u, v)$.
– Game_4: In this game, an encryption key and a target ciphertext is computed as follows:
  • Key generation: $(h_0, h_1) \leftarrow \mathcal{H}_w$ and $h := h_1 \cdot h_0^{-1}$.
  • Encryption: $u \leftarrow \mathcal{F}_{t \bmod 2}$; $v \leftarrow \mathbb{F}_2^{256}$; return $c = (u, v)$.
Game_0 and Game_1 are equivalent, since μ in Game_0 and v in Game_1 is chosen uniformly at random. Game_1 and Game_2 are computationally indistinguishable under the 2-DQCCF assumption. Game_2 and Game_3 are computationally indistinguishable under the 2-DQCSD assumption. Game_3 and Game_4 are computationally indistinguishable under the 2-DQCCF assumption. Summing up those (in)equalities, we obtain the lemma. □

We next consider BIKE-PKE is IND-CPA-secure in the QROM. The proposers showed the security in the *ROM* as follows:

Lemma N.3 ([ABB+20, Theorem 2]). *If the 2-DQCCF and 2-CQCSD assumptions hold, then BIKE-PKE is* IND-CPA*-secure in the ROM.*

Unfortunately, applying their idea directly to the QROM setting, the security proof becomes loose since it will involve the O2H lemma (Corollary A.1). We here show the IND-CPA security of BIKE-PKE in the QROM *tightly* using the idea of [SXY18].

Lemma N.4. *Assume that the 2-DQCCF and 2-DQCSD assumptions hold and BIKE-PKE is δ-correct with negligible δ. Then, BIKE-PKE is* IND-CPA*-secure (and* OW-CPA*-secure) in the QROM.*

Proof (Proof Sketch). We consider $\mathsf{Game}_{i,b}$ for $b \in \{0,1\}$ and $i = 0,\dots,4$ defined as follows:

- $\mathsf{Game}_{0,b}$: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $(h_0, h_1) \leftarrow \mathcal{H}_w$ and $h := h_1 \cdot h_0^{-1}$.
  - Encryption given μ_0 and μ_1: $(e_0, e_1) \leftarrow \mathcal{E}_t$; compute $u := e_0 + he_1$, $k := \mathsf{L}(e_0, e_1)$, and $v := \mu_b \oplus k$; return $c = (u, v)$.
- $\mathsf{Game}_{1,b}$: In this game, we use another random oracle $\mathsf{L}_q : \mathcal{R} \to \{0,1\}^{256}$ and define $\mathsf{L}(e_0, e_1) = \mathsf{L}_q(he_0 + e_1)$. an encryption key and a target ciphertext is computed as follows:
  - Key generation: $(h_0, h_1) \leftarrow \mathcal{H}_w$ and $h := h_1 \cdot h_0^{-1}$.
  - Encryption given μ_0 and μ_1: $(e_0, e_1) \leftarrow \mathcal{E}_t$; compute $u := e_0 + he_1$, $k := \mathsf{L}_q(u)$, and $v := \mu_b \oplus k$; return $c = (u, v)$.
- $\mathsf{Game}_{2,b}$: In this game, we use random h. An encryption key and a target ciphertext is computed as follows:
  - Key generation: $h \leftarrow \mathcal{F}_1$.
  - Encryption given μ_0 and μ_1: $(e_0, e_1) \leftarrow \mathcal{E}_t$; compute $u := e_0 + he_1$, $k := \mathsf{L}_q(u)$, and $v := \mu_b \oplus k$; return $c = (u, v)$.
- $\mathsf{Game}_{3,b}$: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $h \leftarrow \mathcal{F}_1$.
  - Encryption given μ_0 and μ_1: $u \leftarrow \mathcal{F}_{t \bmod 2}$; compute $k := \mathsf{L}_q(u)$, and $v := \mu_b \oplus k$; return $c = (u, v)$.
- $\mathsf{Game}_{4,b}$: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $h \leftarrow \mathcal{F}_1$.
  - Encryption given μ_0 and μ_1: $u \leftarrow \mathcal{F}_{t \bmod 2}$, $k \leftarrow \{0,1\}^{256}$; compute $v := \mu_b \oplus k$; return $c = (u, v)$.

$\mathsf{Game}_{0,b}$ and $\mathsf{Game}_{1,b}$ are equivalent if the mapping $(e_0, e_1) \mapsto he_0 + e_1$ is injective, which is satisfied if a key pair is accurate. $\mathsf{Game}_{1,b}$ and $\mathsf{Game}_{2,b}$ are computationally indistinguishable under the 2-DQCCF assumption. $\mathsf{Game}_{2,b}$ and $\mathsf{Game}_{3,b}$ are computationally indistinguishable under the 2-DQCSD assumption. $\mathsf{Game}_{3,b}$ and $\mathsf{Game}_{4,b}$ are equivalent if u is in outside of the image of the mapping $(e_0, e_1) \mapsto e_0 + e_1 h$, which occurs with overwhelming probability. $\mathsf{Game}_{4,0}$ and $\mathsf{Game}_{4,1}$ are equivalent since k is uniformly at random. Summing up those (in)equalities, we obtain the lemma. □

Remark N.3. We can replace the term δ with the probability that the mapping $(e_0, e_1) \mapsto e_0 + e_1 h$ is injective for random $h \leftarrow \mathcal{F}_1$.

Security of BIKE-DPKE: We then consider BIKE-DPKE obtained by applying T to BIKE-PKE.

Lemma N.5. *Assume that the 2-DQCCF and 2-DQCSD assumptions hold. Then, BIKE-DPKE is strongly disjoint-simulatable.*

Proof. Statistical disjointess follows from the fact that $|\mathcal{S}(1^\kappa)| \approx 2^r/2 \cdot 2^{256}$ and $|\mathsf{Enc}(ek, \mathcal{M})| \leq 2^{256}$. We can show ciphertext indistinguishability by invoking Theorem D.1 since BIKE-PKE is ciphertext-indistinguishable (Lemma N.2) and oneway (Lemma N.4). □

We next consider BIKE-DPKE's XCFR-security:

Lemma N.6. *Let ϵ_u be a probability that $h_0 - h_1 \notin \mathcal{R}^*$ holds for two randomly generated keys h_0 and h_1. Let ϵ_0 be a probability that an efficient adversary finds μ such that $e_1 = 0$ where $(e_0, e_1) := \mathcal{E}_t(\mathrm{G}(\mu))$. Suppose that and $\epsilon := \epsilon_u + \epsilon_0$ is negligible. Then, BIKE-DPKE is* XCFR*-secure.*

Proof (Proof sketch:). Let us consider $ek_i = h_i$ and $dk_i = (h_0, h_1)$ for $i \in \{0,1\}$. If the adversary outputs $c = (u, v)$, it should be decrypted into μ by using dk_0 and dk_1, respectively. Let $(e_0, e_1) = \mathcal{E}_t(\mathrm{G}(\mu))$. We have $u = e_0 + e_1 h_0 = e_0 + e_1 h_1$ in the re-encryption check. This implies $(h_0 - h_1) \cdot e_1 = 0 \in \mathcal{R}$. If $e_1 \neq 0$ and $h_0 - h_1 \in \mathcal{R}^*$, then this leads a contradiction. Thus, the lemma holds. □

N.3 Properties of BIKE

Combining BIKE-DPKE's strong disjoint-simulatability and XCFR security with previous theorems on $\mathsf{U}^{\not\perp}$, we obtain the following theorems.

Theorem N.1. *Suppose that the 2-DQCCF and 2-DQCSD assumptions hold and BIKE-DPKE is δ-correct with negligible δ. Then, BIKE is* SPR-CCA*-secure and* SSMT-CCA*-secure in the QROM.*

Proof. Under the 2-DQCCF and 2-DQCSD assumptions, BIKE-DPKE is strongly disjoint-simulatable (Lemma N.5). Applying Theorem E.2 and Theorem E.3, we obtain the theorem. □

Theorem N.2. *Let ϵ_u be a probability that $h_0 - h_1 \notin \mathcal{R}^*$ holds for two randomly generated keys h_0 and h_1. Let ϵ_0 be a probability that an efficient adversary finds μ such that $e_1 = 0$ where $(e_0, e_1) := \mathcal{E}_t(\mathsf{G}(\mu))$. Suppose that and $\epsilon := \epsilon_u + \epsilon_0$ is negligible. Then, BIKE is* SCFR-CCA*-secure in the QROM.*

Proof. Under the hypothesis, BIKE-DPKE is XCFR-secure (Lemma N.6). Applying Theorem E.4, we have that BIKE is SCFR-CCA-secure in the QROM. □

Theorem N.3. *Suppose that the 2-DQCCF and 2-DQCSD assumptions hold and BIKE-DPKE is δ-correct with negligible δ. Then, BIKE is* ANON-CCA*-secure in the QROM.*

Proof. Due to Theorem N.1, under the hypothesis, BIKE is SPR-CCA-secure in the QROM. Thus, applying Theorem 2.5, we have that, under those assumptions, BIKE is ANON-CCA-secure in the QROM. □

Theorem N.4. *Let ϵ_u be a probability that $h_0 - h_1 \notin \mathcal{R}^*$ holds for two randomly generated keys h_0 and h_1. Let ϵ_0 be a probability that an efficient adversary finds μ such that $e_1 = 0$ where $(e_0, e_1) := \mathcal{E}_t(\mathsf{G}(\mu))$. Suppose that and $\epsilon := \epsilon_u + \epsilon_0$ is negligible. Suppose that the 2-DQCCF and 2-DQCSD assumptions hold and BIKE-DPKE is δ-correct with negligible δ. Then, BIKE leads to* ANON-CCA*-secure and* SROB-CCA*-secure hybrid PKE in the QROM, combined with* SPR-OTCCA*-secure and* FROB*-secure DEM.*

Proof. Due to Theorem N.1, under the 2-DQCCF and 2-DQCSD assumptions and the assumption on the correctness, BIKE is SPR-CCA-secure and SSMT-CCA-secure in the QROM. Thus, combining BIKE with SPR-OTCCA-secure DEM, we obtain a SPR-CCA-secure hybrid PKE in the QROM (Theorem 3.2). Moreover, BIKE is SCFR-CCA-secure in the QROM (Theorem N.2) under the hypothesis on ϵ. Thus, if DEM is FROB-secure, then the hybrid PKE is SROB-CCA-secure (Theorem 2.2). □

O FrodoKEM

Review of FrodoKEM: FrodoKEM [NAB+20] is an LWE-based KEM scheme in the alternates candidates. The underlying PKE scheme of FrodoKEM, which we call FrodoKEM-PKE, is summarized as follows:

- $\mathsf{Gen}(pp)$: The key-generation algorithm outputs ek and dk.
- $\mathsf{Enc}(ek, \mu; \rho)$: The encryption algorithm is probabilistic. Taking $\mu \in \{0,1\}^k$, it outputs c.
- $\mathsf{Dec}(dk, c)$: The decryption algorithm is deterministic and outputs $\mu' \in \{0,1\}^k$.

