SO-CCA Secure PKE in the Quantum Random Oracle Model or the Quantum Ideal Cipher Model*

Shingo Sato† Junji Shikata†

May 20, 2022

Abstract

Selective opening (SO) security is one of the most important security notions of public key encryption (PKE) in a multi-user setting. Even though messages and random coins used in some ciphertexts are leaked, SO security guarantees the confidentiality of the other ciphertexts. Actually, it is shown that there exist PKE schemes which meet the standard security such as indistinguishability against chosen ciphertext attacks (IND-CCA security) but do not meet SO security against chosen ciphertext attacks. Hence, it is important to consider SO security in the multi-user setting. On the other hand, many researchers have studied cryptosystems in the security model where adversaries can submit quantum superposition queries (i.e., quantum queries) to oracles. In particular, IND-CCA secure PKE and KEM schemes in the quantum random oracle model have been intensively studied so far.

In this paper, we show that two kinds of constructions of hybrid encryption schemes meet simulation-based SO security against chosen ciphertext attacks (SIM-SO-CCA security) in the quantum random oracle model or the quantum ideal cipher model. The first scheme is constructed from any IND-CCA secure KEM and any simulatable data encapsulation mechanism (DEM). The second one is constructed from any IND-CCA secure KEM based on Fujisaki-Okamoto transformation and any strongly unforgeable message authentication code (MAC). We can apply any IND-CCA secure KEM scheme to the first one if the underlying DEM scheme meets simulatability, whereas we can apply strongly unforgeable MAC to the second one if the underlying KEM is based on Fujisaki-Okamoto transformation.

1 Introduction

1.1 Background

Security against chosen ciphertext attacks, which is called CCA security, has been studied as one of the most important security notions of public key encryption (PKE). However, as the security of PKE in a multi-user setting, security against selective opening attacks, which is called SO security, was introduced by Bellare, Hofheinz and Yilek in [4]. SO security guarantees that even though an adversary gets secret information such as messages and random coins used in several ciphertexts, the other ciphertexts meet confidentiality. In a real world, there exist such situations where secret information of some ciphertexts is leaked because of factors except for cryptosystems. Furthermore, it is shown that there exist PKE schemes which meet CCA security but do not satisfy SO security [3, 23, 22]. Hence, it is important to consider SO security. In particular, several SO secure PKE schemes have been proposed so far: PKE [4, 16, 17, 21], hybrid encryption [14, 33, 18, 34], identity-based encryption [7, 31], and lattice-based PKE [11, 32]. SO security is roughly classified as simulation-based SO (SIM-SO) security and indistinguishability-based SO (IND-SO) security. In this paper, we consider SIM-SO security against chosen ciphertext attacks called $\mathsf{SIM\text{-}SO\text{-}CCA}$ security, since it seems that it is harder to achieve SIM-SO security [8, 21] and many works have aimed at

*This paper is the full version of our paper which appears at IMACC 2019.
†Yokohama National University, Yokohama, Japan. sato-shingo-zk@ynu.ac.jp, shikata-junji-rb@ynu.ac.jp.

proposing SIM-SO-CCA secure PKE schemes [14, 17, 33, 18, 21, 32, 34]. Hence, it is natural to consider SIM-SO-CCA security as our goal in the multi-user setting.

On the other hand, we consider the model where adversaries can submit quantum superposition queries (i.e., quantum queries) to oracles. In particular, secure cryptosystems in the quantum random oracle model (QROM) have been intensively studied. The QROM, whose notion was introduced by [9], is a model where any users can issue quantum queries to random oracles. There exist several works related to PKE schemes in the QROM: PKE [9, 36], key encapsulation mechanism (KEM) [20, 35, 27, 25, 28, 29], digital signatures (DSs) [10, 30, 19, 13]. Moreover, almost all PKE/KEM and DS schemes submitted to the post-quantum cryptography standardization process of NIST (National Institute of Standards and Technology) satisfy securities in the QROM. Therefore, it is interesting and important to consider secure PKE schemes in the QROM. PKE/KEM schemes in the QROM that have already been proposed are summarized as follows. A PKE scheme constructed from trapdoor permutations meets indistinguishability against chosen ciphertext attacks (called IND-CCA security) in the QROM [9]. [36] proved that Fujisaki-Okamoto (FO) transformation [15] and OAEP [6] with additional hash satisfy IND-CCA security in the QROM. [20] analyzed FO-based KEM schemes. Based on the proof technique of [9], [35] proposed a tightly secure KEM scheme starting from any disjunct-simulatable deterministic PKE scheme. [27] revisited FO-based KEM schemes with implicit rejection and proved that they meet tighter IND-CCA security without additional hash. [28] proposed IND-CCA secure KEM schemes with explicit rejection. [25] gave a tighter security proof for the KEM scheme based on FO transformation by utilizing the proof techniques proposed in [1]. [29] also gave tighter security proofs for generic constructions of KEM by utilizing the techniques in [1].

1.2 Our Contribution

Our goal is to present SIM-SO-CCA secure PKE schemes obtained from KEM schemes in the QROM or the quantum ideal cipher model (QICM). Our main motivation is to transform any PKE/KEM schemes submitted to the post-quantum cryptography standardization to SIM-SO-CCA secure PKE without loss of efficiency in terms of key-size, ciphertext-size, and time-complexity.

In the classical random oracle model, classical ideal cipher model, or the standard model (i.e., the model without random oracles and ideal ciphers), several SIM-SO-CCA secure PKE schemes constructed from KEM schemes have been studied. Liu and Paterson proposed a SIM-SO-CCA secure PKE scheme constructed from a KEM scheme secure against tailored constrained chosen ciphertext attacks and a strengthened cross authentication code (XAC) [33]. Heuer et al. proposed a SIM-SO-CCA secure construction by combining KEM secure against plaintext checking attacks and a message authentication codes (MAC) [17]. Heuer and Poettering proved that a PKE scheme in the KEM/DEM framework meets SIM-SO-CCA security in the classical ideal cipher model if the underlying KEM scheme satisfies IND-CCA security, and the underlying DEM scheme satisfies both simulatability and one-time integrity of chosen ciphertext attacks, which is called OT-INT-CTXT security [18]. Lyu et al. proposed a tightly secure PKE starting from any KEM scheme meeting both of security notions multi-encapsulation pseudorandom security and random encapsulation rejection security, and any strengthened XAC [34]. Table 1 shows the underlying primitives and security models of these existing constructions.

In the QROM or QICM, how to construct PKE schemes meeting SIM-SO-CCA security is not obvious because of the following reason: In the classical random oracle model or classical ideal cipher model, the security proofs of the existing schemes [33, 18] utilize the lists of query-response pairs of random oracles or ideal ciphers. In the QROM and QICM, we cannot use such lists, since it is impossible to record query-response pairs in principle, because of the quantum no-cloning theorem. Hence, it is worth to consider secure PKE schemes in the models where quantum queries are issued.

Notice that as for the SIM-SO-CCA secure PKE schemes obtained from KEM schemes in the standard model [33, 34], the ciphertext-size and time-complexity of these encryption and decryption algorithms linearly depend on the bit-length of a message. Since we are aiming at constructing practical PKE schemes, we do not focus on these schemes in this paper, because of the lack of efficiency in terms of ciphertext-size and time-complexity.

Table 1: $\mathsf{SIM\text{-}SO\text{-}CCA}$ secure PKE constructed from KEM schemes

Scheme	Underlying Primitives	Standard Model ?
LP [33]	IND-tCCCA secure KEM, XAC	✓
HJKS [17]	OW-PCA secure KEM, sUF-OT-CMA secure MAC	Random Oracle Model
HP [18]	IND-CCA secure KEM, Simulatable DEM	Ideal Cipher Model
LLHG [34]	mPR-CCCA and RER secure KEM, XAC	✓
Our Scheme PKE_1^{hy}	IND-CCA secure KEM, Simulatable DEM	Quantum Ideal Cipher Model
Our Scheme PKE_2^{hy}	FO-based KEM (from IND-CPA secure PKE), sUF-OT-CMA secure MAC	Quantum Random Oracle Model

IND-tCCCA means indistinguishability against tailored constrained chosen ciphertext attacks. IND-PCA means indistinguishability agasint plaintext checking attacks. mPR-CCCA means multi-encapsulation pseudorandom security against constrained chosen ciphertext attacks. RER means random encapsulation rejection security. XAC means (strengthened) cross authentication code. IND-CPA means indistinguishability against chosen message attacks. FO-based KEM means $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{QFO}^{\not\perp}$, and $\mathsf{QFO}_m^{\not\perp}$. Standard model denotes the security model without random oracles and ideal ciphers.

In this paper, we propose two constructions of $\mathsf{SIM\text{-}SO\text{-}CCA}$ secure PKE schemes from KEM schemes and symmetric key encryption (SKE) schemes. The details are as follows:

1. The first scheme PKE_1^{hy} is the KEM/DEM scheme [12]. We prove that this scheme meets $\mathsf{SIM\text{-}SO\text{-}CCA}$ security in the QICM if the underlying KEM scheme satisfies $\mathsf{IND\text{-}CCA}$ security, and the underlying DEM scheme satisfies both simulatability [18] and one-time integrity of chosen ciphertext attacks ($\mathsf{OT\text{-}INT\text{-}CTXT}$ security) [5]. The advantage of this scheme is that we can apply any $\mathsf{IND\text{-}CCA}$ secure KEM scheme such as any PKE/KEM schemes submitted to the post-quantum cryptography standardization, and we can obtain a $\mathsf{SIM\text{-}SO\text{-}CCA}$ secure PKE schemes in the QICM.
2. The second one PKE_2^{hy} is a concrete scheme constructed from any FO-based KEM scheme such as $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{QFO}^{\not\perp}$, and $\mathsf{QFO}_m^{\not\perp}$, which are categorized in [20], and any MAC meeting strong unforgeability against one-time chosen message attacks called $\mathsf{sUF\text{-}OT\text{-}CMA}$ security. The underlying KEM scheme is FO-based KEM with implicit rejection. That is, these schemes output a random key which is not encapsulated if a given ciphertext is invalid. We require that the underlying PKE scheme in $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{QFO}^{\not\perp}$, or $\mathsf{QFO}_m^{\not\perp}$ is injective and satisfies indistinguishability against chosen plaintext attacks called $\mathsf{IND\text{-}CPA}$ security. In addition, almost all KEM schemes submitted to the NIST post-quantum cryptography standardization are classified as $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{QFO}^{\not\perp}$, or $\mathsf{QFO}_m^{\not\perp}$. Hence, the advantage of PKE_2^{hy} is that a lot of PKE/KEM schemes submitted to the post-quantum standardization can satisfy $\mathsf{SIM\text{-}SO\text{-}CCA}$ security without demanding any special property such as simulatability for the underlying SKE.

The difference between PKE_1^{hy} and PKE_2^{hy} is given as follows:

- Any $\mathsf{IND\text{-}CCA}$ secure KEM scheme can be applied to PKE_1^{hy} while a particular KEM scheme (i.e., $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{QFO}^{\not\perp}$, or $\mathsf{QFO}_m^{\not\perp}$) can be applied to PKE_2^{hy}.
- PKE_1^{hy} requires that the underlying DEM scheme satisfies a special property such as simulatability[1] while PKE_2^{hy} does not require that the underlying MAC satisfies such a special property.

[1]As far as we know, there is no simulatable DEM scheme in the quantum ideal cipher model.

In Sections 3.1 and 3.2, we describe concrete primitives which can be applied to PKE_1^{hy} and PKE_2^{hy}, respectively.

2 Preliminaries

For a positive integer n, let $[n]$ be a set $\{1, 2, \ldots, n\}$. For a set $\mathcal{X}$, let $|\mathcal{X}|$ be the number of elements in $\mathcal{X}$ (the size of $\mathcal{X}$). For a set $\mathcal{X}$ and an element $x \in \mathcal{X}$, we write $|x|$ as the bit-length of x. We write that a function $\epsilon = \epsilon(\lambda)$ is negligible, if for a large enough λ and all polynomial $p(\lambda)$, it holds that $\epsilon(\lambda) < 1/p(\lambda)$. For a randomized algorithm A and any input x of A, $\mathsf{A}(x; r)$ denotes a deterministic algorithm, where r is a random coin used in A. In this paper, probabilistic polynomial-time is abbreviated as PPT, and quantum polynomial-time is abbreviated as QPT.