We next consider an intermediate PKE scheme $\mathsf{PKE}_0 = (\mathsf{Gen}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ where the encryption algorithm uses pseudorandomness, which we call FrodoKEM-PKE-PRG:

- $\mathsf{Gen}_0(pp) = \mathsf{Gen}(pp)$:
- $\mathsf{Enc}_0(ek, \mu; r)$: It uses $\rho = \mathsf{SHAKE128}_X(\mathtt{0x96} \| r)$ to sample randomness ρ. It then outputs $c := \mathsf{Enc}(ek, \mu; \rho)$.
- $\mathsf{Dec}_0(dk, c) = \mathsf{Dec}(dk, c)$:

FrodoKEM applies a variant of the FO transform with implicit rejection to FrodoKEM-PKE-PRG, where H', G, and H are SHAKE128 or SHAKE256, and is defined as in Figure 22: We can treat them as different random oracles because their input length differ.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, c)$, where $\overline{dk} = (dk, ek, h, s)$
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$\mu \leftarrow \{0,1\}^k$	$\mu' := \mathsf{Dec}_0(dk, c)$
$h \leftarrow \mathsf{H}'(ek)$	$(\bar{K}, r) := \mathsf{G}(\mu, \mathsf{H}'(ek))$	$(\bar{K}', r') := \mathsf{G}(\mu', h)$
$s \leftarrow \{0,1\}^k$	$c := \mathsf{Enc}_0(ek, \mu; r)$	$c' := \mathsf{Enc}_0(ek, \mu'; r')$
$\overline{dk} := (dk, ek, h, s)$	$K := \mathsf{H}(\bar{K}, c)$	**if** $c \neq c'$, **then return** $K := \mathsf{H}(s, c)$
return $(ek, \overline{dk})$	**return** (c, K)	**else return** $K := \mathsf{H}(\bar{K}', c)$

Fig. 22. FrodoKEM

Security: Grubbs et al. [GMP21a] fortunately show the security of the variant of the FO transform. Thus, we can apply their result to FrodoKEM.

P HQC

We briefly review HQC [AAB+20] in subsection P.1, discuss the security properties of the underlying PKE, HQC-PKE, and its derandomized version, HQC-DPKE, in subsection P.2, and discuss the security properties of HQC in subsection P.3. We want to show that, under appropriate assumptions, HQC is ANON-CCA-secure in the QROM, and HQC leads to ANON-CCA-secure and SROB-CCA-secure hybrid PKE in the QROM. In order to do so, we show that the underlying HQC-DPKE of HQC-196 is strongly disjoint-simulatable under appropriate assumptions in subsection P.2. Unfortunately, we find that HQC-128/256 is *not* anonymous. HQC is obtained by applying $\mathsf{HU}^{\perp}$ to HQC-DPKE, and the strong disjoint simulatability implies that HQC-196 is SPR-CCA-secure and SSMT-CCA-secure in the QROM under those assumptions. We directly prove that HQC is SROB-CCA-secure in the QROM under an appropriate assumption. Those three properties lead to the anonymity and robustness of HQC-196 and hybrid PKE in the QROM as we wanted.

P.1 Review of HQC

HQC [AAB+20] is another code-based KEM scheme in the alternate candidates.

Let $\mathcal{R} := \mathbb{F}_2[x]/(x^r - 1)$. Let C be a decodable $[n_1 n_2, k]$ code generated by $G \in \mathbb{F}_2^{k \times n_1 n_2}$, where $n_1 n_2 \leq r$. Let decode be a decoder algorithm which corrects an error up to δ. Let $\mathcal{S}_w := \{x \in \mathcal{R} \mid \mathsf{HW}(x) = w\}$. For a polynomial $A = \sum_i a_i x^i \in \mathcal{R}$, we define $\mathsf{trunc}(A, l) = (a_0, \ldots, a_{l-1}) \in \mathbb{F}_2^l$. For concrete values, see Table 14.

Table 14. Parameter sets of HQC in Round 3.

parameter sets	r	n_1	k_1	d_1	n_2	k_2	d_2	w	w_e	w_r
hqc-128	17,669	46	16	31	384	8	192	66	75	75
hqc-192	35,851	56	24	32	640	8	320	100	114	114
hqc-256	57,637	90	32	59	640	8	320	131	149	149

The underlying PKE scheme of HQC, which we call HQC-PKE, is summarized as follows:

- $\mathsf{Gen}(pp)$: $h_0 \leftarrow \mathcal{R}$. $(x, y) \leftarrow \mathcal{S}_w^2$. Compute $h_1 := x + h_0 y$. Output $dk := (x, y)$ and $ek := (h_0, h_1)$.
- $\mathsf{Enc}(ek, \mu \in \mathbb{F}_2^k; (e, f, t) \in \mathcal{S}_{w_e} \times \mathcal{S}_{w_r} \times \mathcal{S}_{w_r})$: Output

$$c = (u, v) := (h_0 t + f, \mathsf{trunc}(h_1 t + e, n_1 n_2) \oplus \mu G) \in \mathcal{R} \times \mathbb{F}_2^{n_1 n_2}.$$

- $\mathsf{Dec}(dk, (u, v))$: Compute $a := v \oplus \mathsf{trunc}(uy, n_1 n_2) \in \mathbb{F}_2^{n_1 n_2}$ and output $\mathsf{decode}(a)$.

We next consider an intermediate PKE scheme $\mathsf{PKE}_0 = (\mathsf{Gen}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ where the encryption algorithm uses pseudorandomness, which we call HQC-PKE-PRG:

- $\mathsf{Gen}_0(pp) = \mathsf{Gen}(pp)$:
- $\mathsf{Enc}_0(ek, \mu; r)$: Use $\rho = \mathsf{SHAKE256}(r, \mathtt{0x02})$ to sample $(e, f, t) \in \mathcal{S}_{w_e} \times \mathcal{S}_{w_r} \times \mathcal{S}_{w_r}$. Output $(u, v) := \mathsf{Enc}(ek, \mu; (e, f, t))$.
- $\mathsf{Dec}_0(dk, (u, v)) = \mathsf{Dec}(dk, (u, v))$:

HQC applies a variant of the FO transform with explicit rejection $\mathsf{HFO}^{\perp} = \mathsf{HU}^{\perp} \circ \mathsf{T}$ to HQC-PKE-PRG PKE_0, where $\mathsf{G}(\mu) = \mathsf{SHAKE256}_{512}(\mu, \mathtt{0x03})$, $\mathsf{F}(\mu) = \mathsf{SHAKE256}_{512}(\mu, \mathtt{0x04})$. and $\mathsf{H}(\mu, (c_0, c_1)) = \mathsf{SHAKE256}_{512}(\mu, \mathtt{0x05})$. We can treat them as different random oracles because their input length differ.

Recall that $\mathsf{HFO}^{\perp}$ is $\mathsf{HU}^{\perp} \circ \mathsf{T}$. In what follows, we first study HQC-PKE's and HQC-PKE-PRG's properties and then study HQC-DPKE's properties, where HQC-DPKE is obtained by derandomizing HQC-PKE-PRG with transform T.