2.1 Quantum Computaions

We define an n-qubit state as $|\varphi\rangle = \sum_{x\in\{0,1\}^n} \psi_x |x\rangle$ with a basis $\{|x\rangle\}_{x\in\{0,1\}^n}$ and amplitudes $\psi_x \in \mathbb{C}$ such that $\sum_{x\in\{0,1\}^n} |\psi_x|^2 = 1$. If $|\varphi\rangle = \sum_{x\in\{0,1\}^n} \psi_x |x\rangle$ is measured in the computational basis, $|\varphi\rangle$ will become a classical state $|x\rangle$ with probability $|\psi_x|^2$. For a quantum oracle $\mathsf{O} : \mathcal{X} \to \mathcal{Y}$, submitting a quantum query $\sum_{x\in\mathcal{X},y\in\mathcal{Y}} \psi_{x,y}|x, y\rangle$ to O (quantum access to O) is written as

$$\sum_{x\in\mathcal{X},y\in\mathcal{Y}} \psi_{x,y}|x, y\rangle \mapsto \sum_{x\in\mathcal{X},y\in\mathcal{Y}} \psi_{x,y}|x, y \oplus \mathsf{O}(x)\rangle.$$

The quantum random oracle model (QROM) is defined as the model where a quantum adversary can submit quantum queries to random oracles. The quantum ideal (block) cipher model (QICM) which was introduced in [24] is defined as follows: A block cipher with a key space $\mathcal{K}$ and a message space $\mathcal{X}$ is defined as a mapping $E : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ which is a permutation over $\mathcal{X}$ for any key in $\mathcal{K}$. In the QICM, a quantum adversary is allowed to issue quantum queries to oracles $E^+ : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ and $E^- : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ such that for any $\mathsf{k} \in \mathcal{K}$ and any $x, y \in \mathcal{X}$, the response of $E^-(\mathsf{k}, y)$ is x meeting $E^+(\mathsf{k}, x) = y$. In this paper, QROM (resp. QICM) denote the security model where quantum adversaries are allowed to issue quantum queries to random oracles (resp. ideal ciphers), but submit only classical queries to the other oracles.

Semi-Classical Oracle. We describe semi-classical oracle which was introduced in [1] and utilize this oracle for our security proofs. We consider quantum access to an oracle with a domain $\mathcal{X}$. A semi-classical oracle O_S^{SC} for a subset $S \subseteq \mathcal{X}$ uses an indicator function $f_S : \mathcal{X} \to \{0, 1\}$ with the subset S which evaluates 1 if $x \in S$ is given, and evaluates 0 otherwise. When O_S^{SC} is given a quantum query $\sum_{x\in\mathcal{X}} \psi_x|x\rangle|0\rangle$ with the input register Q and the output register R, it maps

$$\sum_{x\in\mathcal{X}} \psi_{x,z}|x\rangle|0\rangle \mapsto \sum_{x\in\mathcal{X}} \psi_x|x\rangle|f_S(x)\rangle,$$

and measures the register R. Then, the quantum query $\sum_{x\in\mathcal{X}} \psi_x|x\rangle|0\rangle$ collapses to either $\sum_{x\in\mathcal{X}\setminus S} \psi'_x|x\rangle|0\rangle$ or $\sum_{x\in S} \psi'_x|x\rangle|1\rangle$. Let Find be the event that O_S^{SC} returns $\sum_{x\in S} \psi'_x|x\rangle|1\rangle$ for a quantum query $\sum_{x\in S} \psi_x|x\rangle$. For a quantum oracle H with domain $\mathcal{X}$ and a subset $S \subseteq \mathcal{X}$, let $\mathsf{H}\setminus S$ be an oracle which first queries O_S^{SC} and then H.

By using semi-classical oracles, [1] proved the following propositions. We notice that for query depth d and the number of queries q, we use q such that $q \geq d$ in the same way as [25, Theorem 2.8].

Proposition 1 ([1, Theorem 1])**.** *Let $S \subseteq \mathcal{X}$ be random. Let $\mathsf{H} : \mathcal{X} \to \mathcal{Y}$, $\mathsf{G} : \mathcal{X} \to \mathcal{Y}$ be random functions such that $\mathsf{H}(x) = \mathsf{G}(x)$ for all $x \in \mathcal{X}\setminus S$, and let z be a random bit-string (S, H, G and z may have an arbitrary joint distribution). Let A be any quantum algorithm issuing at most q quantum queries to oracles. Then, it holds that*

$$\left|\Pr[1 \leftarrow \mathsf{A}^{\mathsf{H}}(z)] - \Pr[1 \leftarrow \mathsf{A}^{\mathsf{G}}(z)]\right| \leq 2\sqrt{q \cdot \Pr[\mathsf{Find} \mid 1 \leftarrow \mathsf{A}^{\mathsf{H}\setminus S}(z)]}.$$

Proposition 2 ([1, Corollary 1]). *Let A be any quantum algorithm issuing at most q quantum queries to a semi-classical oracle with domain $\mathcal{X}$. Suppose that $S \subseteq \mathcal{X}$ and $z \in \{0,1\}^*$ are independent. Then, it holds that* $\Pr[\mathsf{Find} \mid \mathsf{A}^{O_S^{SC}}(z)] \le 4q \cdot P_{\max}$, *where* $P_{\max} = \max_{x \in X} \Pr[x \in S]$.

2.2 Definitions of Cryptographic Primitives

2.2.1 Public Key Encryption

A public key encryption (PKE) scheme consists of three polynomial-time algorithms ($\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec}$): For a security parameter λ, let $\mathcal{M} = \mathcal{M}(\lambda)$ be a message space, and let $\mathcal{CT} = \mathcal{CT}(\lambda)$ be a ciphertext space.

- **Key Generation** KGen is a randomized algorithm which, on input a security parameter 1^λ, outputs a public key pk and a secret key sk.
- **Encryption** Enc is a randomized or deterministic algorithm which, on input a public key pk and a message $\mathsf{m} \in \mathcal{M}$, outputs a ciphertext ct.
- **Decryption** Dec is a deterministic algorithm which, on input a secret key sk and a ciphertext ct, outputs a message $\mathsf{m} \in \mathcal{M}$ or an invalid symbol $\perp$.

Definition 1 (Correctness). *A PKE scheme* $\mathsf{PKE} = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *is* δ-$\mathsf{correct}$ *if*

$$\mathbf{E}\left[\max_{\mathsf{m} \in \mathcal{M}} \Pr[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, \mathsf{m})) \neq \mathsf{m}] \mid (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)\right] \le \delta.$$

Then, δ *denotes the decryption failure probability of* PKE. *In addition,* PKE *is* $\mathsf{correct}$ *if* $\delta = 0$.

We describe two security notions of PKE: *indistinguishability against chosen message attacks* (denoted by $\mathsf{IND\text{-}CPA}$ security) and *simulation-based selective opening security against chosen ciphertext attacks* (denoted by $\mathsf{SIM\text{-}SO\text{-}CCA}$ security).

Definition 2 ($\mathsf{IND\text{-}CPA}$ security). *A PKE scheme* $\mathsf{PKE} = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *satisfies* $\mathsf{IND\text{-}CPA}$ *security if for any PPT adversary* A *against* PKE, *the advantage* $\mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{A}}(\lambda) := |2 \cdot \Pr[\mathsf{A} \textit{ wins}] - 1|$ *is negligible in* λ, *where* $[\mathsf{A} \textit{ wins}]$ *is the event that* A *wins in the following game:*

Setup: *A challenger generates* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(\lambda)$.

Challenge: *When* A *submits* $(\mathsf{m}_0, \mathsf{m}_1)$ *such that* $|\mathsf{m}_0| = |\mathsf{m}_1|$, *the challenger chooses* $b \xleftarrow{\$} \{0,1\}$ *and returns* $\mathsf{ct}^* \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathsf{m}_b)$.

Output: A *outputs the guessing bit* $b' \in \{0,1\}$. A *wins if* $b = b'$ *holds.*

Definition 3 ($\mathsf{SIM\text{-}SO\text{-}CCA}$ security). *A PKE scheme* $\mathsf{PKE} = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *satisfies* $\mathsf{SIM\text{-}SO\text{-}CCA}$ *security if for any PPT algorithms* $\mathsf{A} = (\mathsf{A}_0, \mathsf{A}_1)$, $\mathsf{S} = (\mathsf{S}_0, \mathsf{S}_1)$ *and any relation* R, *its advantage* $\mathsf{Adv}^{\text{sim-so-cca}}_{\mathsf{PKE},\mathsf{A},\mathsf{S},R}(\lambda)$ *is negligible in* λ. $\mathsf{Adv}^{\text{sim-so-cca}}_{\mathsf{PKE},\mathsf{A},\mathsf{S},R}(\lambda)$ *is defined as follows:*

$$\mathsf{Adv}^{\text{sim-so-cca}}_{\mathsf{PKE},\mathsf{A},\mathsf{S},R}(\lambda) := \left|\Pr[\mathsf{Expt}^{\text{real-so-cca}}_{\mathsf{PKE},\mathsf{A}}(\lambda) \to 1] - \Pr[\mathsf{Expt}^{\text{ideal-so-cca}}_{\mathsf{PKE},\mathsf{S}}(\lambda) \to 1]\right|,$$

where the two experiments $\mathsf{Expt}^{\text{real-so-cca}}_{\mathsf{PKE},\mathsf{A}}(\lambda)$ *and* $\mathsf{Expt}^{\text{ideal-so-cca}}_{\mathsf{PKE},\mathsf{S}}(\lambda)$ *are defined in Figure 1.*

2.2.2 Key Encapsulation Mechanism

A key encapsulation mechanism (KEM) scheme consists of three polynomial-time algorithms ($\mathsf{KGen}, \mathsf{Encaps}, \mathsf{Decaps}$) with a key space $\mathcal{K} = \mathcal{K}(\lambda)$ for a security parameter λ.

Key Generation KGen is a randomized algorithm which, on input a security parameter 1^λ, outputs a public key pk and a secret key sk.

$\underline{\mathsf{Expt}^{\text{real-so-cca}}_{\mathsf{PKE},\mathsf{A}}(\lambda)}$

$I \leftarrow \emptyset$
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$
$(\mathcal{M}_\mathrm{D}, \mathsf{st}) \leftarrow \mathsf{A}_0^{\mathsf{DEC}}(\mathsf{pk})$
$(\mathsf{m}_1, \ldots, \mathsf{m}_n) \xleftarrow{\$} \mathcal{M}_\mathrm{D}$
$(r_1, \ldots, r_n) \xleftarrow{\$} \mathcal{R}$
$\forall i \in [n], \mathsf{ct}_i = \mathsf{Enc}(\mathsf{pk}, \mathsf{m}_i; r_i)$
$out \leftarrow \mathsf{A}_1^{\mathsf{OPEN},\mathsf{DEC}}(\mathsf{st}, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)$
return $R(\mathcal{M}_\mathcal{D}, \mathsf{m}_1, \ldots, \mathsf{m}_n, I, out)$

$\underline{\mathsf{OPEN}(i)}$

$I \leftarrow I \cup \{i\}$
return (m_i, r_i)

$\underline{\mathsf{DEC}(\mathsf{ct})}$

if $\mathsf{ct} \in \{\mathsf{ct}_i\}_{i\in[n]}$, **return** $\perp$
$\mathsf{m} \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$
return $\mathsf{m} \in \mathcal{M} \cup \{\perp\}$

$\underline{\mathsf{Expt}^{\text{ideal-so-cca}}_{\mathsf{PKE},\mathsf{S}}(\lambda)}$

$I \leftarrow \emptyset$
$(\mathcal{M}_\mathrm{D}, \mathsf{st}) \leftarrow \mathsf{S}_0(1^\lambda)$
$(\mathsf{m}_1, \ldots, \mathsf{m}_n) \xleftarrow{\$} \mathcal{M}_\mathrm{D}$
$out \leftarrow \mathsf{S}_1^{\mathsf{OPEN}}(\mathsf{st}, |\mathsf{m}_1|, \ldots, |\mathsf{m}_n|)$
return $R(\mathcal{M}_\mathcal{D}, \mathsf{m}_1, \ldots, \mathsf{m}_n, I, out)$

$\underline{\mathsf{OPEN}(i)}$

$I \leftarrow I \cup \{i\}$
return m_i

Figure 1: Experiments in REAL-SIM-SO-CCA and IDEAL-SIM-SO-CCA games

Encapsulation Encaps is a randomized algorithm which, on input a public key pk, outputs a ciphertext ct and a key $\mathsf{k} \in \mathcal{K}$.

Decapsulation Decaps is a deterministic algorithm which, on input a secret key sk and a ciphertext ct, outputs a key $\mathsf{k} \in \mathcal{K}$ or an invalid symbol $\perp$.

Then, we require a KEM scheme to be δ-correct with a negligible function δ for λ.

Definition 4 (Correctness)**.** *A KEM scheme* (KGen, Encaps, Decaps) *is* δ-correct *if for any* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$*, it holds that* $\mathsf{k} = \mathsf{Decaps}(\mathsf{sk}, \mathsf{ct})$ *with at least probability* $1 - \delta$*, where* $(\mathsf{ct}, \mathsf{k}) \leftarrow \mathsf{Encaps}(\mathsf{pk})$*.*

We describe a security notion of KEM: *indistinguishability against chosen ciphertext attacks* (denoted by IND-CCA security).

Definition 5 (IND-CCA security)**.** *A KEM scheme* $\mathsf{KEM} = (\mathsf{KGen}, \mathsf{Encaps}, \mathsf{Decaps})$ *satisfies* IND-CCA *security if for any PPT adversary* A *against* KEM*, the advantage* $\mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{A}}(\lambda) := |2 \cdot \Pr[\mathsf{A}\ wins] - 1|$ *is negligible in* λ*.* [A *wins*] *is the event that* A *wins in the following game:*

Setup: *A challenger generates* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(\lambda)$ *and sends* pk *to* A*.*

Oracle Access: A *is allowed to access the following oracles:*

- Challenge()*: Given a challenge request, the challenger computes* $(\mathsf{ct}^*, \mathsf{k}_0) \leftarrow \mathsf{Encaps}(\mathsf{pk})$ *and chooses* $\mathsf{k}_1 \in \mathcal{K}$ *uniformly at random. It returns* $(\mathsf{ct}^*, \mathsf{k}_b)$ *for* $b \xleftarrow{\$} \{0, 1\}$*.*
- DEC(ct)*: Given a ciphertext query* ct*, the decapsulation oracle* DEC(ct) *returns* $\mathsf{k}' \leftarrow \mathsf{Decaps}(\mathsf{sk}, \mathsf{ct}) \in \mathcal{K} \cup \{\perp\}$*.* A *is not allowed to submit* ct^* *to* $\mathsf{DEC}(\cdot)$*.*

Output: A *outputs the guessing bit* $b' \in \{0, 1\}$*.* A *wins if* $b = b'$ *holds.*

2.2.3 Data Encapsulation Mechanism

A data encapsulation mechanism (DEM) scheme consists of two polynomial-time algorithms (Enc, Dec) with a key space $\mathcal{K} = \mathcal{K}(\lambda)$ and a message space $\mathcal{M} = \mathcal{M}(\lambda)$ for a security parameter λ.

Encapsulation Enc is an algorithm which, on input a secret key $\mathsf{k} \in \mathcal{K}$ and a message $\mathsf{m} \in \mathcal{M}$, outputs a ciphertext ct.

Decryption Dec is a deterministic algorithm which, on input a secret key $\mathsf{k} \in \mathcal{K}$, a ciphertext ct, outputs a message $\mathsf{m} \in \mathcal{M}$ or an invalid symbol $\perp$.

We require that a DEM scheme satisfies correctness.

Definition 6 (Correctness)**.** *A DEM scheme* $(\mathsf{Enc}, \mathsf{Dec})$ *is* correct *if for any* $\mathsf{k} \in \mathcal{K}$ *and any* $\mathsf{m} \in \mathcal{M}$*, it holds that* $\mathsf{m} = \mathsf{Dec}(\mathsf{k}, \mathsf{ct})$*, where* $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{k}, \mathsf{m})$*.*

We describe a security notion of DEM: *one-time integrity of chosen ciphertext attacks* (denoted by OT-INT-CTXT security) [5].

Definition 7 (OT-INT-CTXT security)**.** *A DEM scheme* $\mathsf{DEM} = (\mathsf{Enc}, \mathsf{Dec})$ *satisfies* OT-INT-CTXT *security if for any PPT adversary* A *against* DEM*, the advantage* $\mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{A},\mathsf{DEM}}(\lambda) := \Pr[\mathsf{A} \text{ wins}]$ *is negligible in* λ*, where* $[\mathsf{A} \text{ wins}]$ *is the event that* A *wins in the following game:*

Setup: *A challenger chooses a key* $\mathsf{k} \in \mathcal{K}$ *uniformly at random, and sets* $\mathsf{win} \leftarrow 0$ *and* $C \leftarrow \emptyset$*.*

Oracle Access: A *is allowed to access the following oracles:*

- $\mathsf{ENC}(\mathsf{m})$*: If* $C \neq \emptyset$*, the encryption oracle* $\mathsf{ENC}(\mathsf{m})$ *returns* $\perp$*. Otherwise, it returns* $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{k}, \mathsf{m})$*, and sets* $C \leftarrow C \cup \{\mathsf{ct}\}$*.*
- $\mathsf{VRFY}(\mathsf{ct})$*: Given a ciphertext query* ct*, the verification oracle* $\mathsf{VRFY}(\mathsf{ct})$ *runs* $\mathsf{m}' \leftarrow \mathsf{Dec}(\mathsf{k}, \mathsf{m})$*. If* $\mathsf{m}' \neq \perp$ *and* $\mathsf{ct} \notin C$*, it sets* $\mathsf{win} \leftarrow 1$*. It returns* 1 *if* $\mathsf{m}' \neq \perp$*, and returns* 0 *otherwise.*

Final: A *wins if* $\mathsf{win} = 1$*.*

In this paper, we view DEM as block cipher-based DEM which uses a block cipher in a black-box way. In addition, we view the key space $\mathcal{K}$ of DEM schemes as a product set $\mathcal{K} = \mathcal{K}' \times \mathcal{K}''$, where $\mathcal{K}'$ is the key space of a block cipher, and $\mathcal{K}''$ is the key space of encryption using a block cipher as a black-box.

To define simulatable DEM, oracle DEM and permutation-driven DEM are defined following [18].

Definition 8 (Oracle DEM)**.** *A DEM scheme* $(\mathsf{O.Enc}^{\pi}, \mathsf{O.Dec}^{\pi})$ *with a key space* $\mathcal{K}$ *and a message space* $\mathcal{M}$ *is an oracle DEM scheme for a domain* $\mathcal{X}$ *if* $(\mathsf{O.Enc}, \mathsf{O.DEM})$ *has access to a permutation* π *on* $\mathcal{D}$*, and if for all permutations* $\pi : \mathcal{X} \to \mathcal{X}$*, all* $\mathsf{k} \in \mathcal{K}$*, and all* $\mathsf{m} \in \mathcal{M}$*, it holds that* $\mathsf{m} = \mathsf{Dec}^{\pi}(\mathsf{k}, \mathsf{ct})$*, where* $\mathsf{ct} \leftarrow \mathsf{Enc}^{\pi}(\mathsf{k}, \mathsf{m})$*.*

Definition 9 (Permutation-Driven DEM)**.** *A DEM scheme* $\mathsf{DEM} = (\mathsf{Enc}, \mathsf{Dec})$ *with a key space* $\mathcal{K} = \mathcal{K}' \times \mathcal{K}''$ *and a message space* $\mathcal{M}$ *is a* $(\mathcal{K} \times \mathcal{X})$*-permutation-driven DEM if the following conditions hold:*

- DEM *is an oracle DEM* $(\mathsf{O.Enc}^{\pi}, \mathsf{O.Dec}^{\pi})$ *for a domain* $\mathcal{X}$ *with a block cipher* $\{E_{\mathsf{k}'} : \mathcal{X} \to \mathcal{X}\}_{\mathsf{k}' \in \mathcal{K}'}$ *as the permutation* π *over* $\mathcal{X}$*.*
- *For any key* $(\mathsf{k}', \mathsf{k}'') \in \mathcal{K}$*, any message* $\mathsf{m} \in \mathcal{M}$*, and any ciphertexts* ct*, it holds that* $\mathsf{Enc}((\mathsf{k}', \mathsf{k}''), \mathsf{m}) = \mathsf{O.Enc}^{E_{\mathsf{k}'}}(\mathsf{k}'', \mathsf{m})$ *and* $\mathsf{Dec}((\mathsf{k}', \mathsf{k}''), \mathsf{ct}) = \mathsf{O.Dec}^{E_{\mathsf{k}'}}(\mathsf{k}'', \mathsf{ct})$*.*

Then, the simulatability of oracle DEM [18] is defined as follows.

Definition 10 (Simulatability of Oracle DEM)**.** *Let* $\mathsf{DEM} = (\mathsf{Enc}, \mathsf{Dec})$ *with a key space* $\mathcal{K} = \mathcal{K}' \times \mathcal{K}''$ *and a message space* $\mathcal{M}$ *be an oracle DEM scheme for a domain* $\mathcal{X}$*. And, we assume that* DEM *has the following algorithms* Fake *and* Make*:*

- Fake*: A randomized algorithm which, given a key* $\mathsf{k}'' \in \mathcal{K}''$ *and the bit-length* $|\mathsf{m}|$ *of a message, outputs a ciphertext* ct *and a state-information* st*.*
- Make*: A randomized algorithm which, given a state-information* st *and a message* $\mathsf{m} \in \mathcal{M}$*, outputs a relation* $\tilde{\pi} \in \mathcal{X} \times \mathcal{X}$ *which has functions* $\tilde{\pi}^{+} : \mathcal{X} \to \mathcal{X}$ *and* $\tilde{\pi}^{-} : \mathcal{X} \to \mathcal{X}$ *such that if* $(\alpha, \beta) \in \tilde{\pi}$*,* $\alpha = \tilde{\pi}^{+}(\beta)$ *and* $\beta = \tilde{\pi}^{-}(\alpha)$*.*

The oracle DEM scheme DEM *meets* ϵ*-simulatability if for all* $\mathsf{k} = (\mathsf{k}', \mathsf{k}'') \in \mathcal{K}$*, all* $\mathsf{m} \in \mathcal{M}$*, and the set* $\Pi^{\mathsf{m}}_{\mathsf{k}''} := \{\tilde{\pi} \mid (\mathsf{ct}, \mathsf{st}) \leftarrow \mathsf{Fake}(\mathsf{k}'', |\mathsf{m}|); \tilde{\pi} \leftarrow \mathsf{Make}(\mathsf{st}, \mathsf{m})\}$*, the following conditions hold:*

- *The set* $\Pi^{\mathsf{m}}_{\mathsf{k}''}$ *can be extended to a set of uniformly distributed permutations on* $\mathcal{X}$*.*
- *For any permutation* π *extended* $\Pi^{\mathsf{m}}_{\mathsf{k}''}$*, it holds that* $\Pr[\mathsf{ct} \neq \mathsf{O.Enc}^{\pi}(\mathsf{k}'', \mathsf{m})] \leq \epsilon$*, where* $\mathsf{ct} \leftarrow \mathsf{Fake}(\mathsf{k}'', |\mathsf{m}|)$*.*
- *The time-complexity of algorithms* $\mathsf{Fake}(\mathsf{k}', |\mathsf{m}|)$ *and* $\mathsf{Make}(\mathsf{st}, \mathsf{m})$ *does not exceed the time-complexity of algorithm* $\mathsf{Enc}(\mathsf{k}, \mathsf{m})$ *without counting that of oracles which is accessed by* $\mathsf{Enc}(\cdot)$*.*

2.2.4 Message Authentication Code

A message authentication code (MAC) consists of two polynomial time algorithms $(\mathsf{Tag}, \mathsf{Vrfy})$ with a key space $\mathcal{K} = \mathcal{K}(\lambda)$ and a message space $\mathcal{M} = \mathcal{M}(\lambda)$ for a security parameter λ.

Tagging Tag is an algorithm which, on input a secret key $\mathsf{k} \in \mathcal{K}$ and a message $\mathsf{m} \in \mathcal{M}$, outputs a tag τ.

Verification Vrfy is a deterministic algorithm which, on input a secret key $\mathsf{k} \in \mathcal{K}$, a message m ,and a tag τ, outputs 1 or 0.

We require a MAC scheme to be $\mathsf{correct}$, as follows

Definition 11 (Correctness)**.** *A MAC scheme* $\mathsf{MAC} = (\mathsf{Tag}, \mathsf{Vrfy})$ *with a key space* $\mathcal{K}$ *and a message space* $\mathcal{M}$ *is* $\mathsf{correct}$ *if for all* $\mathsf{k} \in \mathcal{K}$ *and all* $\mathsf{m} \in \mathcal{M}$*, it holds that* $1 = \mathsf{Vrfy}(\mathsf{k}, \mathsf{m}, \tau)$*, where* $\tau \leftarrow \mathsf{Tag}(\mathsf{k}, \mathsf{m})$*.*

As a security notion of MACs, *strong unforgeability against one-time chosen message attacks* (denoted by $\mathsf{sUF\text{-}OT\text{-}CMA}$ security) is defined as follows.

Definition 12 ($\mathsf{sUF\text{-}OT\text{-}CMA}$ security)**.** *A MAC scheme* $\mathsf{MAC} = (\mathsf{Tag}, \mathsf{Vrfy})$ *meets* $\mathsf{sUF\text{-}OT\text{-}CMA}$ *security if for any PPT adversary* A *against* MAC*, the advantage* $\mathsf{Adv}^{\text{suf-cma}}_{\mathsf{A},\mathsf{MAC}} := \Pr[\mathsf{A}\ \textit{wins}]$ *is negligible, where* $[\mathsf{A}\ \textit{wins}]$ *is the event that* A *wins in the following game:*

Setup: *A challenger chooses a key* $\mathsf{k} \in \mathcal{K}$ *uniformly at random and sets* $T \leftarrow \emptyset$ *and* $\mathsf{win} \leftarrow 0$*.*

Oracle Access: A *is allowed to access the following oracles:*

- $\mathsf{TAG}(\mathsf{m})$*: If* $T \neq \emptyset$*, the tagging oracle* $\mathsf{TAG}(\mathsf{m})$ *returns* $\perp$*. Otherwise, it returns* $\tau \leftarrow \mathsf{Tag}(\mathsf{k}, \mathsf{m})$ *and sets* $T \leftarrow T \cup \{(\mathsf{m}, \tau)\}$*.*
- $\mathsf{VRFY}(\mathsf{m}, \tau)$*: Given a message and a tag* (m, τ)*, the verification oracle* $\mathsf{VRFY}(\mathsf{m}, \tau)$ *returns* $b \leftarrow \mathsf{Vrfy}(\mathsf{k}, \mathsf{m}, \tau)$*. If* $b = 1$ *and* $(\mathsf{m}, \tau) \notin T$*, it sets* $\mathsf{win} \leftarrow 1$*.*

Final: A *wins if* $\mathsf{win} = 1$*.*

3 SIM-SO-CCA secure PKE from KEM schemes

3.1 KEM/DEM framework

In this section, we focus on the standard KEM/DEM scheme PKE^{hy}_1 starting from any $\mathsf{IND\text{-}CCA}$ secure KEM and any simulatable DEM, and prove that PKE^{hy}_1 satisfies $\mathsf{SIM\text{-}SO\text{-}CCA}$ security in the QICM. This security proof is based on the proof of Theorem 2 in [18]. However, it is not obvious that it satisfies $\mathsf{SIM\text{-}SO\text{-}CCA}$ security in the QICM because the proof in [18] uses the list of query-response pairs issued to ideal cipher oracles and we cannot apply this technique due to the quantum no-cloning theorem. To resolve this problem, we utilize a semi-classical oracle to check whether or not quantum queries meeting a condition are submitted to ideal cipher oracles, instead of using the list of ideal cipher oracles.