P.2 Properties of HQC-PKE

Grubbs et al. [GMP21a] showed properties of a variant of $\mathsf{HFO}^{\perp}$, in which $c_1 = \mathsf{F}(\mu, c_0)$ instead of $c_1 = \mathsf{F}(\mu)$. We here show HQC's anonymity directly by using properties of $\mathsf{HFO}^{\perp} = \mathsf{HU}^{\perp} \circ \mathsf{T}$.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek)$
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$\mu \leftarrow \{0,1\}^k$	$\mu' \coloneqq \mathsf{Dec}_0(dk, c_0)$
$\overline{dk} \coloneqq (dk, ek)$	$r \coloneqq \mathsf{G}(\mu)$	**if** $\mu' = \perp$, **then return** $K \coloneqq \perp$
return $(ek, \overline{dk})$	$c_0 \coloneqq \mathsf{Enc}_0(ek, \mu; r)$	$r' \coloneqq \mathsf{G}(\mu')$
	$c_1 \coloneqq \mathsf{F}(\mu)$	$c_0' \coloneqq \mathsf{Enc}_0(ek, \mu'; r')$
	$K \coloneqq \mathsf{H}(\mu, c_0, c_1)$	$c_1' \coloneqq \mathsf{F}(\mu')$
	return $((c_0, c_1), K)$	**if** $(c_0, c_1) \neq (c_0', c_1')$, **then return** $K \coloneqq \perp$
		else return $K \coloneqq \mathsf{H}(\mu', c_0, c_1)$

Fig. 23. HQC

Assumptions: For $b \in \{0,1\}$, define the finite set $\mathcal{F}_b \coloneqq \{h \in \mathcal{R} : h(1) \equiv b \pmod 2\}$, that is, a set of all binary vectors of length r and parity b. Similarly, for $b, b_0, b_1 \in \{0,1\}$, we define the sets

$$\mathcal{F}_b^{1,2} \coloneqq \{H = [1, h] \in \mathcal{R}^2 : h \in \mathcal{F}_b\}$$
$$\mathcal{F}_{b_0,b_1}^{2,3} \coloneqq \left\{H = \left[\begin{smallmatrix} 1 & 0 & h_0 \\ 0 & 1 & h_1 \end{smallmatrix}\right] \in \mathcal{R}^{2\times 3} : h_0 \in \mathcal{F}_{b_0} \wedge h_1 \in \mathcal{F}_{b_1}\right\}.$$

Definition P.1 (The 2-Decisional Quasi-Cyclic Syndrome Decoding (2-DQCSD) assumption [AAB+20]). *Fix $b \in \{0,1\}$, w, and $b' \coloneqq (1+b)w \bmod 2$. For any (Q)PPT adversary, it is hard to distinguish the following two distributions:*

- $(H, H \cdot (x, y))$, *where* $H \leftarrow \mathcal{F}_b^{1,2}$ *and* $(x_1, x_2) \leftarrow \mathcal{S}_w^2$.
- (H, z), *where* $H \leftarrow \mathcal{F}_b^{1,2}$ *and* $y \leftarrow \mathcal{F}_{b'}$.

Definition P.2 (The 3-Decisional Quasi-Cyclic Syndrome Decoding (3-DQCSD) assumption [AAB+20]). *Fix $b_0, b_1 \in \{0,1\}$, and w. Let $b_0' \coloneqq (1+b_0)w \bmod 2$ and $b_1' \coloneqq (1+b_1)w \bmod 2$. For any (Q)PPT adversary, it is hard to distinguish the following two distributions:*

- $(H, H \cdot (x_0, x_1, x_2))$, *where* $H \leftarrow \mathcal{F}_{b_0,b_1}^{2,3}$ *and* $(x_0, x_1, x_2) \leftarrow \mathcal{S}_w^3$.
- $(H, (z_0, z_1))$, *where* $H \leftarrow \mathcal{F}_{b_0,b_1}^{2,3}$, $z_0 \leftarrow \mathcal{F}_{b_0'}$, *and* $z_1 \leftarrow \mathcal{F}_{b_1'}$.

For collision-freeness, we define the following new assumption:

Definition P.3 (The 3-Computational Quasi-Cyclic Codeword Finding (3-CQCCF) assumption). *For any (Q)PPT adversary, given $(1, h, h')$ where $h, h' \leftarrow \mathcal{R}$, it is hard to find a non-zero codeword (f, t, t') whose Hamming weight is at most $4w_r$.*

Security of HQC-PKE: Using those assumptions, the proposers showed the IND-CPA security of HQC-PKE:

Lemma P.1 ([AAB+20, Theorem 5.1], adapted). *Assume that the 2-DQCSD and 3-DQCSD assumptions hold. Then, HQC-PKE is* IND-CPA*-secure (and* OW-CPA*-secure).*

By mimicking their proof, we can show that it is ciphertext-indistinguishable as follows:

Lemma P.2. *Assume that the 2-DQCSD and 3-DQCSD assumptions hold. Then, HQC-PKE is ciphertext-indistinguishable with a simulator that outputs $u \leftarrow \mathcal{F}_{b_0'}$ and $v \leftarrow \mathbb{F}_2^{n_1 n_2}$, where $b_0' \coloneqq (1 + h_0(1))w_r \bmod 2$.*

Proof (Proof Sketch). In what follows, we define the parity of h_1 as $b_1 \coloneqq (1 + h_0(1))w \bmod 2$, the parity of u as $b_0' \coloneqq (1 + h_0(1))w_r \bmod 2$, and the parity of $\tilde{v} = h_1 t + e$ as $b_1' \coloneqq w_e + b_1 w_r \bmod 2$. We consider games Game_i for $i = 0, \ldots, 4$ defined as follows:

- Game_0: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $h_0 \leftarrow \mathcal{R}$, $x, y \leftarrow \mathcal{S}_w$, and $h_1 \coloneqq x + h_0 y$.
  - Encryption: $\mu \leftarrow \mathbb{F}_2^k$, $e \leftarrow \mathcal{S}_{w_e}$, $t, f \leftarrow \mathcal{S}_{w_r}$, and compute $u \coloneqq h_0 t + f$ and $v \coloneqq \mathsf{trunc}(h_1 t + e, n_1 n_2) \oplus \mu G$.
- Game_1: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $h_0 \leftarrow \mathcal{R}$, $h_1^+ \leftarrow \mathcal{F}_{b_1}$.
  - Encryption: $\mu \leftarrow \mathbb{F}_2^k$, $e \leftarrow \mathcal{S}_{w_e}$, $t, f \leftarrow \mathcal{S}_{w_r}$, and compute $u \coloneqq h_0 t + f$ and $v \coloneqq \mathsf{trunc}(h_1^+ t + e, n_1 n_2) \oplus \mu G$.

– Game_2: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $h_0 \leftarrow \mathcal{R}$, $h_1^+ \leftarrow \mathcal{F}_{b_1}$.
  - Encryption: $u \leftarrow \mathcal{F}_{b_0'}$, $\tilde{y} \leftarrow \mathcal{F}_{b_1'}$, and $v := \mathsf{trunc}(\tilde{y}) \oplus \mu G$.
– Game_3: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $h_0 \leftarrow \mathcal{R}$, $h_1^+ \leftarrow \mathcal{F}_{b_1}$.
  - Encryption: $u \leftarrow \mathcal{F}_{b_0'}$ and $v \leftarrow \mathbb{F}_2^{n_1 n_2}$.
– Game_4: In this game, an encryption key and a target ciphertext is computed as follows:
  - Key generation: $h_0 \leftarrow \mathcal{R}$, $x, y \leftarrow \mathcal{S}_w$, and $h_1 := x + h_0 y$.
  - Encryption: $u \leftarrow \mathcal{F}_{b_0'}$ and $v \leftarrow \mathbb{F}_2^{n_1 n_2}$.

Game_0 and Game_1 are computationally indistinguishable under the 2-DQCSD assumption. Game_1 and Game_2 are computationally indistinguishable under the 3-DQCSD assumption. Game_2 and Game_3 are statistically indistinguishable, because trunc truncates $r - n_1 n_2$ bits of $\tilde{y} \leftarrow \mathcal{F}_{b_1'}$ in Game_2 and thus, $\mathsf{trunc}(\tilde{y}, n_1 n_2)$'s distribution is statistically close to the uniform distribution over $\mathbb{F}_2^{n_1 n_2}$. Game_3 and Game_4 are computationally indistinguishable under the 2-DQCSD assumption. Summing up those (in)equalities, we obtain the lemma. □

We notice that HQC-196 are strongly pseudorandom, while HQC-128 and HQC-256 are not:[7]

Corollary P.1. *HQC-196 is strongly ciphertext-indistinguishable, while HQC-128 and HQC-256 are not.*

Proof. Let us compute the parity of h_0, $h_0(1) \bmod 2$, the parity of $h_1 = x + h_0 y$, $b_1 := (1 + h_0(1))w \bmod 2$, the parity of $u = h_0 t + f$, $b_u := (1 + h_0(1))w_r \bmod 2$, and the parity of $h_1 t + e$, $b_v := (1 + h_1(1))w_r \bmod 2$, According to Table 14, the parity b_1 of h_1 is 0, 0, and $1+h_0(1) \bmod 2$, and the parity b_u of u is $1+h_0(1) \bmod 2$, 0, and $1 + h_0(1) \bmod 2$, for HQC-128/192/256, respectively. Thus, the simulator for HQC-192 can ignore the encryption key (h_0, h_1) and we can say that HQC-196 are strongly ciphertext-indistinguishable. However, the simulator for HQC-128/256 depends on $h_0(1)$ and HQC-128/256 is not strongly ciphertext-indistinguishable. Indeed, the parity of u leaks the information of h_0 of the encryption key for HQC-128/256. □

Security of HQC-PKE-PRG: We next consider HQC-PKE-PRG, whose encryption algorithm uses a PRG $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$ instead of true randomness. The IND-CPA security and ciphertext indistinguishability of HQC-PKE-PRG follows from PRG's quantum security tightly.

Lemma P.3. *Assume that the 2-DQCSD and 3-DQCSD assumptions hold and* $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$ *is quantumly-secure PRG. Then, HQC-PKE-PRG is ciphertext-indistinguishable and* IND-CPA*-secure (and* OW-CPA*-secure). In addition, HQC-PKE-PRG for HQC-196 is strongly ciphertext-indistinguishable.*

Security of HQC-DPKE: We then consider HQC-DPKE obtained by derandomizing HQC-PKE-PRG by T.