To construct PKE^{hy}_1 with a message space $\mathcal{M}$, we use the following primitives: Let $\mathsf{KEM} = (\mathsf{KGen}^{asy}, \mathsf{Encaps}, \mathsf{Decaps})$ be a KEM scheme with a key space $\mathcal{K} = \mathcal{K}' \times \mathcal{K}''$ and a randomness space $\mathcal{R}^{asy}$. Let $\mathsf{DEM} = (\mathsf{Enc}^{sym}, \mathsf{Dec}^{sym})$ be a DEM scheme with a key space $\mathcal{K} = \mathcal{K}' \times \mathcal{K}''$ and a message space $\mathcal{M}$.

The PKE scheme $\mathsf{PKE}^{hy}_1 = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ is described as follows:

- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$:
  1. Generate $(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}) \leftarrow \mathsf{KGen}^{asy}(1^\lambda)$.
  2. Output $\mathsf{pk} := \mathsf{pk}^{asy}$ and $\mathsf{sk} := \mathsf{sk}^{asy}$.
- $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathsf{m})$:
  1. Compute $(e, \mathsf{k}) \leftarrow \mathsf{Encaps}(\mathsf{pk}^{asy})$, and $d \leftarrow \mathsf{Enc}^{sym}(\mathsf{k}, \mathsf{m})$.
  2. Output $\mathsf{ct} := (e, d)$.
- $\mathsf{m}/\perp \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$:
  1. Parse $\mathsf{ct} = (e, d)$.
  2. Compute $\mathsf{k} \leftarrow \mathsf{Decaps}(\mathsf{sk}^{asy}, e)$.
  3. Output $\mathsf{m}' \leftarrow \mathsf{Dec}^{sym}(\mathsf{k}, d)$ if $\mathsf{k} \neq \perp$, and output $\perp$ otherwise.

If KEM is δ-correct, and DEM is correct, then the PKE scheme PKE_1^{hy} is also δ-correct, clearly. The following theorem shows the security of PKE_1^{hy}.

Theorem 1. *If a KEM scheme* KEM *meets* IND-CCA *security, and a $(\mathcal{K}, \mathcal{X})$-permutation-driven DEM scheme* DEM *corresponding to an oracle DEM* (O.Enc, O.Dec) *for a domain $\mathcal{X}$ and a block cipher E meets both ϵ_{sim}-simulatability and* OT-INT-CTXT *security, then* PKE_1^{hy} *satisfies* SIM-SO-CCA *security in the quantum ideal cipher model.*

Proof. Let A be a QPT adversary against PKE_1^{hy}. Let q_e be the total number of queries issued to the oracles $E^+(\cdot)$ and $E^-(\cdot)$. For $J \subseteq [n]$, let $K'_J := \{\mathsf{k}'_j \mid j \in J\}$. For each $i \in \{0, 1, 2, 3, 4\}$, we consider a security game Game_i, and let W_i be the event that A outputs *out* such that $R(\mathcal{M}_{\mathrm{D}}, \mathsf{m}_1, \ldots, \mathsf{m}_n, I, out) = 1$ in Game_i.

<u>Game_0:</u> This game is the same as the REAL-SIM-SO-CCA security game. Then, we have $\Pr[\mathsf{Expt}^{\text{real-so-cca}}_{\mathsf{PKE}_1^{hy},\mathsf{A}}(\lambda) \rightarrow 1] = \Pr[W_0]$.

<u>Game_1:</u> This game is the same as Game_0 except that the DEC oracle on input a decryption query $\mathsf{ct} = (e, d)$ returns $\perp$ if $e \in \{e_i\}_{i \in [n] \setminus I}$, and returns $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$ otherwise.

Let Bad be the event that A issues a decryption query $\mathsf{ct} = (e, d)$ such that $\mathsf{ct} \notin \{\mathsf{ct}_i\}_{i \in [n]}$, $e \in \{e_i\}_{i \in [n] \setminus I}$, and $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \neq \perp$. Unless Bad occurs, Game_1 is identical to Game_0. Thus, we have $|\Pr[W_0] - \Pr[W_1]| \leq \Pr[\mathsf{Bad}]$. We show $\Pr[\mathsf{Bad}] \leq n \cdot (\mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_1}(\lambda) + \mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{DEM},\mathsf{F}}(\lambda))$. To do this, we fix an index $i^* \in [n]$ and consider a security game Game'_1 which is the same as Game_1 except that the key k_{i^*} is chosen uniformly at random. In addition, let $\mathsf{Bad}^{(i^*)}$ (resp., $\mathsf{Bad}^{(i^*)\prime}$) be the event that A submits a decryption query (e, d) such that $e = e_{i^*}$ and $\mathsf{Dec}(\mathsf{sk}, (e, d)) \neq \perp$ in Game_1 (resp., Game'_1).

To show $\left|\Pr[\mathsf{Bad}^{(i^*)}] - \Pr[\mathsf{Bad}^{(i^*)\prime}]\right| \leq \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_1^{(i^*)}}(\lambda)$, we construct a PPT algorithm $\mathsf{D}_1^{(i^*)}$ breaking the IND-CCA security of KEM in the following way: $\mathsf{D}_1^{(i^*)}$ is given the public key pk^{asy} of KEM. At the beginning of the security game, it chooses $f_E : \mathcal{K}' \times \mathcal{X} \rightarrow \mathcal{X}$ uniformly at random, such that $f_E(\mathsf{k}', \cdot)$ is a permutation over $\mathcal{X}$ for each $\mathsf{k}' \in \mathcal{K}'$, in order to simulate the E^+ and E^- oracles. Then, $\mathsf{D}_1^{(i^*)}$ sets $I \leftarrow \emptyset$ and sends $\mathsf{pk} := \mathsf{pk}^{asy}$ to A. When A submits $\mathcal{M}_{\mathrm{D}}$, $\mathsf{D}_1^{(i^*)}$ does the following for each $i \in [n]$:

1. If $i = i^*$, obtain $(e_{i^*}, \mathsf{k}_{i^*})$ by accessing the Challenge oracle in the IND-CCA security game. Otherwise, compute $(e_i, \mathsf{k}_i) \leftarrow \mathsf{Encaps}(\mathsf{pk}; r_i)$, where $r_i \in \mathcal{R}^{asy}$ is sampled at random.
2. Choose $\mathsf{m}_i \leftarrow \mathcal{M}_D$ and compute $d_i \leftarrow \mathsf{Enc}^{sym}(\mathsf{k}_i, \mathsf{m}_i)$.

Then, it returns $(\mathsf{ct}_i)_{i \in [n]}$ to A, where $\mathsf{ct}_i = (e_i, d_i)$ for $i \in [n]$. In addition, $\mathsf{D}_1^{(i^*)}$ simulates the DEC and OPEN oracles, as follows:

- DEC(ct): Take $\mathsf{ct} = (e, d)$ as input. In the case $e = e_{i^*}$, halt and output 1 if $(e, d) \neq (e_{i^*}, d_{i^*})$ and $\mathsf{Dec}^{sym}(\mathsf{k}_{i^*}, d) \neq \perp$, and return $\perp$ otherwise. In the case $e \neq e_{i^*}$, submit e to the given decapsulation oracle and receive k. Return $\perp$ if $\mathsf{k} = \perp$, and return $\mathsf{Dec}^{sym}(\mathsf{k}, d)$ if $\mathsf{k} \neq \perp$.

- $\mathsf{OPEN}(i)$: Set $I \leftarrow I \cup \{i\}$. Abort if $i = i^*$. Return (m_i, r_i) if $i \neq i^*$.

The E^+ and E^- oracles are simulated by using f_E. When A outputs out, $\mathsf{D}_1^{(i^*)}$ outputs 0 if $\mathsf{Bad}_1^{(i^*)}$ does not happen.

$\mathsf{D}_1^{(i^*)}$ simulates the view of A completely. If A submits a decryption query (e, d) such that $e = e_{i^*}$ and $\mathsf{Dec}(\mathsf{sk}, (e, d)) \neq \perp$, $\mathsf{D}_1^{(i^*)}$ breaks the $\mathsf{IND\text{-}CCA}$ security in the straightforward way. Thus, the probability of distinguishing the two games Game_1 and Game_1' is at most $\mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_1^{(i^*)}}(\lambda)$.

To show $\Pr[\mathsf{Bad}^{(i^*)\prime}] \leq \mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{DEM},\mathsf{F}^{(i^*)}}(\lambda)$, we construct a PPT algorithm $\mathsf{F}^{(i^*)}$ breaking the $\mathsf{OT\text{-}INT\text{-}CTXT}$ security of DEM, as follows: $\mathsf{F}^{(i^*)}$ is given the two oracles ENC and VRFY in the $\mathsf{OT\text{-}INT\text{-}CTXT}$ security game. At the beginning of the security game, $\mathsf{F}^{(i^*)}$ generates $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$, sets $I \leftarrow \emptyset$, and gives pk to A. When A submits a distribution $\mathcal{M}_{\mathrm{D}}$, $\mathsf{F}^{(i^*)}$ does the following for each $i \in [n]$:

1. Compute $(e_i, \mathsf{k}_i) \leftarrow \mathsf{Encaps}(\mathsf{pk}; r_i)$, where $r_i \in \mathcal{R}^{asy}$ is sampled at random.
2. Choose $\mathsf{m}_i \leftarrow \mathcal{M}_{\mathrm{D}}$.
3. If $i = i^*$, obtain d_{i^*} by accessing $\mathsf{ENC}(\mathsf{m}_{i^*})$. If $i \in [n]\backslash\{i^*\}$, compute $d_i \leftarrow \mathsf{Enc}^{sym}(\mathsf{k}_i, \mathsf{m}_i)$.
4. Set $\mathsf{ct}_i \leftarrow (e_i, d_i)$.

Then, $\mathsf{F}^{(i^*)}$ returns $(\mathsf{ct}_i)_{i\in[n]}$ to A. $\mathsf{F}^{(i^*)}$ simulates $\mathsf{OPEN}(\cdot)$, $E^+(\cdot,\cdot)$, and $E^-(\cdot,\cdot)$ in the same way as the above algorithm $\mathsf{D}_1^{(i^*)}$. The DEC oracle is simulated as follows: If $e = e_{i^*}$ for a given $\mathsf{ct} = (e, d)$, $\mathsf{F}^{(i^*)}$ submits d to the VRFY oracle. If VRFY returns 1, $\mathsf{F}^{(i^*)}$ halts and wins in the $\mathsf{sUF\text{-}OT\text{-}CMA}$ security game. Otherwise, it returns $\perp$. If $e \neq e_{i^*}$, $\mathsf{F}^{(i^*)}$ computes $\mathsf{k} \leftarrow \mathsf{Decaps}(\mathsf{sk}^{asy}, e)$ and returns $\mathsf{Dec}^{sym}(\mathsf{k}, d) \in \mathcal{M} \cup \{\perp\}$. When A outputs out, $\mathsf{F}^{(i^*)}$ aborts this game if $\mathsf{Bad}^{(i^*)\prime}$ has never happened.

The winning condition of $\mathsf{F}^{(i^*)}$ is identical to the condition that $\mathsf{Bad}^{(i^*)\prime}$ occurs. Hence, it wins in the $\mathsf{OT\text{-}INT\text{-}CTXT}$ security game if A outputs a ciphertext query (e, d) such that $e \neq e_{i^*}$ and the VRFY on input d returns 1.

Therefore, we have $|\Pr[W_0] - \Pr[W_1]| \leq n \cdot (\mathsf{Adv}^{\text{ind-cca}}_{\mathsf{PKE}^{hy},\mathsf{D}_1}(\lambda) + \mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{DEM},\mathsf{F}}(\lambda))$ by taking the union bound over $i^* \in [n]$.

$\underline{\mathsf{Game}_2}$: This game is the same as Game_1 except that the security game is aborted if the challenger generates $(e_i, (\mathsf{k}_i', \mathsf{k}_i'')) \leftarrow \mathsf{Encaps}(\mathsf{pk})$ such that $\mathsf{k}_i' \in K'_{[i-1]}$ for $i \in [n]$.

The probability of choosing $\mathsf{k}_i' \in K'_{[i-1]}$ by running $\mathsf{Encaps}(\mathsf{pk})$ for $i \in [n]$ is at most $n^2/|\mathcal{K}'|$. Thus, we have $|\Pr[W_1] - \Pr[W_2]| \leq n^2/|\mathcal{K}'|$.