Lemma P.4. *Assume that the 2-DQCSD and 3-DQCSD assumptions hold and* $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$ *is quantumly-secure PRG. Then, HQC-DPKE is disjoint-simulatable. Especially, HQC-DPKE for HQC-196 is strongly disjoint-simulatable.*

Proof. Statistical disjointess follows from the fact that $|\mathcal{S}(1^\kappa)| \approx 2^r/2 \cdot 2^{n_1 n_2}$ and $|\mathsf{Enc}'(ek, \mathcal{M})| \leq 2^k$. We can show ciphertext indistinguishability by invoking Theorem D.1 since HQC-PKE-PRG is ciphertext indistinguishable and OW-CPA-secure (Lemma P.3). Strong disjoint simulatability for HQC-196 follows from Lemma P.3. □

P.3 Properties of HQC

Combining HQC-DPKE's strong disjoint-simulatability with previous theorems on $\mathsf{HU}^\perp$, we obtain the following theorems.

Theorem P.1. *Assume that the 2-DQCSD and 3-DQCSD assumptions hold and* $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$ *is quantumly-secure PRG. Then, HQC-196 is* SPR-CCA*-secure in the QROM. It is also* $1/2^{512}$*-sparse in the QROM.*

Proof. Under the 2-DQCSD and 3-DQCSD assumptions and quantum security of $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$, HQC-DPKE for HQC-196 is strongly disjoint-simulatable (Lemma P.4). Applying Theorem G.2, we obtain the SPR-CCA security in the QROM. In addition, using the fact that $\mathsf{F}(\cdot) = \mathsf{SHAKE256}_{512}(\cdot, \mathtt{0x04})$'s range is $\{0,1\}^{512}$ and applying Theorem G.4, we obtain $1/2^{512}$-sparseness in the QROM. □

Theorem P.2. *Suppose that the 2-DQCSD and 3-DQCSD assumptions hold and* $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$ *is quantumly-secure PRG. Then, HQC-196 is* ANON-CCA*-secure in the QROM.*

[7] Modified in 2022-09-22: In the previous versions, we consider HQC-128 and HQC-196 are SPR and HQC-256 is not.

Proof. Due to Theorem P.1, under the hypothesis, HQC-196 is SPR-CCA-secure in the QROM. Thus, applying Theorem 2.5, we have that, under those assumptions, HQC-196 is ANON-CCA-secure in the QROM. □

We next consider HQC's SROB-CCA security.

Theorem P.3. *Suppose that the 3-CQCCF assumption holds. Then, HQC is* SROB-CCA*-secure.*

Proof (Proof sketch:). Given $(1, h_{0,0}, h_{1,0})$ with $h_{0,0}, h_{1,0} \leftarrow \mathcal{R}$, we generate decryption keys and encryption keys $ek_i = (h_{i,0}, h_{i,1})$ and $dk_i = (x_i, y_i)$ for $i \in zo$. We give them to an adversary against SROB-CCA security of KEM. Suppose that the adversary outputs $c = (u, v)$ and the adversary wins. If so, it should be decapsulated into $K_0 \neq \perp$ and $K_1 \neq \perp$. Thus, c should be decrypted into μ_0 and μ_1 by using dk_0 and dk_1, respectively. In re-encryption check, we have $(e_0, f_0, t_0) := \mathsf{SHAKE256}(\mathsf{G}(\mu_0), \mathtt{0x02})$ and $(e_1, f_1, t_1) := \mathsf{SHAKE256}(\mathsf{G}(\mu_1), \mathtt{0x02})$, and $u = h_{0,0}t_0 + f_0 = h_{1,0}t_1 + f_1$. This implies $(1, h_{0,0}, h_{1,0}) \cdot (f_0 + f_1, t_0, t_1) = 0$ and $(f_0 + f_1, t_0, t_1)$ is the solution of the 3-CQCCF problem. □

We finally consider the anonymity and robustness of the hybrid PKE using HQC as KEM.

Theorem P.4. *Suppose that the 2-DQCSD, 3-DQCSD, and 3-CQCCF assumptions hold,* $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$ *is quantumly-secure PRG, and HQC-DPKE is* δ*-correct with negligible* δ*. In addition, we assume that* $1/2^{512}$ *is negligible. Then, HQC-196 leads to* ANON-CCA*-secure and* SROB-CCA*-secure hybrid PKE in the QROM, combined with* SPR-OTCCA*-secure and* INT-CTXT*-secure DEM.*

Proof. Due to Theorem P.1, under the 2-DQCSD aand 3-DQCSD assumptions and the assumptions on the quantum security of $\mathsf{SHAKE256}(\cdot, \mathtt{0x02})$ and the correctness, HQC-196 is SPR-CCA-secure and SSMT-CCA-secure in the QROM. Thus, combining HQC-196 with SPR-OTCCA-secure and INT-CTXT-secure DEM, we obtain a SPR-CCA-secure hybrid PKE in the QROM (Theorem 3.1).
Moreover, HQC is SROB-CCA-secure in the QROM (Theorem P.3) under the 3-CQCCF assumption. Thus, the hybrid PKE is SROB-CCA-secure under the same assumption (Theorem 2.1). □

Q Streamlined NTRU Prime

Review of Streamlined NTRU Prime: Streamlined NTRU Prime is one of two KEMs in NTRU Prime [BBC+20]. We briefly review Streamlined NTRU Prime. The underlying CPA-secure PKE scheme, which is called as 'Streamlined NTRU Prime Core', is summarized as follows:

- $\mathsf{Gen}(pp)$: The key-generation algorithm outputs ek and dk.
- $\mathsf{Enc}(ek, \mu)$: The encryption algorithm is deterministic. Taking $\mu \in \mathcal{M}$, it outputs c.
- $\mathsf{Dec}(dk, c)$: The decryption algorithm is deterministic and outputs $\mu \in \mathcal{M}$ or special $\mu_{\mathsf{invalid}} \in \mathcal{M}$.

Streamlined NTRU Prime [BBC+20] applies $\mathsf{HU}^{\not\perp,\mathsf{prf}}$ to Streamlined NTRU Prime Core, where $\mathsf{H}(\mu, c) = \mathsf{SHA512}_{256}\big(\mathtt{0x01}, \mathsf{SHA512}_{256}(\mathtt{0x03}, \mu), c\big)$ $\mathsf{H}_{\mathsf{prf}}(s, c) = \mathsf{SHA512}_{256}\big(\mathtt{0x00}, \mathsf{SHA512}_{256}(\mathtt{0x03}, s), c\big)$ $\mathsf{F}(\mu, ek) = \mathsf{SHA512}_{256}\big(\mathtt{0x02}, \mathsf{SHA512}_{256}(\mathtt{0x03}, \mu), \mathsf{SHA512}_{256}(\mathtt{0x04}, ek)\big)$, and is defined as in Figure 24.

$\mathsf{Gen}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek, s)$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	$\mu \leftarrow \mathcal{M}$	$\mu' := \mathsf{Dec}(dk, c_0)$
$s \leftarrow \{0,1\}^l$	$c_0 := \mathsf{Enc}(ek, \mu)$	**if** $\mu' = \perp$, **then return** $K := \mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$
$\overline{dk} := (dk, ek, s)$	$c_1 := \mathsf{F}(\mu, ek)$	$c_0' := \mathsf{Enc}(ek, \mu')$
return $(ek, \overline{dk})$	$K := \mathsf{H}(\mu, c_0, c_1)$	$c_1' := \mathsf{F}(\mu', ek)$
	return $((c_0, c_1), K)$	**if** $(c_0, c_1) = (c_0', c_1')$, **then return** $K := \mathsf{H}(\mu', c_0, c_1)$
		else return $K := \mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$

Fig. 24. Streamlined NTRU Prime

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek, s)$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	$\mu \leftarrow \mathcal{M}$	$\mu' := \mathsf{Dec}(dk, c_0)$
$s \leftarrow \{0,1\}^\ell$	$c_0 := \mathsf{Enc}(ek, \mu)$	**if** $\mu' = \bot$, **then return** $K := \mathsf{H}_0(\mathsf{H}_3(s), c_0, c_1)$
$\overline{dk} := (dk, ek, s)$	$c_1 := \mathsf{H}_2(\mathsf{H}_3(\mu), \mathsf{H}_4(ek))$	$c_0' := \mathsf{Enc}(ek, \mu')$
return $(ek, \overline{dk})$	$K := \mathsf{H}_1(\mathsf{H}_3(\mu), c_0, c_1)$	$c_1' := \mathsf{H}_2(\mathsf{H}_3(\mu'), \mathsf{H}_4(ek))$
	return $((c_0, c_1), K)$	**if** $(c_0, c_1) \neq (c_0', c_1')$, **then return** $K := \mathsf{H}_0(\mathsf{H}_3(s), c_0, c_1)$
		else return $K := \mathsf{H}_1(\mathsf{H}_3(\mu'), c_0, c_1)$

Fig. 25. $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf},\prime}[\mathsf{PKE}, \mathsf{H}_0, \mathsf{H}_1, \mathsf{H}_2, \mathsf{H}_3, \mathsf{H}_4]$.

Security: We found that Streamlined NTRU Prime has a problem. For simplicity, let $\mathsf{H}_i(x) = \mathsf{SHA512}_{256}(\texttt{0x0}i\|x)$ as in [BBC+20]. Using this notation, we have

- $\mathsf{H}(\mu, c) = \mathsf{H}_1(\mathsf{H}_3(\mu)\|c)$
- $\mathsf{H}_{\mathsf{prf}}(s, c) = \mathsf{H}_0(\mathsf{H}_3(s)\|c)$
- $\mathsf{F}(\mu, ek) = \mathsf{H}_2(\mathsf{H}_3(\mu)\|\mathsf{H}_4(ek))$.

Using them, the conversion of Streamlined NTRU Prime is summarized as in Figure 25.

We can assume H_i as random oracles. The IND-CCA security proof in the *ROM* is straightforward because, intuitively speaking, the adversary cannot distinguish real K with random one unless it asks μ of c_0 to H_3 and the simulation of decapsulation is done by the list of queries to the random oracles. Unfortunately, we have a technical obstacle for the IND-CCA security in the *QROM*.

We have tried to show its security via an intermediate transform $\mathsf{HU}^{\bot,\prime}[\mathsf{PKE}, \mathsf{H}_2, \mathsf{H}_3, \mathsf{H}_4]$, in which $K = \mathsf{H}_3(\mu)$ and the decapsulation algorithm returns $\bot$ for a invalid ciphertext. If this was secure, then we can convert the security proof of $\mathsf{HU}^{\bot\prime}$ into that of $\mathsf{HU}^{\not\perp,\mathsf{prf},\prime}$ using a simple reduction. Unfortunately, $\mathsf{HU}^{\bot,\prime}$ is not IND-CPA-secure because $c_1 = \mathsf{H}_2(\mathsf{H}_3(\mu), \mathsf{H}_4(ek)) = \mathsf{H}_2(K, \mathsf{H}_4(ek))$ and we can check if K is real by checking if $c_1 = \mathsf{H}_2(K, \mathsf{H}_4(ek))$ or not.

If H_3 is length-preserving, we could use the technique by Grubbs et al. [GMP21a] for QROM security proof. Unfortunately, μ is longer than 256-bits and this is not length-preserving.

If F is not nested on μ, we can prove the security as follows: We first consider $\mathsf{HU}^{\bot}_m[\mathsf{PKE}, \mathsf{H}_3, \mathsf{F}]$, which is SPR-CCA-secure if PKE is strongly disjoint-simulatable. We then consider an indifferentiable reduction defined as follows: if $K \neq \bot$, then we rewrite the decapsulation result as $\mathsf{H}_1(K\|c)$; if $K = \bot$, then we rewrite the decapsulation result as $\mathsf{H}_0(\mathsf{H}_3(s)\|c)$. It is easy to verify that $\mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}, \mathsf{H}, \mathsf{F}, \mathsf{H}_{\mathsf{prf}}]$ is SPR-CCA-secure if $\mathsf{HU}^{\bot}_m[\mathsf{PKE}, \mathsf{H}_3, \mathsf{F}]$ is SPR-CCA-secure.

Bernstein [Ber21] suggests to use the domain extension of quantum random oracles in [Zha19, Section 5], which is shown *quantumly indifferentiable*. Let $C^{\mathsf{H}_1,\mathsf{H}_2}(x, y) = \mathsf{H}_1(\mathsf{H}_2(x), y)$. Roughly speaking, we say $C^{\mathsf{H}_1,\mathsf{H}_2}$ is indifferentiable if any efficient adversary cannot distinguish oracles $\mathsf{H}_1, \mathsf{H}_2, C^{\mathsf{H}_1,\mathsf{H}_2}$ from $\mathsf{Sim}^{\mathsf{H}}, \mathsf{H}$, where Sim queries to H and simulates H_1 and H_2. We did not check the detail and leave to show the IND-CCA security (and anonymity) in the QROM as an open problem.

It might be interesting to study anonymity and robustness in the ROM.

R NTRU LPRime

NTRU LPRime is the other KEM in NTRU Prime [BBC+20].

We briefly review NTRU LPRime [BBC+20] in subsection R.1, discuss the security properties of the underlying PKEs, NTRU LPRime Core and NTRU LPRime Expand, and its derandomized version, NTRU LPRime DPKE, in subsection R.2, and discuss the security properties of NTRU LPRime in subsection R.3. We want to show that, under appropriate assumptions, NTRU LPRime is ANON-CCA-secure in the QROM, and NTRU LPRime leads to ANON-CCA-secure and SROB-CCA-secure hybrid PKE in the QROM. In order to do so, we show that the underlying NTRU LPRime DPKE is strongly disjoint-simulatable under appropriate assumptions in subsection R.2. NTRU LPRime is obtained by applying a variant of $\mathsf{HU}^{\not\perp,\mathsf{prf}}$ to NTRU LPRime DPKE, and the strong disjoint simulatability implies that NTRU LPRime is SPR-CCA-secure and SSMT-CCA-secure in the QROM under those assumptions. We directly prove that NTRU LPRime is SCFR-CCA-secure in the QROM under an appropriate assumption. Those three properties lead to the anonymity of NTRU LPRime and hybrid PKE in the QROM as we wanted.

Table 15. Parameter sets of ntrulpr of NTRU Prime

parameter sets	p	q	w	δ	τ_0	τ_1	τ_2	τ_3
ntrulpr653	653	4621	252	289	2175	113	2031	290
ntrulpr761	761	4591	250	292	2156	114	2007	287
ntrulpr857	857	5167	281	329	2433	101	2265	324
ntrulpr953	953	6343	345	404	2997	82	2798	400
ntrulpr1013	1013	7177	392	450	3367	73	3143	449
ntrulpr1277	1277	7879	429	502	3724	66	3469	496

R.1 Review of NTRU LPRime

NTRU LPRime has parameter sets $p, q, w, \delta, \tau_0, \tau_1, \tau_2$, and τ_3. We note that $q = 6q' + 1$ for some q' and $q \geq 16w + 2\delta + 3$. For concrete values, see Table 15.

Let $\mathcal{R} := \mathbb{Z}[x]/(x^p - x - 1)$ and $\mathcal{R}_q := \mathbb{Z}_q[x]/(x^p - x - 1)$. Let $\mathcal{S} := \{a = \sum_{i=0}^{p-1} a_i x^i \in \mathcal{R} \mid a_i \in \{-1, 0, +1\}, \mathsf{HW}(a) = w\}$, a set of "short" polynomials.

For $a \in [-(q-1)/2, (q-1)/2]$, define $\mathsf{Round}(a) = 3 \cdot \lceil a/3 \rfloor$.[8] For a polynomial $A = \sum_i a_i x^i \in \mathcal{R}_q$, we define $\mathsf{trunc}(A, l) = (a_0, \ldots, a_{l-1}) \in \mathbb{Z}_q^l$. For $C \in [0, q)$, define $\mathsf{Top}(C) = \lfloor (\tau_1(C + \tau_0) + 2^{14})/2^{15} \rfloor$. For $T \in [0, 16)$, define $\mathsf{Right}(T) = \tau_3 T - \tau_2 \in \mathbb{Z}_q$. For $a \in \mathbb{Z}$, define $\mathsf{Sign}(a) = 1$ if $a < 0$, 0 otherwise.

The underlying CPA-secure PKE scheme 'NTRU LPRime Core' is defined as follows:

- $\mathsf{Gen}(pp)$: Generate $A \leftarrow \mathcal{R}_q$ and $dk \leftarrow \mathcal{S}$. Compute $B := \mathsf{Round}(A \cdot dk)$. Output $ek := (A, B)$ and dk.
- $\mathsf{Enc}(ek, \mu \in \{0,1\}^{256})$: Choose $t \leftarrow \mathcal{S}$ and output

$$(U, V) := \big(\mathsf{Round}(t \cdot A), \mathsf{Top}(\mathsf{trunc}(t \cdot B, 256) + \mu(q-1)/2)\big).$$

- $\mathsf{Dec}(dk, (U, V))$: Compute $r := \mathsf{Right}(V) - \mathsf{trunc}(dk \cdot U, 256) + (4w + 1) \cdot \mathbf{1}_{256} \in \mathbb{Z}^{256}$ and outputs $\mu := \mathsf{Sign}(r \bmod^{\pm} q)$.

NTRU LPRime Core is perfectly correct.

We next consider an intermediate PKE scheme $\mathsf{PKE}_0 = (\mathsf{Gen}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ where the encryption algorithm uses pseudorandomness, which is called as 'NTRU LPRime Expand':

- $\mathsf{Gen}_0(pp) = \mathsf{Gen}(pp)$:
- $\mathsf{Enc}_0(ek, \mu; r)$: Use $\rho = \mathsf{AES256\text{-}CTR}(r)$ to sample $t \leftarrow \mathcal{S}$. Output $(U, V) := \mathsf{Enc}(ek, \mu; t)$.
- $\mathsf{Dec}_0(dk, (U, V)) = \mathsf{Dec}(dk, (U, V))$:

NTRU LPRime applies a variant of $\mathsf{HFO}^{\not\perp, \mathsf{prf}}$ to NTRU LPRime Expand PKE_0, where $\mathsf{G}(\mu) = \mathsf{SHA512}_{256}(\mathtt{0x05}, \mu)$, $\mathsf{H}(\mu, c) = \mathsf{SHA512}_{256}(\mathtt{0x01}, \mu, c)$, $\mathsf{H}_{\mathsf{prf}}(s, c) = \mathsf{SHA512}_{256}(\mathtt{0x00}, s, c)$, $\mathsf{F}(\mu, \mathsf{H}'(ek)) = \mathsf{SHA512}_{256}\big(\mathtt{0x02}, \mu, \mathsf{SHA512}_{256}(\mathtt{0x04}, ek)\big)$ and is defined as in Figure 26.