$\underline{\mathsf{Game}_3}$: This game is the same as Game_2 except that given a distribution $\mathcal{M}_{\mathrm{D}}$, the challenger does the following for $i \in [n]$:

1. Generate $(e_i, (\mathsf{k}_i', \mathsf{k}_i'')) \leftarrow \mathsf{Encaps}(\mathsf{pk})$. Abort if $\mathsf{k}_i' \in K'_{[i-1]}$.
2. Compute $(d_i, \mathsf{st}_i) \leftarrow \mathsf{O.Enc}^{E_{\mathsf{k}_i'}}(\mathsf{k}_i'', |\mathsf{m}_i|)$.
3. Compute $\tilde{\pi}_i \leftarrow \mathsf{Make}(\mathsf{st}_i, \mathsf{m}_i)$, and set $E^+(\mathsf{k}_i', \cdot) \leftarrow \tilde{\pi}_i^+(\cdot)$ and $E^-(\mathsf{k}_i', \cdot) \leftarrow \tilde{\pi}_i^-(\cdot)$.
4. Abort if $d_i \neq \mathsf{O.Enc}^{E_{\mathsf{k}_i'}}(\mathsf{k}_i'', \mathsf{m}_i)$.
5. Set $\mathsf{ct}_i \leftarrow (e_i, d_i)$.

Then, it returns $(\mathsf{ct}_i)_{i\in[n]}$ to the adversary A.

Due to the simulatability of DEM, the probability that the challenger aborts when producing d_i is at most ϵ_{sim}. In addition, since the challenger sets $E^+(\mathsf{k}_i', \cdot) \leftarrow \tilde{\pi}_i^+(\cdot)$, $E^-(\mathsf{k}_i', \cdot) \leftarrow \tilde{\pi}_i^-(\cdot)$ when producing the ciphertexts $(\mathsf{ct}_i)_{i\in[n]}$, the indistinguishability of E^+, E^- in the two games follows [37, Lemma 13]. Hence, we have $|\Pr[W_2] - \Pr[W_3]| \leq n \cdot \epsilon_{sim} + 4nq_e/\sqrt{|\mathcal{K}'|}$ owing to the union bound over $i \in [n]$.

$\underline{\mathsf{Game}_4}$: This game is the same as Game_3 except that the procedures of the challenger and the OPEN oracle are modified as follows: Given a distribution $\mathcal{M}_\mathrm{D}$, the challenger computes $(e_i, (\mathsf{k}'_i, \mathsf{k}''_i)) \leftarrow \mathsf{Encaps}(\mathsf{pk}; r_i)$ (aborts if $\mathsf{k}'_i \in K'_{[i-1]}$) and $(d_i, \mathsf{st}_i) \leftarrow \mathsf{Fake}(\mathsf{k}''_i, |\mathsf{m}_i|)$, and then sets $\mathsf{ct}_i \leftarrow (e_i, d_i)$ for each $i \in [n]$. In addition, the OPEN oracle on input i is modified as follows:

1. Set $I \leftarrow I \cup \{i\}$.
2. Choose $\mathsf{m}_i \leftarrow \mathcal{M}_\mathrm{D}$.
3. Compute $\tilde{\pi}_i \leftarrow \mathsf{Make}(\mathsf{st}_i, \mathsf{m}_i)$, and set $E^+(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^+(\cdot)$ and $E^-(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^-(\cdot)$.
4. Abort if $d_i \neq \mathsf{O.Enc}^{E_{\mathsf{k}'_i}}(\mathsf{k}''_i, \mathsf{m}_i)$.
5. Return (m_i, r_i).

In order to show the indistinguishability between Game_3 and Game_4, we fix an index $i^* \in [n]$ and consider the oracles $E^+\backslash\{\mathsf{k}'_{i^*}\}$ and $E^-\backslash\{\mathsf{k}'_{i^*}\}$ which first query the semi-classical oracle $O^{SC}_{\{\mathsf{k}'_{i^*}\}}$ and then E^+ and E^-, respectively. Furthermore, let $\mathsf{Hybrid}^{(0)}$ be the same game as Game_3. For $i^* \in [n]$, we consider the game $\mathsf{Hybrid}^{(i^*)}$ which is the same as $\mathsf{Hybrid}^{(i^*-1)}$ except for the following: Given $\mathcal{M}_\mathrm{D}$, the challenger generates the i^*-th ciphertext, as follows:

1. Compute $(e_{i^*}, (\mathsf{k}'_{i^*}, \mathsf{k}''_{i^*})) \leftarrow \mathsf{Encaps}(\mathsf{pk}; r_{i^*})$. Abort if $\mathsf{k}'_{i^*} \in K'_{[i^*-1]}$.
2. Compute $(d_{i^*}, \mathsf{st}_{i^*}) \leftarrow \mathsf{Fake}(\mathsf{k}''_{i^*}, |\mathsf{m}_{i^*}|)$.
3. Set $\mathsf{ct}_{i^*} \leftarrow (e_{i^*}, d_{i^*})$.

In addition, the OPEN oracle on input i^* does the following:

1. Set $I \leftarrow I \cup \{i^*\}$.
2. Choose $\mathsf{m}_{i^*} \leftarrow \mathcal{M}_\mathrm{D}$.
3. Compute $\tilde{\pi}_{i^*} \leftarrow \mathsf{Make}(\mathsf{st}_{i^*}, \mathsf{m}_{i^*})$, and set $E^+(\mathsf{k}'_{i^*}, \cdot) \leftarrow \tilde{\pi}_{i^*}^+(\cdot)$ and $E^-(\mathsf{k}'_{i^*}, \cdot) \leftarrow \tilde{\pi}_{i^*}^-(\cdot)$.
4. Abort if $d_{i^*} \neq \mathsf{O.Enc}^{E_{\mathsf{k}'_{i^*}}}(\mathsf{k}''_{i^*}, \mathsf{m}_{i^*})$.
5. Return $(\mathsf{m}_{i^*}, r_{i^*})$.

Then, notice that $\mathsf{Hybrid}^{(n)}$ is identical to Game_4. For $i^* \in \{0, 1, \ldots, n\}$, let $W_H^{(i^*)}$ be the event that A outputs out such that $R(\mathcal{M}_\mathrm{D}, \mathsf{m}_1, \ldots, \mathsf{m}_n, I, out) = 1$ in $\mathsf{Hybrid}^{(i^*)}$, and let $\mathsf{Find}^{(i^*)}$ be the event that the semi-classical oracle $O^{SC}_{\{\mathsf{k}'_{i^*}\}}$ returns 1 in the same game as $\mathsf{Hybrid}^{(i^*)}$ except that the E^+ and E^- oracles are replaced by $E^+\backslash\{\mathsf{k}'_{i^*}\}$ and $E^-\backslash\{\mathsf{k}'_{i^*}\}$, respectively.

By applying Proposition 1, we have $\left|\Pr[W_H^{(i^*-1)}] - \Pr[W_H^{(i^*)}]\right| \leq 2\sqrt{q_e \cdot \Pr[\mathsf{Find}^{(i^*)}]}$. In order to show that $\Pr[\mathsf{Find}^{(i^*)}]$ is negligible, we consider the event $\mathsf{Find}^{(i^*)\prime}$ which the semi-classical oracle $O^{SC}_{\{\mathsf{k}'_{i^*}\}}$ returns 1 in the same game as $\mathsf{Hybrid}^{(i^*)}$ except that $\mathsf{k}_{i^*} = (\mathsf{k}'_{i^*}, \mathsf{k}''_{i^*})$ is chosen uniformly at random (denoted by $\mathsf{Hybrid}^{(i^*)\prime}$). To show $\left|\Pr[\mathsf{Find}^{(i^*)}] - \Pr[\mathsf{Find}^{(i^*)\prime}]\right| \leq \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_2^{(i^*)}}(\lambda)$, we construct a PPT algorithm $\mathsf{D}_2^{(i^*)}$ breaking the $\mathsf{IND\text{-}CCA}$ security of KEM, as follows: $\mathsf{D}_2^{(i^*)}$ is given the public key pk^{asy} of KEM. At the beginning of the security game, $\mathsf{D}_2^{(i^*)}$ sets $I \leftarrow \emptyset$ and $\mathsf{find} \leftarrow 0$, and gives $\mathsf{pk} := \mathsf{pk}^{asy}$ to A. When A submits a distribution $\mathcal{M}_\mathrm{D}$, $\mathsf{D}_2^{(i^*)}$ does the following for $i \in [n]$:

- In the case $i \leq i^*$:

1. If $i = i^*$, obtain $(e_{i^*}, (\mathsf{k}'_{i^*}, \mathsf{k}''_{i^*}))$ by accessing the $\mathsf{Challenge}$ oracle in the $\mathsf{IND\text{-}CCA}$ security game. If $i \le i^* - 1$, compute $(e_i, (\mathsf{k}'_i, \mathsf{k}''_i)) \leftarrow \mathsf{Encaps}(\mathsf{pk}^{asy}; r_i)$.
2. Abort if $\mathsf{k}'_i \in K'_{[i-1]}$.
3. Compute $(d_i, \mathsf{st}_i) \leftarrow \mathsf{Fake}(\mathsf{k}''_i, |\mathsf{m}_i|)$.
4. Set $\mathsf{ct}_i \leftarrow (e_i, d_i)$.

- In the case $i \ge i^* + 1$:
  1. Compute $(e_i, (\mathsf{k}'_i, \mathsf{k}''_i)) \leftarrow \mathsf{Encaps}(\mathsf{pk}^{asy}; r_i)$. Abort if $\mathsf{k}'_i \in K'_{[i-1]}$.
  2. Compute $(d_i, \mathsf{st}_i) \leftarrow \mathsf{O.Enc}^{E_{\mathsf{k}'_i}}(\mathsf{k}''_i, \mathsf{m}_i)$.
  3. Compute $\tilde{\pi}_i \leftarrow \mathsf{Make}(\mathsf{st}_i, \mathsf{m}_i)$, and set $E^+(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^+(\cdot)$ and $E^-(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^-(\cdot)$.
  4. Abort if $d_i \ne \mathsf{O.Enc}^{E_{\mathsf{k}'_i}}(\mathsf{k}''_i, \mathsf{m}_i)$.
  5. Set $\mathsf{ct}_i \leftarrow (e_i, d_i)$.

Then it returns $(\mathsf{ct}_i)_{i\in[n]}$ to A. The DEC and OPEN oracles are simulated as follows:

- $\mathsf{DEC}(\mathsf{ct})$: Take $\mathsf{ct} = (e, d)$ as input. If $e \in \{e_i\}_{i\in[n]\setminus I}$, return $\perp$. If $e \notin \{e_i\}_{i\in[n]\setminus I}$, submit e to the given decapsulation oracle and receive k. Return $\perp$ if $\mathsf{k} = \perp$, and return $\mathsf{Dec}^{sym}(\mathsf{k}, d)$ if $\mathsf{k} \ne \perp$.
- $\mathsf{OPEN}(i)$:
  1. Abort if $i = i^*$. Otherwise, set $I \leftarrow I \cup \{i\}$.
  2. If $i \le i^* - 1$:
     (a) Choose $\mathsf{m}_i \leftarrow \mathcal{M}_{\mathrm{D}}$.
     (b) Compute $\tilde{\pi}_i \leftarrow \mathsf{Make}(\mathsf{st}_i, \mathsf{m}_i)$, and set $E^+(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^+(\cdot)$ and $E^-(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^-(\cdot)$.
     (c) Abort if $d_i \ne \mathsf{O.Enc}^{E_{\mathsf{k}'_i}}(\mathsf{k}''_i, \mathsf{m}_i)$.
  3. Return (m_i, r_i).

The E^+ and E^- oracles are simulated in the same way as the previous algorithms $\mathsf{D}_1^{(i^*)}$ and $\mathsf{F}^{(i^*)}$. When A outputs out, $\mathsf{D}_2^{(i^*)}$ outputs find.

If k_{i^*} is generated by the Encaps algorithm, $\mathsf{D}_2^{(i^*)}$ simulates $\mathsf{Hybrid}^{(i^*)}$. If k_{i^*} is uniformly random, it simulates $\mathsf{Hybrid}^{(i^*)\prime}$. Hence, we have $\left|\Pr[\mathsf{Find}^{(i^*)}] - \Pr[\mathsf{Find}^{(i^*)\prime}]\right| \le \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_2^{(i^*)}}(\lambda)$.