$\overline{\mathsf{Gen}}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_0, c_1))$, where $\overline{dk} = (dk, ek, s)$
$(ek, dk) \leftarrow \mathsf{Gen}_0(1^\kappa)$	$\mu \leftarrow \{0,1\}^{\ell(\kappa)}$	$\mu' := \mathsf{Dec}_0(dk, c_0)$
$s \leftarrow \{0,1\}^{\ell(\kappa)}$	$r := \mathsf{G}(\mu)$	$r' := \mathsf{G}(\mu')$
$\overline{dk} := (dk, ek, s)$	$c_0 := \mathsf{Enc}_0(ek, \mu; r)$	$c_0' := \mathsf{Enc}_0(ek, \mu'; r')$
return $(ek, \overline{dk})$	$c_1 := \mathsf{F}(\mu, \mathsf{H}'(ek))$	$c_1' := \mathsf{F}(\mu', \mathsf{H}'(ek))$
	$K := \mathsf{H}(\mu, c_0, c_1)$	**if** $(c_0, c_1) \neq (c_0', c_1')$
	return $((c_0, c_1), K)$	**then return** $K := \mathsf{H}_{\mathsf{prf}}(s, c_0, c_1)$
		else return $K := \mathsf{H}(\mu', c_0, c_1)$

Fig. 26. NTRU LPRime

[8] When $q = 6q' + 1$, $\mathsf{Round}([-(q-1)/2, (q-1)/2]) \in [-(q-1)/2, (q-1)/2]$.

R.2 Properties of NTRU LPRime Core, NTRU LPRime Expand, and NTRU LPRime DPKE

Security of NTRU LPRime Core and NTRU LPRime Expand: We directly assume that NTRU LPRime Core is ciphertext-indistinguishable with simulator $\mathcal{S}$ that samples $a \leftarrow \mathcal{R}$, computes $U := \mathsf{Round}(a)$, samples $V \leftarrow (\mathbb{Z}/16\mathbb{Z})^{256}$, and outputs (U, V). Moreover, we assume that NTRU LPRime Core is IND-CPA-secure (and OW-CPA-secure). The IND-CPA security and ciphertext indistinguishability of NTRU LPRime Expand follows from PRG's quantum security tightly.

Lemma R.1. *Assume that NTRU LPRime Core is ciphertext-indistinguishable with simulator $\mathcal{S}$ and is* IND-CPA*-secure, and* AES256-CTR *is quantumly-secure PRG. Then, NTRU LPRime Expand is strongly ciphertext-indistinguishable and* IND-CPA*-secure (and* OW-CPA*-secure).*

Security of NTRU LPRime DPKE: We then consider NTRU LPRime DPKE obtained by applying T to NTRU LPRime Expand.

Lemma R.2. *Suppose that NTRU LPRime Core is ciphertext-indistinguishable with simulator $\mathcal{S}$ and is* IND-CPA*-secure, and* AES256-CTR *is quantumly-secure PRG. Then, NTRU LPRime DPKE is strongly disjoint-simulatable.*

Proof. Statistical disjointess follows from the fact that $|\mathcal{S}(1^\kappa)| \approx (q/3)^p \cdot 16^{256}$ and $|\mathsf{Enc}(ek, \mathcal{M})| \leq 2^{256}$. We can show ciphertext indistinguishability by invoking Theorem D.1 since NTRU LPRime Expand is ciphertext-indistinguishable and one-way (Lemma R.1). □

R.3 Properties of NTRU LPRime

$\mathsf{PKE}' := \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$ is strongly disjoint-simulatable. Recall that $\mathsf{HFO}_{\not\perp,\mathsf{prf}}$ is $\mathsf{HU}^{\not\perp,\mathsf{prf}} \circ \mathsf{T}$. Applying $\mathsf{HU}^{\not\perp,\mathsf{prf}}$ to $\mathsf{PKE}' = \mathsf{T}[\mathsf{PKE}_0, \mathsf{G}]$, we obtain $\mathsf{KEM} = \mathsf{HU}^{\not\perp,\mathsf{prf}}[\mathsf{PKE}', \mathsf{H}, \mathsf{F}]$. After applying our theorems, we summarize the security properties of NTRU LPRime as follows:

- Assume that the underlying DPKE of NTRU LPRime PKE' is strongly disjoint-simulatable with simulator that samples $a \leftarrow \mathcal{R}$, computes $U := \mathsf{Round}(a)$, samples $V \leftarrow (\mathbb{Z}/16\mathbb{Z})^{256}$, and outputs (U, V).
- Then, NTRU LPRime is SPR-CCA-secure and SSMT-CCA-secure in the QROM.
- NTRU LPRime is SCFR-CCA-secure if the colliding probability of ek is negligible since F takes μ and ek as input.
- NTRU LPRime is ANON-CCA-secure.
- NTRU LPRime leads to ANON-CCA-secure, SROB-CCA-secure hybrid PKE.

Combining NTRU LPRime DPKE's strong disjoint-simulatability with previous theorems on $\mathsf{HU}^{\not\perp,\mathsf{prf}}$, we obtain the following theorem.

Theorem R.1. *Suppose that NTRU LPRime Core is ciphertext-indistinguishable with simulator $\mathcal{S}$ and is* IND-CPA*-secure, and* AES256-CTR *is quantumly-secure PRG. Then, NTRU LPRime is* SPR-CCA*-secure and* SSMT-CCA*-secure in the QROM.*

Proof. Suppose that NTRU LPRime Core is ciphertext-indistinguishable with simulator $\mathcal{S}$ and is IND-CPA-secure, NTRU LPRime DPKE is strongly disjoint-simulatable (Lemma R.2). Applying Theorem I.2 and Theorem I.4, we obtain the theorem. □

Next, we directly prove the SCFR-CCA security in the QROM. The proof is very similar to that for the modified Classic McEliece (Theorem K.3).

Theorem R.2. *Let $\mathsf{Col}_{\mathsf{Gen}_0}$ be the event that when generating two keys $(ek_i, dk_i) \leftarrow \mathsf{Gen}_0(1^\kappa)$ for $i \in \{0, 1\}$, they collide, that is, $ek_0 = ek_1$. If $\Pr[\mathsf{Col}_{\mathsf{Gen}_0}]$ is negligible, then NTRU LPRime is* SCFR-CCA*-secure in the QROM.*

Proof. Suppose that an adversary outputs a ciphertext $c = (c_0, c_1)$ which is decapsulated into $K \neq \perp$ by $\overline{dk}_0$ and $\overline{dk}_1$, that is, $\overline{\mathsf{Dec}}(\overline{dk}_0, c) = \overline{\mathsf{Dec}}(\overline{dk}_1, c)$. Let us define $\mu'_i := \mathsf{Dec}_0(dk_i, c_0)$ for $i \in \{0, 1\}$. We also define $\mu_i := \mu'_i$ if $c_0 = \mathsf{Enc}_0(ek_i, \mu'_i; \mathsf{G}(\mu'_i))$ and $c_1 = \mathsf{F}(\mu'_i, \mathsf{H}'(ek_i))$, and $\perp$ otherwise.
We consider seven cases defined as follows:

1. Case 1-1 ($\mu_0 = \mu_1 \neq \perp$ and $ek_0 = ek_1$): This case rarely occurs since $\Pr[\mathsf{Col}_{\mathsf{Gen}}]$ is negligible.
2. Case 1-2 ($\mu_0 = \mu_1 \neq \perp$, $ek_0 \neq ek_1$, and $\mathsf{H}'(ek_0) = \mathsf{H}'(ek_1)$): In this case, we have $\mathsf{H}'(ek_0) = \mathsf{H}'(ek_1)$ with $ek_0 \neq ek_1$ and we succeed to find a collision for H', which is negligible for any QPT adversary (Lemma 2.3).
3. Case 1-3 ($\mu_0 = \mu_1 \neq \perp$, $ek_0 \neq ek_1$, and $\mathsf{H}'(ek_0) \neq \mathsf{H}'(ek_1)$): In this case, we have $d = \mathsf{F}(\mu_0, \mathsf{H}'(ek_0)) = \mathsf{F}(\mu_1, \mathsf{H}'(ek_1))$ with $(\mu_0, \mathsf{H}'(ek_0)) \neq (\mu_1, \mathsf{H}'(ek_1))$ and we succeed to find a collision for F, which is negligible for any QPT adversary (Lemma 2.3).

4. Case 2 ($\bot \neq \mu_0 \neq \mu_1 \neq \bot$): In this case, the decapsulation algorithm outputs $K = \mathsf{H}(\mu_0) = \mathsf{H}(\mu_1)$ and we succeed to find a collision for H, which is negligible for any QPT adversary (Lemma 2.3).
5. Case 3 ($\mu_0 = \bot$ and $\mu_1 \neq \bot$): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1)$ and $\mathsf{H}(\mu_1, c_0, c_1)$ and we find a claw $((s_0, c_0, c_1), (\mu_1, c_0, c_1))$ of $\mathsf{H}_{\mathsf{prf}}$ and H. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
6. Case 4 ($\mu_0 \neq \bot$ and $\mu_1 = \bot$): In this case, the decapsulation algorithms output $K = \mathsf{H}(\mu_0, c_0, c_1) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a claw $((\mu_0, c_0, c_1), (s_1, c_0, c_1))$ of H and $\mathsf{H}_{\mathsf{prf}}$. The probability that we find such claw is negligible for any QPT adversary (Lemma 2.4).
7. Case 5 (The other cases): In this case, the decapsulation algorithms output $K = \mathsf{H}_{\mathsf{prf}}(s_0, c_0, c_1) = \mathsf{H}_{\mathsf{prf}}(s_1, c_0, c_1)$ and we find a collision $((s_0, c_0, c_1), (s_1, c_0, c_1))$ of $\mathsf{H}_{\mathsf{prf}}$ if $s_0 \neq s_1$, which occurs with probability at least $1 - 1/2^n$. The probability that we find such collision is negligible for any QPT adversary (Lemma 2.3).

Thus, we conclude that the advantage of the adversary is negligible. □

Theorem R.3. *Suppose that NTRU LPRime Core is ciphertext-indistinguishable with simulator $\mathcal{S}$ and is* IND-CPA*-secure, and* AES256-CTR *is quantumly-secure PRG. Then, NTRU LPRime is* ANON-CCA*-secure in the QROM.*

Proof. Due to Theorem R.1, under the hypothesis, NTRU LPRime is SPR-CCA-secure in the QROM. Thus, applying Theorem 2.5, we have that, under those assumptions, NTRU LPRime is ANON-CCA-secure in the QROM. □

Theorem R.4. *Let $\mathsf{Col}_{\mathsf{Gen}_0}$ be the event that when generating two keys $(ek_i, dk_i) \leftarrow \mathsf{Gen}_0(1^\kappa)$ for $i \in \{0,1\}$, they collide, that is, $ek_0 = ek_1$. Suppose that $\Pr[\mathsf{Col}_{\mathsf{Gen}_0}]$ is negligible. Suppose that NTRU LPRime Core is ciphertext-indistinguishable with simulator $\mathcal{S}$ and is* IND-CPA*-secure, and* AES256-CTR *is quantumly-secure PRG. Then, NTRU LPRime leads to* ANON-CCA*-secure and* SROB-CCA*-secure hybrid PKE in the QROM, combined with* SPR-OTCCA*-secure and* FROB*-secure DEM.*

Proof. Due to Theorem R.1, under the hypothesis, NTRU LPRime is SPR-CCA-secure and SSMT-CCA-secure in the QROM. Thus, combining NTRU LPRime with SPR-OTCCA-secure DEM, we obtain a SPR-CCA-secure hybrid PKE in the QROM (Theorem 3.2). Moreover, NTRU LPRime is SCFR-CCA-secure in the QROM (Theorem R.2) under the assumption that $\Pr[\mathsf{Col}_{\mathsf{Gen}_0}]$ is negligible. Thus, if DEM is FROB-secure, then the hybrid PKE is SROB-CCA-secure (Theorem 2.2). □

S SIKE

We briefly review SIKE [JAC+20] in subsection S.1, discuss the security properties of the underlying PKE, SIKE-PKE, and its derandomized version, SIKE-DPKE, in subsection S.2, and discuss the security properties of SIKE in subsection S.3. We want to show that, under appropriate assumptions, SIKE is ANON-CCA-secure in the QROM, and SIKE leads to ANON-CCA-secure and SROB-CCA-secure hybrid PKE in the QROM. In order to do so, we show that the underlying SIKE-DPKE of SIKE is strongly disjoint-simulatable under appropriate assumptions and XCFR-secure in subsection S.2. SIKE is obtained by applying $\mathsf{U}^{\not\perp}$ to SIKE-DPKE, and the former implies that SIKE is SPR-CCA-secure and SSMT-CCA-secure in the QROM under those assumptions and the latter implies that SIKE is SCFR-CCA-secure in the QROM. Those three properties lead to the anonymity of SIKE and hybrid PKE in the QROM as we wanted.

S.1 Review of SIKE

SIKE [JAC+20] is KEM scheme based on SIDH [JD11, ?]. For a survey of isogeny-based cryptography, we recommend reading [?].

Let $p = 2^{e_2}3^{e_3} - 1$. Let E be a supersingular elliptic curve over $\mathbb{F}_{p^2}$. Let $P_2, Q_2 \in E[2^{e_2}]$ and $P_3, Q_3 \in E[3^{e_3}]$ linearly independent points of order 2^{e_2} and 3^{e_3} respectively. Let $\{0,1\}^n$ be a message space and let $\mathsf{L} : \mathbb{F}_{p^2} \to \{0,1\}^n$ be a random oracle, instantiated by $\mathsf{SHAKE256}_n(\cdot)$.

Roughly speaking, the underlying PKE scheme [JAC+20, Algorithm 1], which we call SIKE-PKE, is summarized as follows (for the details, see the specification):

- $\mathsf{isogen}_\ell(dk_\ell)$ with $(m, \ell) = (2,3)$ or $(3,2)$: On input $dk_\ell \in [0, \ell^{e_\ell})$, compute $S := P_\ell + [dk_\ell]Q_\ell$, compute isogeny $\phi_\ell : E \to E/\langle S \rangle$, and compute $E'_m := E/\langle S \rangle = \phi_\ell(E)$. Compute $P'_m := \phi_\ell(P_m)$ and $Q'_m := \phi_\ell(Q_m)$. Output $ek_\ell := (E'_m, P'_m, Q'_m)$.[9]

[9] Correctly speaking, this algorithm outputs $(P'_m, Q'_m, R'_m := P'_m - Q'_m)$ and omits E'_m. We can reconstruct E'_m from P'_m, Q'_m, and R'_m.

- $\mathsf{isoex}_\ell(ek_m, dk_\ell)$ with $(m, \ell) = (2, 3)$ or $(3, 2)$: On input $ek_m = (E'_\ell, P'_\ell, Q'_\ell)$ and $dk_\ell \in [0, \ell^{e_\ell})$, compute $S := P'_\ell + [dk_\ell]Q'_\ell$ and compute $E''_\ell := E'_\ell/\langle S\rangle = E'_\ell/\langle \phi_m(P_\ell + [dk_\ell]Q_\ell)\rangle$. Compute j_ℓ as the j-invariant of E''_ℓ.
- $\mathsf{Gen}(pp)$: Choose $dk_3 \leftarrow [0, 3^{e_3})$ and $ek_3 := \mathsf{isogen}_3(dk_3)$. Output ek_3 and dk_3.
- $\mathsf{Enc}(ek_3, \mu)$: Choose $dk_2 \leftarrow [0, 2^{e_2})$ and $c_2 := \mathsf{isogen}_2(dk_2)$. Compute $j := \mathsf{isoex}_2(ek_3, dk_2)$. Compute $z := \mathsf{L}(j) \oplus \mu$. Output (c_2, z).
- $\mathsf{Dec}(dk_3, (c_2, z))$: Compute $j' := \mathsf{isoex}_3(c_2, dk_3)$ and output $\mu' := z \oplus \mathsf{L}(j')$.

SIKE applies $\mathsf{FO}^{\not\perp}$ to SIKE-PKE, where $\mathsf{G} = \mathsf{SHAKE256}_{e_2}$ and $\mathsf{H} = \mathsf{SHAKE256}_k$, and defined as in Figure 27.

$\mathsf{Gen}(1^\kappa)$	$\overline{\mathsf{Enc}}(ek)$	$\overline{\mathsf{Dec}}(\overline{dk}, (c_2, z))$, where $\overline{dk} = (dk, ek, s)$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$	$\mu \leftarrow \{0,1\}^n$	$\mu' := \mathsf{Dec}(dk, (c_2, z))$
$s \leftarrow \{0,1\}^n$	$r := \mathsf{G}(\mu, ek)$	$r' := \mathsf{G}(\mu', ek)$
$\overline{dk} := (dk, ek, s)$	$(c_2, z) := \mathsf{Enc}(ek, \mu; r)$	$c'_2 := \mathsf{isogen}_2(r')$
return $(ek, \overline{dk})$	$K := \mathsf{H}(\mu, c_2, z)$	**if** $c_2 \neq c'_2$, **then return** $K := \mathsf{H}(s, c_2, z)$
	return $((c_2, z), K)$	**else return** $K := \mathsf{H}(\mu', c_2, z)$

Fig. 27. SIKE

Remark S.1. SIKE's $\overline{\mathsf{Dec}}$ performs the test $c_2 = c'_2$ but omits the test $z = z'$. Since Dec retrieves $\mu' := z \oplus k$ *deterministically*, we do not need to check the equality of z and z'.

S.2 Properties of SIKE-PKE and SIKE-DPKE

Although we can invoke theorems on $\mathsf{FO}^{\not\perp}$ by Grubbs et al. [GMP21a] to show SIKE's anonymity and collision-freeness, we take another way to show SIKE's anonymity.

Assumptions: The security of SIKE is related to the following two variants of the Diffie-Hellman assumption:

Definition S.1 (Supersingular Computational Diffie-Hellman (SSCDH) Assumption [JD11], adapted). *Let $\phi_3 : E \to E'_2$ be an isogeny whose kernel is equal to $\langle P_3 + [dk_3]Q_3\rangle$, where $dk_3 \leftarrow [0, 3^{e_3})$. Let $\phi_2 : E \to E'_3$ be an isogeny whose kernel is equal to $\langle P_2 + [dk_2]Q_2\rangle$, where $dk_2 \leftarrow [0, 2^{e_2})$.*
For any QPT adversary, given the curves E'_2 and E'_3 and the points $\phi_3(P_2)$, $\phi_3(Q_2)$, $\phi_2(P_3)$, and $\phi_2(Q_3)$, finding the j-invariant of $E/\langle P_3 + [dk_3]Q_3, P_2 + [dk_2]Q_2\rangle$ is hard.

Definition S.2 (Supersingular Decisional Diffie-Hellman (SSDDH) Assumption [JD11], adapted). *For any QPT adversary, given a tuple, it is hard to determine which distribution of the following two distributions generates the tuple:*

- $(E'_2, \phi_3(P_2), \phi_3(Q_2), E'_3, \phi_2(P_3), \phi_2(Q_3), E_{23})$, *where $E'_2, \phi_3(P_2), \phi_3(Q_2), E'_3, \phi_2(P_3), \phi_2(Q_3)$ are as in the SSCDH assumption and*
$$E_{23} \simeq E/\langle P_3 + [dk_3]Q_3, P_2 + [dk_2]Q_2\rangle.$$
- $(E'_2, \phi_3(P_2), \phi_3(Q_2), E'_3, \phi_2(P_3), \phi_2(Q_3), E_c)$, *where $E'_2, \phi_3(P_2), \phi_3(Q_2), E'_3, \phi_2(P_3), \phi_2(Q_3)$ are as in the SSCDH assumption and*
$$E_c \simeq E/\langle P_3 + [dk'_3]Q_3, P_2 + [dk'_2]Q_2\rangle,$$
where $dk'_3 \leftarrow [0, 3^{e_3})$ and $dk'_2 \leftarrow [0, 2^{e_2})$.