In addition, we have $\Pr[\mathsf{Find}^{(i^*)\prime}] \le 4q_e/|\mathcal{K}'|$ from Proposition 2, because only the two oracles E^+, E^- contain the information of the uniformly random k_{i^*}. Hence, we obtain the following inequality

$$\Pr[\mathsf{Find}^{(i^*)}] \le \left|\Pr[\mathsf{Find}^{(i^*)}] - \Pr[\mathsf{Find}^{(i^*)\prime}]\right| + \Pr[\mathsf{Find}^{(i^*)\prime}] \le \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_2^{(i^*)}}(\lambda) + \frac{4q_e}{|\mathcal{K}'|}.$$

Therefore, we obtain

$$|\Pr[W_3] - \Pr[W_4]| \le 2n\sqrt{q_e \cdot \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_2}(\lambda) + \frac{4q_e^2}{|\mathcal{K}'|}} \le 2n\sqrt{q_e \cdot \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_2}(\lambda)} + \frac{4nq_e}{\sqrt{|\mathcal{K}'|}}.$$

Finally, we prove $\Pr[\mathsf{Expt}^{\text{ideal-so-cca}}_{\mathsf{PKE}_1^{hy},\mathsf{S}}(\lambda) \to 1] = \Pr[W_4]$. We construct a simulator S in the following way: It is given the $\overline{\mathsf{OPEN}}$ oracle in the $\mathsf{IDEAL\text{-}SIM\text{-}SO\text{-}CCA}$ security game. At the beginning of this game, S generates $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$, sets $I \leftarrow \emptyset$, and gives pk to A. When A submits $\mathcal{M}_{\mathrm{D}}$, it receives $|\mathsf{m}_1|, \ldots, |\mathsf{m}_n|$ in the $\mathsf{IDEAL\text{-}SIM\text{-}SO\text{-}CCA}$ security game, generates $(e_i, \mathsf{k}_i) \leftarrow \mathsf{Encaps}(\mathsf{pk}; r_i)$ and $d_i \leftarrow \mathsf{Fake}(\mathsf{k}''_i, |\mathsf{m}_i|)$ for $i \in [n]$, and returns $(\mathsf{ct}_i)_{i\in[n]}$ (where $\mathsf{ct}_i = (e_i, d_i)$ for $i \in [n]$). In the same way as the game-hop of Game_4, S simulates E^+ and E^- by using $(\mathsf{Fake}, \mathsf{Make})$ and a function $f_E : \mathcal{K}' \times \mathcal{X} \to \mathcal{X}$ chosen uniformly at random. It simulates the DEC and OPEN oracles as follows:

- $\mathsf{DEC}(\mathsf{ct})$:

  1. Parse $\mathsf{ct} = (e, d)$.
  2. Return $\perp$ if $e \in \{e_i\}_{i \in [n] \setminus I}$.
  3. Compute $\mathsf{k} \leftarrow \mathsf{Decaps}(\mathsf{sk}, e)$.
  4. Return $\perp$ if $\mathsf{k} = \perp$. Return $\mathsf{Dec}^{sym}(\mathsf{k}, d) \in \mathcal{M} \cup \{\perp\}$ otherwise.

- $\mathsf{OPEN}(i)$:

  1. Set $I \leftarrow I \cup \{i\}$.
  2. Obtain m_i by accessing the given open oracle $\overline{\mathsf{OPEN}}$.
  3. Compute $\tilde{\pi}_i \leftarrow \mathsf{Make}(\mathsf{st}_i, \mathsf{m}_i)$ and set $E^+(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^+(\cdot)$ and $E^-(\mathsf{k}'_i, \cdot) \leftarrow \tilde{\pi}_i^-(\cdot)$.
  4. Abort if $d_i \neq \mathsf{O.Enc}^{E_{\mathsf{k}'_i}}(\mathsf{k}''_i, \mathsf{m}_i)$.
  5. Return (m_i, r_i).

When A outputs out, S halts and outputs $R(\mathcal{M}_{\mathrm{D}}, m_1, \ldots, m_n, I, out)$.

S completely simulates Game_4 by using only the given oracle $\overline{\mathsf{OPEN}}$. Thus, we have $\Pr[\mathsf{Expt}^{\text{ideal-so-cca}}_{\mathsf{PKE}_1^{hy},\mathsf{S}}(\lambda) \rightarrow 1] = \Pr[W_4]$.

Therefore, we obtain the following advantage

$$\mathsf{Adv}^{\text{sim-so-cca}}_{\mathsf{PKE}_1^{hy},\mathsf{A},\mathsf{S},R}(\lambda) \leq n \cdot \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_1}(\lambda) + 2n\sqrt{q_e \cdot \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathsf{D}_2}(\lambda)} + n \cdot \mathsf{Adv}^{\text{int-ctxt}}_{\mathsf{DEM},\mathsf{F}}(\lambda) + n \cdot \epsilon_{sim} + \frac{8nq_e}{\sqrt{|\mathcal{K}'|}} + \frac{n^2}{|\mathcal{K}'|}.$$

From the discussion above, the proof is completed. $\square$

3.2 PKE from $\mathsf{FO}^{\not\perp}$ Transformation

We describe a PKE scheme PKE_2^{hy} constructed from an FO-based KEM $\mathsf{FO}^{\not\perp}$ and a MAC, and prove that this scheme satisfies $\mathsf{SIM\text{-}SO\text{-}CCA}$ security in the QROM. Concretely, we use the FO-based KEM scheme $\mathsf{FO}^{\not\perp}$ and any $\mathsf{sUF\text{-}OT\text{-}CMA}$ secure MAC. As KEM schemes, we can apply not only $\mathsf{FO}^{\not\perp}$ but also other transformations $\mathsf{FO}_m^{\not\perp}$, $\mathsf{QFO}^{\not\perp}$, and $\mathsf{QFO}_m^{\not\perp}$, which are classified in [20]. In this paper, we select $\mathsf{FO}^{\not\perp}$ to construct PKE_2^{hy}. Notice that in the same way as the security proof of PKE_2^{hy}, it is possible to prove the security of PKE_2^{hy} using $\mathsf{FO}_m^{\not\perp}$, $\mathsf{QFO}^{\not\perp}$, or $\mathsf{QFO}_m^{\not\perp}$, instead of $\mathsf{FO}^{\not\perp}$.

To construct PKE_2^{hy} with a message space $\mathcal{M}$, we use the following primitives: Let $\mathsf{PKE} = (\mathsf{KGen}^{asy}, \mathsf{Enc}^{asy}, \mathsf{Dec}^{asy})$ be a (δ-correct) PKE scheme with a message space $\mathcal{M}^{asy}$, a randomness space $\mathcal{R}^{asy}$, and a ciphertext space $\mathcal{C}^{asy}$. Let $\mathsf{MAC} = (\mathsf{Tag}, \mathsf{Vrfy})$ be a MAC scheme with a key space $\mathcal{K}^{mac}$. Let $\mathsf{G} : \mathcal{M}^{asy} \rightarrow \mathcal{R}^{asy}$, $\mathsf{H} : \mathcal{M}^{asy} \times \mathcal{C}^{asy} \rightarrow \mathcal{K}^{sym} \times \mathcal{K}^{mac}$ be random oracles, where $\mathcal{K}^{sym} = \mathcal{M}$ is a key space.

$\mathsf{PKE}_2^{hy} = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ is constructed as follows:

- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$:

  1. Generate $(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}) \leftarrow \mathsf{KGen}^{asy}(1^\lambda)$.
  2. Choose $s \xleftarrow{\$} \mathcal{M}^{asy}$.
  3. Output $\mathsf{pk} := \mathsf{pk}^{asy}$ and $\mathsf{sk} := (\mathsf{sk}^{asy}, s)$.

- $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathsf{m})$:

  1. Choose $r \xleftarrow{\$} \mathcal{M}^{asy}$.
  2. Choose $e \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r; \mathsf{G}(r))$.
  3. Compute $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow \mathsf{H}(r, e)$.

4. Compute $d \leftarrow \mathsf{k}^{sym} \oplus \mathsf{m}$, $\tau \leftarrow \mathsf{Tag}(\mathsf{k}^{mac}, d)$.
5. Output $\mathsf{ct} := (e, d, \tau)$.

- $\mathsf{m}/\bot \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$:
  1. Parse $\mathsf{ct} = (e, d, \tau)$.
  2. Choose $r' \leftarrow \mathsf{Dec}^{asy}(\mathsf{sk}^{asy}, e)$.
  3. Compute $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow \mathsf{H}(r', e)$ if $e = \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r'; \mathsf{G}(r'))$.
     Otherwise, compute $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow \mathsf{H}(s, e)$.
  4. Output $\mathsf{m} := d \oplus \mathsf{k}^{sym}$ if $\mathsf{Vrfy}(\mathsf{k}^{mac}, d, \tau) = 1$, and output $\bot$ otherwise.

It is clear that PKE_2^{hy} is δ-correct if PKE is δ-correct, and MAC is correct. The following theorem shows the security of PKE_2^{hy}.

Theorem 2. *If* PKE *meets* IND-CPA *security, and* MAC *meets* sUF-OT-CMA *security, then* PKE_2^{hy} *satisfies* SIM-SO-CCA *security in the quantum random oracle model.*

Proof. Let A be a QPT adversary against PKE_2^{hy}. Let q_g be the number of queries issued to the G oracle, and q_h be the number of queries issued to the H oracle. We consider a sequence of security games $\mathsf{Game}_0, \dots, \mathsf{Game}_7$. For $i \in \{0, 1, \dots, 7\}$, let W_i be the event that A outputs *out* such that $R(\mathcal{M}_{\mathrm{D}}, \mathsf{m}_1, \dots, \mathsf{m}_n, I, out) = 1$ in Game_i.

<u>Game_0:</u> This is the REAL-SIM-SO-CCA security game. Then, we have $\Pr[W_0] = \Pr[\mathsf{Expt}_{\mathsf{PKE}}^{\text{real-so-cca}}(\lambda) \to 1]$.

<u>Game_1:</u> This game is the same as Game_0 except that the DEC oracle computes $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow \mathsf{H}'_q(e)$ instead of $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow \mathsf{H}(s, e)$, if $e \neq \mathsf{Enc}^{asy}(\mathsf{pk}, r'; \mathsf{G}(r'))$, where $\mathsf{H}'_q : \mathcal{CT}^{asy} \to \mathcal{K}^{sym} \times \mathcal{K}^{mac}$ is a random oracle. Due to [27, Lemma 4], we have $|\Pr[W_0] - \Pr[W_1]| \leq 2q_h/\sqrt{|\mathcal{M}^{asy}|}$.

We define $\mathsf{G}' : \mathcal{M}^{asy} \to \mathcal{R}^{asy}$ as a random oracle which, on input $r \in \mathcal{M}^{asy}$, returns a value sampled from the uniform distribution over a set of "good" random coins $\mathcal{R}^{asy}_{good}(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}, r) = \{\hat{r} \in \mathcal{R}^{asy} \mid \mathsf{Dec}^{asy}(\mathsf{sk}^{asy}, \mathsf{Enc}^{asy}(\mathsf{pk}, r; \hat{r})) = r\}$. Let $\delta(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}, r) = |\mathcal{R}^{asy} \backslash \mathcal{R}^{asy}_{good}(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}, r)|/|\mathcal{R}^{asy}|$ denote the fraction of bad random coins, and let $\delta(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}) = \max_{r \in \mathcal{M}^{asy}} \delta(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}, r)$. Then, we have $\delta = \mathbf{E}[\delta(\mathsf{pk}^{asy}, \mathsf{sk}^{asy})]$ as the expectation of $\delta(\mathsf{pk}^{asy}, \mathsf{sk}^{asy})$, which is taken over $(\mathsf{pk}^{asy}, \mathsf{sk}^{asy}) \leftarrow \mathsf{KGen}^{asy}(1^\lambda)$.

<u>Game_2:</u> This game is the same as Game_1 except that we replace the G oracle by the random oracle $\mathsf{G}' : \mathcal{M}^{asy} \to \mathcal{R}^{asy}$. Due to [27, Lemma 2] (i.e., generic search problem [2, 26, 27]), we have $|\Pr[W_1] - \Pr[W_2]| \leq 2q_g\sqrt{\delta}$.

<u>Game_3:</u> This game is the same as Game_2 except that the random oracle $\mathsf{H}(r, e)$ returns $\mathsf{H}_q(\mathsf{Enc}^{asy}(\mathsf{pk}, r; \mathsf{G}(r)))$ if $e = \mathsf{Enc}^{asy}(\mathsf{pk}, r; \mathsf{G}(r))$, and returns $\mathsf{H}'(r, e)$ otherwise. $\mathsf{H}_q : \mathcal{C}^{asy} \to \mathcal{K}^{sym} \times \mathcal{K}^{mac}$ and $\mathsf{H}' : \mathcal{M}^{asy} \times \mathcal{C}^{asy} \to \mathcal{K}^{sym} \times \mathcal{K}^{mac}$ are random oracles.

Since the G' oracle returns "good" random coins, $\mathsf{Enc}^{asy}(\mathsf{pk}, \cdot\, ; \mathsf{G}(\cdot))$ is injective. Thus, we can view $\mathsf{H}_q(\mathsf{Enc}^{asy}(\mathsf{pk}, \cdot\, ; \mathsf{G}(\cdot)))$ as a perfect random oracle, and $\Pr[W_3] = \Pr[W_2]$ holds.

<u>Game_4:</u> This game is the same as Game_3 except that the DEC oracle is modified as follows: Given a decryption query $\mathsf{ct} = (e, d, \tau)$, DEC computes $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow \mathsf{H}_q(e)$. Then, it returns $\mathsf{m} \leftarrow \mathsf{k}^{sym} \oplus d$ if $\mathsf{Vrfy}(\mathsf{k}^{mac}, d, \tau) = 1$, and returns $\bot$ otherwise.

In the case $e = \mathsf{Enc}^{asy}(\mathsf{pk}, r; \mathsf{G}(r))$, both the DEC oracles in Game_3 and Game_4 return the same value. In the case $e \neq \mathsf{Enc}^{asy}(\mathsf{pk}, r; \mathsf{G}(r))$, A cannot distinguish between Game_3 and Game_4 since both the H oracles in the two games return uniformly random values. Thus, we have $\Pr[W_4] = \Pr[W_3]$.

<u>Game_5:</u> This game is the same as Game_4 except that we replace the G' oracle by the G oracle. In the same way as the game-hop of Game_2, we have $|\Pr[W_4] - \Pr[W_5]| \leq 2q_g\sqrt{\delta}$.

<u>Game_6:</u> This game is the same as Game_5 except for the way of producing ciphertexts $(\mathsf{ct}_i)_{i \in [n]}$ and the procedure of the OPEN oracle:

- At the beginning of the security game, the challenger chooses $r_i \xleftarrow{\$} \mathcal{M}^{asy}$ and $\hat{r}_i \xleftarrow{\$} \mathcal{R}^{asy}$, and computes $e_i \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_i; \hat{r}_i)$ for $i \in [n]$.

- When A submits a distribution $\mathcal{M}_{\mathrm{D}}$, the challenger chooses $d_i \xleftarrow{\$} \mathcal{K}^{sym}$ and $\mathsf{k}_i^{mac} \xleftarrow{\$} \mathcal{K}^{mac}$, computes $\tau_i \leftarrow \mathsf{Tag}(\mathsf{k}_i^{mac}, d_i)$, and then returns $(\mathsf{ct}_i)_{i\in[n]}$, where $\mathsf{ct}_i = (e_i, d_i, \tau_i)$ for $i \in [n]$.
- Given $i \in [n]$, the OPEN oracle chooses $\mathsf{m}_i \leftarrow \mathcal{M}_{\mathrm{D}}$, sets $I \leftarrow I \cup \{i\}$, $\mathsf{G}(r_i) \leftarrow \hat{r}_i$, and $\mathsf{H}(r_i, e_i) \leftarrow (d_i \oplus \mathsf{m}_i, \mathsf{k}_i^{mac})$, and then returns (m_i, r_i).

Regarding the indistinguishability between Game_5 and Game_6, the following lemma holds.

Lemma 1. *For any QPT algorithm A against PKE_2^{hy} that makes at most q_g queries to G and at most q_h queries to H, there exists a PPT algorithm D against PKE such that*

$$|\Pr[W_5] - \Pr[W_6]| \leq 2n\sqrt{(q_g + q_h) \cdot \mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{D}}(\lambda)} + \frac{8n(q_g + q_h)}{\sqrt{|\mathcal{M}^{asy}|}}.$$

Lemma 1 is proven below. Due to this lemma, $|\Pr[W_5] - \Pr[W_6]|$ is negligible in λ if PKE satisfies $\mathsf{IND\text{-}CPA}$ security.

<u>Game_7:</u> This game is the same as Game_6 except that the DEC oracle on input $\mathsf{ct} = (e, d, \tau)$ returns $\perp$ if $\mathsf{ct} \notin \{\mathsf{ct}_i\}_{i\in[n]}$ and $e \in \{e_i\}_{i\in[n]\setminus I}$.

In order to show the indistinguishability between Game_6 and Game_7, we consider the event Bad that A issues a decryption query $\mathsf{ct} = (e, d, \tau)$ such that $\mathsf{ct} \notin \{\mathsf{ct}_i\}_{i\in[n]}$, $e \in \{e_i\}_{i\in[n]\setminus I}$, and $\mathsf{Vrfy}(\mathsf{k}^{mac}, d, \tau) = 1$. Then, if Bad does not occur, Game_7 is identical to Game_6. Thus, we have $|\Pr[W_6] - \Pr[W_7]| \leq \Pr[\mathsf{Bad}]$.

In order to show $\Pr[\mathsf{Bad}] \leq n \cdot \mathsf{Adv}^{\text{suf-ot-cma}}_{\mathsf{MAC},\mathsf{F}}(\lambda)$, we fix $i^* \in [n]$ and consider the event $\mathsf{Bad}^{(i^*)}$ that A issues a decryption query $\mathsf{ct} = (e, d, \tau)$ such that $\mathsf{ct} \neq \mathsf{ct}_{i^*}$, $e = e_{i^*}$, and $\mathsf{Vrfy}(\mathsf{k}_{i^*}^{mac}, d, \tau) = 1$. We construct a PPT algorithm $\mathsf{F}^{(i^*)}$ breaking the $\mathsf{sUF\text{-}OT\text{-}CMA}$ security of MAC, as follows: $\mathsf{F}^{(i^*)}$ is given the tagging oracle TAG and verification oracle VRFY of the $\mathsf{sUF\text{-}OT\text{-}CMA}$ security game. At the beginning of the security game, $\mathsf{F}^{(i^*)}$ generates $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$, chooses $2q_h$-wise independent hash function $f_{\mathsf{H}}, f_{\mathsf{H}_q}$ and a $2q_g$-wise independent hash function f_{G}, samples $(r_i, \hat{r}_i) \xleftarrow{\$} \mathcal{M}^{asy} \times \mathcal{R}^{asy}$, and computes $e_i \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_i; \hat{r}_i)$ for $i \in [n]$. It sets $I \leftarrow \emptyset$ and $\mathsf{win} \leftarrow 0$, and gives pk to A. When A submits a distribution $\mathcal{M}_{\mathrm{D}}$, $\mathsf{F}^{(i^*)}$ chooses $d_{i^*} \xleftarrow{\$} \mathcal{K}^{sym}$ and obtains τ_{i^*} by issuing d_{i^*} to the TAG oracle. For $i \in [n]\setminus\{i^*\}$, it chooses $(d_i, \mathsf{k}_i^{mac}) \xleftarrow{\$} \mathcal{K}^{sym} \times \mathcal{K}^{mac}$, and computes $\tau_i \leftarrow \mathsf{Tag}(\mathsf{k}_i^{mac}, d_i)$. Then, it returns $(\mathsf{ct}_i)_{i\in[n]}$, where $\mathsf{ct}_i = (e_i, d_i, \tau_i)$ for $i \in [n]$. The DEC, OPEN, G, and H oracles are simulated as follows:

- $\mathsf{DEC}(\mathsf{ct})$:
  1. Parse $\mathsf{ct} = (e, d, \tau)$.
  2. Halt and output $\mathsf{win} \leftarrow 1$ if $\mathsf{ct} \neq \mathsf{ct}_{i^*}$, $e = e_{i^*}$, and the VRFY oracle on input (d, τ) returns 1.
  3. Compute $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow f_{\mathsf{H}_q}(e)$.
  4. Return $\mathsf{m} \leftarrow \mathsf{k}^{sym} \oplus d$ if $\mathsf{Vrfy}(\mathsf{k}^{mac}, d, \tau) = 1$, and return $\perp$ otherwise.
- $\mathsf{OPEN}(i)$:
  1. Abort if $i = i^*$. Otherwise, set $I \leftarrow I \cup \{i\}$.
  2. Choose $\mathsf{m}_i \leftarrow \mathcal{M}_{\mathrm{D}}$.
  3. Set $\mathsf{G}(r_i) \leftarrow \hat{r}_i$ and $\mathsf{H}(r_i, e_i) \leftarrow (d_i \oplus \mathsf{m}_i, \mathsf{k}_i^{mac})$.
  4. Return (m_i, r_i).
- $\mathsf{G}(r)$:
  1. Abort if $r = r_{i^*}$.
  2. Return $\hat{r}_i$ if $r = r_i$ for some $i \in [n]$.
  3. Return $f_{\mathsf{G}}(r)$.

- $\mathsf{H}(r, e)$:
  1. Abort if $r = r_{i^*}$.
  2. Return $(d_i \oplus \mathsf{m}_i, \mathsf{k}_i^{mac})$ if $(r, e) = (r_i, e_i)$ for some $i \in [n]$.
  3. Return $f_{\mathsf{H}_q}(e)$ if $\mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r; \mathsf{G}(r)) = e$.
  4. Return $f_{\mathsf{H}}(r, e)$.

If A outputs a value out, then $\mathsf{F}^{(i^*)}$ outputs win.

$\mathsf{F}^{(i^*)}$ perfectly simulates the environment of A. Furthermore, the winning condition of $\mathsf{F}^{(i^*)}$ is identical to the condition that $\mathsf{Bad}^{(i^*)}$ occurs. Thus, $\mathsf{F}^{(i^*)}$ wins in the $\mathsf{sUF\text{-}OT\text{-}CMA}$ security game if $\mathsf{Bad}^{(i^*)}$ occurs. Due to the union bound over $i^* \in [n]$, we have $|\Pr[W_6] - \Pr[W_7]| \leq n \cdot \mathsf{Adv}^{\text{suf-ot-cma}}_{\mathsf{MAC},\mathsf{F}}(\lambda)$.

Finally, we prove $\Pr[\mathsf{Expt}^{\text{ideal-so-cca}}_{\mathsf{PKE}_2^{hy},\mathsf{S}}(\lambda) \to 1] = \Pr[W_7]$ by constructing the PPT simulator S in the following way: S is given the open oracle $\overline{\mathsf{OPEN}}$ of the $\mathsf{IDEAL\text{-}SIM\text{-}SO\text{-}CCA}$ security game. At the beginning, S generates $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$ and chooses $2q_h$-wise independent hash functions $f_{\mathsf{H}}, f_{\mathsf{H}_q}$ and a $2q_g$-wise independent hash function f_{G}. In addition, it chooses $(r_i, \hat{r}_i) \xleftarrow{\$} \mathcal{M}^{asy} \times \mathcal{R}^{asy}$ and computes $e_i \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_i; \hat{r}_i)$ for every $i \in [n]$. Then, S sets $I \leftarrow \emptyset$ and gives pk to A. When A submits $\mathcal{M}_{\mathrm{D}}$, S receives $|\mathsf{m}_1|, \ldots, |\mathsf{m}_\ell|$ in the $\mathsf{IDEAL\text{-}SIM\text{-}SO\text{-}CCA}$ security game, chooses $(d_i, \mathsf{k}_i^{mac}) \xleftarrow{\$} \mathcal{K}^{sym} \times \mathcal{K}^{mac}$, and computes $\tau_i \leftarrow \mathsf{Tag}(\mathsf{k}_i^{mac}, d_i)$ for $i \in [n]$. Then, it returns $(\mathsf{ct}_i)_{i \in [n]}$, where $\mathsf{ct}_i = (e_i, d_i, \tau_i)$ for $i \in [n]$. The DEC, OPEN, G, and H oracles are simulated as follows:

- $\mathsf{DEC}(\mathsf{ct})$:
  1. Parse $\mathsf{ct} = (e, d, \tau)$.
  2. Return $\perp$ if $e \in \{e_i\}_{i \in [n] \setminus I}$.
  3. Compute $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow f_{\mathsf{H}_q}(e)$.
  4. Return $\mathsf{m} \leftarrow \mathsf{k}^{sym} \oplus d$ if $\mathsf{Vrfy}(\mathsf{k}^{mac}, d, \tau) = 1$. Return $\perp$ otherwise.
- $\mathsf{OPEN}(i)$:
  1. Set $I \leftarrow I \cup \{i\}$.
  2. Obtain m_i by accessing the given open oracle $\overline{\mathsf{OPEN}}$.
  3. Set $\mathsf{G}(r_i) \leftarrow \hat{r}_i$ and $\mathsf{H}(r_i, e_i) \leftarrow (d_i \oplus \mathsf{m}_i, \mathsf{k}_i^{mac})$.
  4. Return (m_i, r_i).
- $\mathsf{G}(r)$:
  1. Return $\hat{r}_i$ if $r = r_i$ for some $i \in [n]$.
  2. Return $f_{\mathsf{G}}(r)$.
- $\mathsf{H}(r, e)$:
  1. Return $(d_i \oplus \mathsf{m}_i, \mathsf{k}_i^{mac})$ if $(r, e) = (r_i, e_i)$ for some $i \in [n]$.
  2. Return $f_{\mathsf{H}_q}(e)$ if $\mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r; \mathsf{G}(r)) = e$.
  3. Return $f_{\mathsf{H}}(r, e)$.

When A outputs out, S halts and outputs $R(\mathcal{M}_{\mathrm{D}}, \mathsf{m}_1, \ldots, \mathsf{m}_n, I, out)$. S completely simulates the view of A by using the $\overline{\mathsf{OPEN}}$ oracle. Thus, we have $\Pr[\mathsf{Expt}^{\text{ideal-so-cca}}_{\mathsf{PKE}_2^{hy},\mathsf{S}}(\lambda) \to 1] = \Pr[W_7]$.