Security of SIKE-PKE: One can show the IND-CPA security of the underlying PKE of SIKE by assuming the SSDDH assumption and the entropy-smoothing property of L [10] as that in [JD11].

Lemma S.1. *Assume that the SSDDH assumption holds and L is entropy-smoothing. Then, SIKE-PKE is* IND-CPA*-secure (and* OW-CPA*-secure).*

[10] We borrow the notation from [FNP14]. We say a family of hash functions $\mathfrak{H} = \{H : X \to Y\}$ is *entropy smoothing* [IZ89] if for any (Q)PPT adversary, it is hard to distinguish $(H, H(x))$ with (H, y), where $H \leftarrow \mathfrak{H}$, $x \leftarrow X$, and $y \leftarrow Y$.

For ciphertext indistinguishablity, we construct a simulator $\mathcal{S}$ as follows: 1) sample $dk_2 \leftarrow [0, 2^{e_2})$ and compute $c_2 = (E'_3, P'_3, Q'_3) := \mathsf{isogen}_2(dk_2)$; 2) sample $z \leftarrow \{0,1\}^n$; 3) output (c_2, z).
We can show that SIKE-PKE ciphertext indistinguishable with no assumptions:

Lemma S.2. *SIKE-PKE is ciphertext indistinguishable with $\mathcal{S}$.*

Notice that we can remove the assumption on L's property.

Proof (Proof Sketch). We consider two games Game_0 and Game_1.
- Game_0: In this game the challenge ciphertext is computes as

$$\mu \leftarrow \{0,1\}^{256}; dk_2 \leftarrow [0, 2^{e_2}); c_2 := \mathsf{isogen}_2(dk_2); j \leftarrow \mathsf{isoex}_2(ek_3, dk_2); z := \mathsf{L}(j) \oplus \mu; \text{ return } (c_2, z).$$

- Game_1: In this game the challenge ciphertext is computes as

$$dk_2 \leftarrow [0, 2^{e_2}); c_2 := \mathsf{isogen}_2(dk_2); z \leftarrow \{0,1\}^{256}; \text{ return } (c_2, z).$$

Game_0 and Game_1 are equivalent since μ in Game_0 and z in Game_1 are uniformly at random. □

Security of SIKE-DPKE: We next consider SIKE-DPKE obtained by applying T to SIKE-PKE.

Lemma S.3. *Assume that the SSDDH assumption holds and L is entropy-smoothing. Then, SIKE-DPKE is strongly disjoint-simulatable with $\mathcal{S}$.*

Proof. Statistical disjointess follows from the fact that $|\mathcal{S}(1^\kappa)| \approx 2^{e_2} \cdot 2^n$ and $|\mathsf{Enc}'(ek, \mathcal{M})| \leq 2^n$. We can show ciphertext indistinguishability by invoking Theorem D.1 since SIKE-PKE is ciphertext-indistinguishable (Lemma S.2) and oneway (Lemma S.1). In addition, the simulator $\mathcal{S}$ does not take ek as input. Thus, SIKE-DPKE is strongly disjoint-simulatable with $\mathcal{S}$. □

We next consider SIKE-DPKE's collision-freeness.

Lemma S.4. *Let ϵ_3 be a probability that $ek_3^0 \neq ek_3^1$ holds for two keys (ek_3^0, dk_3^0) and (ek_3^1, dk_3^1) generated randomly and independently. Let ϵ_2 be a probability that an efficient quantum adversary, given (ek_3^0, dk_3^0) and (ek_3^1, dk_3^1), finds μ such that $\mathsf{isogen}_2(\mathsf{G}(\mu, ek_3^0)) = \mathsf{isogen}_2(\mathsf{G}(\mu, ek_3^1))$. Suppose that $\epsilon := \epsilon_3 + \epsilon_2$ is negligible. Then, SIKE-DPKE is* XCFR*-secure.*

Proof. The adversary against the XCFR security is given two encryption keys ek_3^0 and ek_3^1 with their decryption keys dk_3^0 and dk_3^1 and outputs (c_2, z). If the adversary wins, then there is μ such that $dk_2^0 = \mathsf{G}(\mu, ek_3^0)$, $dk_2^1 = \mathsf{G}(\mu, ek_3^1)$, $c_2 = \mathsf{isogen}_2(dk_2^0) = \mathsf{isogen}_2(dk_2^1)$, and $z = \mu \oplus \mathsf{L}(j^0) = \mu \oplus \mathsf{L}(j^1)$, where $j^i := \mathsf{isoex}_2(ek_3^i, dk_2^i)$. We consider the following cases:
- Case 1 ($ek_3^0 = ek_3^1$): We assume that this rarely occurs by the correct choices of $dk_3^0, dk_3^1 \leftarrow [0, 3^{\ell_3})$ and the probability is at most ϵ_3.
- Case 2 ($ek_3^0 \neq ek_3^1$ and $dk_2^0 = dk_2^1$): This violates the collision resistance property of the quantum random oracle G since $(\mu, ek_3^0) \neq (\mu, ek_3^1)$ and $\mathsf{G}(\mu, ek_3^0) = \mathsf{G}(\mu, ek_3^1)$.
- Case 3 ($ek_3^0 \neq ek_3^1$, $dk_2^0 \neq dk_2^1$, and $\mathsf{isogen}_2(dk_2^0) = \mathsf{isogen}_2(dk_2^1)$): We assume that it is hard to find μ such that $dk_2^0 = \mathsf{G}(\mu, ek_3^0)$ and $dk_2^1 = \mathsf{G}(\mu, ek_3^1)$ and the probability is at most ϵ_2.

Thus, in any cases, the winning probability of the adversary is negligible and we conclude the proof. □

S.3 Properties of SIKE

Combining SIKE-DPKE's strong disjoint-simulatability and XCFR security with previous theorems on $\mathsf{U}^{\not\perp}$, we obtain the following theorems.

Theorem S.1. *Suppose that the SSDDH assumption holds and L is entropy-smoothing. Then, SIKE is* SPR-CCA*-secure and* SSMT-CCA*-secure in the QROM.*

Proof. Under the SSDDH assumption and the assumption on L, SIKE-DPKE is strongly disjoint-simulatable (Lemma S.3). Applying Theorem E.2 and Theorem E.3, we obtain the theorem. □

Theorem S.2. *Let ϵ_3 be a probability that $ek_3^0 \neq ek_3^1$ holds for two keys (ek_3^0, dk_3^0) and (ek_3^1, dk_3^1) generated randomly and independently. Let ϵ_2 be a probability that an efficient quantum adversary, given (ek_3^0, dk_3^0) and (ek_3^1, dk_3^1), finds μ such that $\mathsf{isogen}_2(\mathsf{G}(\mu, ek_3^0)) = \mathsf{isogen}_2(\mathsf{G}(\mu, ek_3^1))$. Suppose that $\epsilon := \epsilon_3 + \epsilon_2$ is negligible. Then, SIKE is* SCFR-CCA*-secure in the QROM.*

Proof. Under the hypothesis, SIKE-DPKE is XCFR-secure (Lemma N.6). Applying Theorem E.4, we have that SIKE is SCFR-CCA-secure in the QROM. □

Theorem S.3. *Suppose that the SSDDH assumption holds and* L *is entropy-smoothing. Then, SIKE is* ANON-CCA-*secure in the QROM.*

Proof. Due to Theorem S.1, under the hypothesis, SIKE is SPR-CCA-secure in the QROM. Thus, applying Theorem 2.5, we have that, under those assumptions, SIKE is ANON-CCA-secure in the QROM. □

Theorem S.4. *Let* ϵ_3 *be a probability that* $ek_3^0 \neq ek_3^1$ *holds for two keys* (ek_3^0, dk_3^0) *and* (ek_3^1, dk_3^1) *generated randomly and independently. Let* ϵ_2 *be a probability that an efficient quantum adversary, given* (ek_3^0, dk_3^0) *and* (ek_3^1, dk_3^1)*, finds* μ *such that* $\mathsf{isogen}_2(\mathsf{G}(\mu, ek_3^0)) = \mathsf{isogen}_2(\mathsf{G}(\mu, ek_3^1))$*. Suppose that* $\epsilon := \epsilon_3 + \epsilon_2$ *is negligible. Suppose that the SSDDH assumption holds and* L *is entropy-smoothing. Then, SIKE leads to* ANON-CCA-*secure and* SROB-CCA-*secure hybrid PKE in the QROM, combined with* SPR-OTCCA-*secure and* FROB-*secure DEM.*

Proof. Due to Theorem S.1, under the SSDDH assumption and the assumption on L, SIKE is SPR-CCA-secure and SSMT-CCA-secure in the QROM. Thus, combining SIKE with SPR-OTCCA-secure DEM, we obtain a SPR-CCA-secure hybrid PKE in the QROM (Theorem 3.2).
Moreover, SIKE is SCFR-CCA-secure in the QROM (Theorem S.2) under the hypothesis on ϵ. Thus, if DEM is FROB-secure, then the hybrid PKE is SROB-CCA-secure (Theorem 2.2). □