From the discussion above, we obtain

$$\mathsf{Adv}^{\text{sim-so-cca}}_{\mathsf{PKE}_2^{hy},\mathsf{A},\mathsf{S},R}(\lambda) \leq 2n\sqrt{(q_g + q_h) \cdot \mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{D}}(\lambda)} + \frac{8n(q_g + q_h)}{\sqrt{|\mathcal{M}^{asy}|}} + \frac{2q_h}{\sqrt{|\mathcal{M}^{asy}|}} + 4q_g\sqrt{\delta}.$$

The proof is completed. $\square$

Proof of Lemma 1. In order to prove Lemma 1, we consider security games $\mathsf{Hybrid}^{(0)}, \mathsf{Hybrid}^{(1)}, \ldots, \mathsf{Hybrid}^{(n)}$. Let $\mathsf{Hybrid}^{(0)}$ be Game_5 in Theorem 2. For $i^* \in [n]$, we define $\mathsf{Hybrid}^{(i^*)}$ as the same game as $\mathsf{Hybrid}^{(i^*-1)}$ except for the following:

- At the beginning of the game, the challenger chooses $r_{i^*} \xleftarrow{\$} \mathcal{M}^{asy}$ and $\hat{r}_{i^*} \xleftarrow{\$} \mathcal{R}^{asy}$, and computes $e_{i^*} \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_{i^*}; \hat{r}_{i^*})$.
- Given a distribution $\mathcal{M}_{\mathrm{D}}$, the challenger chooses $(d_{i^*}, \mathsf{k}_{i^*}^{mac}) \xleftarrow{\$} \mathcal{K}^{sym} \times \mathcal{K}^{mac}$ and computes $\tau_{i^*} \leftarrow \mathsf{Tag}(\mathsf{k}_{i^*}^{mac}, d_{i^*})$.
- The OPEN oracle on input i^* chooses $\mathsf{m}_{i^*} \leftarrow \mathcal{M}_{\mathrm{D}}$, sets $I \leftarrow I \cup \{i^*\}$, $\mathsf{G}(r_{i^*}) \leftarrow \hat{r}_{i^*}$, and $\mathsf{H}(r_{i^*}, e_{i^*}) \leftarrow (d_{i^*} \oplus \mathsf{m}_{i^*}, \mathsf{k}_{i^*}^{mac})$, and then returns $(\mathsf{m}_{i^*}, r_{i^*})$.

Notice that $\mathsf{Hybrid}^{(n)}$ is identical to Game_6. Furthermore, in order to show the indistinguishability between $\mathsf{Hybrid}^{(i^*-1)}$ and $\mathsf{Hybrid}^{(i^*)}$, we consider additional security games $\mathsf{Hybrid}_0^{(i^*)}, \ldots, \mathsf{Hybrid}_3^{(i^*)}$. In addition, we define $\mathsf{G}\backslash\{r_{i^*}\}$ and $\mathsf{H}\backslash\{r_{i^*}\}$ as random oracles which first query the semi-classical oracle $O^{SC}_{\{r_{i^*}\}}$ and then G and H, respectively. For $i^* \in \{0, \ldots, n\}$ and $j \in \{0, \ldots, 3\}$, let $W_{H,j}^{(i^*)}$ be the event that A outputs *out* such that $R(\mathcal{M}_{\mathrm{D}}, \mathsf{m}_1, \ldots, \mathsf{m}_n, I, out) = 1$ in $\mathsf{Hybrid}_j^{(i^*)}$. For $i^* \in \{0, \ldots, n\}$ and $j \in [3]$, let $\mathsf{Find}_j^{(i^*)}$ be the event that the semi-classical oracle $O^{SC}_{\{r_{i^*}\}}$ returns 1 in the same game as $\mathsf{Hybrid}_j^{(i^*)}$ except that G and H are replaced by $\mathsf{G}\backslash\{r_{i^*}\}$ and $\mathsf{H}\backslash\{r_{i^*}\}$, respectively.

$\underline{\mathsf{Hybrid}_0^{(i^*)}}$: This game is the same as $\mathsf{Hybrid}^{(i^*-1)}$.

$\underline{\mathsf{Hybrid}_1^{(i^*)}}$: This game is the same as $\mathsf{Hybrid}_0^{(i^*)}$ except for the following:

- At the beginning of the game, the challenger chooses $r_{i^*} \xleftarrow{\$} \mathcal{M}^{asy}$ and $\hat{r}_{i^*} \xleftarrow{\$} \mathcal{R}^{asy}$, and computes $e_{i^*} \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_{i^*}; \hat{r}_{i^*})$.
- Given a distribution $\mathcal{M}_{\mathrm{D}}$, the challenger chooses $(d_{i^*}, \mathsf{k}_{i^*}^{mac}) \xleftarrow{\$} \mathcal{K}^{sym} \times \mathcal{K}^{mac}$ and computes $\tau_{i^*} \leftarrow \mathsf{Tag}(\mathsf{k}_{i^*}^{mac}, d_{i^*})$. Then, it sets $\mathsf{G}(r_{i^*}) \leftarrow \hat{r}_{i^*}$ and $\mathsf{H}(r_{i^*}, e_{i^*}) \leftarrow (d_{i^*} \oplus \mathsf{m}_{i^*}, \mathsf{k}_{i^*}^{mac})$.

It is clear that the first change is conceptual. Regarding the second change, the values (d_{i^*}, τ_{i^*}) of the two games $\mathsf{Hybrid}_0^{(i^*)}$, $\mathsf{Hybrid}_1^{(i^*)}$ are identically distributed. Regarding setting $\mathsf{G}(r_{i^*})$ and $\mathsf{H}(r_{i^*}, e_{i^*})$, the probability of distinguishing these oracles in the two games is at most $4(q_g + q_h)/\sqrt{|\mathcal{M}^{asy}|}$, owing to [37, Lemma 13]. Hence, we have $\left|\Pr[W_{H,0}^{(i^*)}] - \Pr[W_{H,1}^{(i^*)}]\right| \leq 4(q_g + q_h)/\sqrt{|\mathcal{M}^{asy}|}$.

$\underline{\mathsf{Hybrid}_2^{(i^*)}}$: This game is the same as $\mathsf{Hybrid}_1^{(i^*)}$ except that the OPEN oracle on input i^* sets $\mathsf{G}(r_{i^*}) \leftarrow \hat{r}_{i^*}$ and $\mathsf{H}(r_{i^*}, e_{i^*}) \leftarrow (d_{i^*} \oplus \mathsf{m}_{i^*}, \mathsf{k}_{i^*}^{mac})$, and the challenger does not program these oracles when generating the ciphertext ct_{i^*}.

For $r \neq r_{i^*}$, the values $\mathsf{G}(r)$, $\mathsf{H}(r, e)$ of $\mathsf{Hybrid}_2^{(i^*)}$ are equal to those of $\mathsf{Hybrid}_1^{(i^*)}$. Thus, we have $\left|\Pr[W_{H,1}^{(i^*)}] - \Pr[W_{H,2}^{(i^*)}]\right| \leq 2\sqrt{(q_g + q_h)\Pr[\mathsf{Find}_2^{(i^*)}]}$ by applying Proposition 1. Notice that $\mathsf{Hybrid}_2^{(i^*)}$ is identical to $\mathsf{Hybrid}^{(i^*)}$. In order to show that the probability $\Pr[\mathsf{Find}_2^{(i^*)}]$ is negligible if PKE fulfills IND-CPA security, we consider the security game $\mathsf{Hybrid}_3^{(i^*)}$.

$\underline{\mathsf{Hybrid}_3^{(i^*)}}$: This game is the same as $\mathsf{Hybrid}_2^{(i^*)}$ except that we replace r_{i^*} by r'_{i^*} when generating e_{i^*}. In order to prove the indistinguishability between $\mathsf{Hybrid}_2^{(i^*)}$ and $\mathsf{Hybrid}_3^{(i^*)}$, we construct a PPT algorithm $\mathsf{D}^{(i^*)}$ breaking the IND-CPA security of PKE, as follows: $\mathsf{D}^{(i^*)}$ is given the public key pk^{asy} of PKE. At the beginning of the security game, it chooses $2q_h$-wise independent hash functions $f_{\mathsf{H}}, f_{\mathsf{H}_q}$ and a $2q_g$-wise independent hash function f_{G}, and does the following for $i \in [i^*]$:

- If $i = i^*$, choose $r_{i^*}, r'_{i^*} \xleftarrow{\$} \mathcal{M}^{asy}$ and obtain the challenge ciphertext e_{i^*} by issuing (r_{i^*}, r'_{i^*}) in the IND-CPA security game.

- If $i \leq i^* - 1$, choose $r_i \xleftarrow{\$} \mathcal{M}^{asy}$ and $\hat{r}_i \xleftarrow{\$} \mathcal{R}^{asy}$, and compute $e_i \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_i; \hat{r}_i)$.

Then, $\mathsf{D}^{(i^*)}$ sets $I \leftarrow \emptyset$ and $\mathsf{find} \leftarrow 0$, and gives $\mathsf{pk} := \mathsf{pk}^{asy}$ to A. When A submits a distribution $\mathcal{M}_{\mathrm{D}}$, $\mathsf{D}^{(i^*)}$ does the following for $i \in [n]$:

- If $i \leq i^*$, choose $(d_i, \mathsf{k}_i^{mac}) \xleftarrow{\$} \mathcal{K}^{sym} \times \mathcal{K}^{mac}$ and compute $\tau_i \leftarrow \mathsf{Tag}(\mathsf{k}_i^{mac}, d_i)$.
- If $i \geq i^*+1$, choose $\mathsf{m}_i \leftarrow \mathcal{M}_{\mathrm{D}}$ and $r_i \xleftarrow{\$} \mathcal{M}^{asy}$, and compute $e_i \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_i; \mathsf{G}(r_i))$, $(\mathsf{k}_i^{sym}, \mathsf{k}_i^{mac}) \leftarrow \mathsf{H}(r_i, e_i)$, $d_i \leftarrow \mathsf{k}_i^{sym} \oplus \mathsf{m}_i$, and $\tau_i \leftarrow \mathsf{Tag}(\mathsf{k}_i^{mac}, d_i)$.

Then, $\mathsf{D}^{(i^*)}$ sets $\mathsf{ct}_i \leftarrow (e_i, d_i, \tau_i)$ for $i \in [n]$ and returns $(\mathsf{ct}_i)_{i\in[n]}$. In addition, the DEC, OPEN, G, and H oracles are simulated as follows:

- $\mathsf{DEC}(\mathsf{ct})$:
  1. Parse $\mathsf{ct} = (e, d, \tau)$.
  2. Compute $(\mathsf{k}^{sym}, \mathsf{k}^{mac}) \leftarrow f_{\mathsf{H}_q}(e)$.
  3. Return $\mathsf{m} \leftarrow \mathsf{k}^{sym} \oplus d$ if $\mathsf{Vrfy}(\mathsf{k}^{mac}, d, \tau) = 1$, and return $\perp$ otherwise.
- $\mathsf{OPEN}(i)$:
  1. Abort if $i = i^*$, and set $I \leftarrow I \cup \{i\}$ otherwise.
  2. If $i \leq i^* - 1$, choose $\mathsf{m}_i \leftarrow \mathcal{M}_{\mathrm{D}}$ and set $\mathsf{G}(r_i) \leftarrow \hat{r}_i$ and $\mathsf{H}(r_i, e_i) \leftarrow (d_i \oplus \mathsf{m}_i, \mathsf{k}_i^{mac})$.
  3. Return (m_i, r_i).
- $\mathsf{G}(r)$: Set $\mathsf{find} \leftarrow 1$ if the semi-classical oracle $O^{SC}_{\{r_{i^*}\}}$ on input a given quantum query returns 1.
  1. Return $\hat{r}_i$ if $r = r_i$ for some $i \leq i^* - 1$.
  2. Return $f_{\mathsf{G}}(r)$.
- $\mathsf{H}(r, e)$: Set $\mathsf{find} \leftarrow 1$ if the semi-classical oracle $O^{SC}_{\{r_{i^*}\}}$ on input a given quantum query returns 1.
  1. Return $(d_i \oplus \mathsf{m}_i, \mathsf{k}_i^{mac})$ if $(r, e) = (r_i, e_i)$ for some $i \leq i^* - 1$.
  2. Return $f_{\mathsf{H}_q}(e)$ if $\mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r) = e$.
  3. Return $f_{\mathsf{H}}(r, e)$.

Finally, when A outputs *out*, then $\mathsf{D}^{(i^*)}$ outputs find. We analyze the $\mathsf{D}^{(i^*)}$ algorithm. If $\mathsf{D}^{(i^*)}$ is given $e_{i^*} \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{asy}, r_{i^*})$, it simulates $\mathsf{Hybrid}_2^{(i^*)}$. If it is given $e_{i^*} \leftarrow \mathsf{Enc}^{asy}(\mathsf{pk}^{ays}, r'_{i^*})$, $\mathsf{Hybrid}_3^{(i^*)}$ is simulated. Hence, we have $\left|\Pr[\mathsf{Find}_2^{(i^*)}] - \Pr[\mathsf{Find}_3^{(i^*)}]\right| \leq \mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{D}^{(i^*)}}(\lambda)$.

Furthermore, in $\mathsf{Hybrid}_3^{(i^*)}$, the information of r'_{i^*} is given by only the G or H oracle. Thus, $\Pr[\mathsf{Find}_3^{(i^*)}] \leq 4(q_g+q_h)/|\mathcal{M}^{asy}|$ holds due to Proposition 2. Therefore, the probability of distinguishing between $\mathsf{Hybrid}^{(i^*-1)}$ and $\mathsf{Hybrid}^{(i^*)}$ is at most

$$\begin{aligned} 2\sqrt{(q_g+q_h)\cdot \mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{D}}(\lambda) + \frac{4(q_g+q_h)^2}{|\mathcal{M}^{asy}|}} + \frac{4(q_g+q_h)}{\sqrt{|\mathcal{M}^{asy}|}} &\leq 2\sqrt{(q_g+q_h)\cdot \mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{D}}(\lambda)} + \frac{4(q_g+q_h)}{\sqrt{|\mathcal{M}^{asy}|}} + \frac{4(q_g+q_h)}{\sqrt{|\mathcal{M}^{asy}|}} \\ &= 2\sqrt{(q_g+q_h)\cdot \mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{D}}(\lambda)} + \frac{8(q_g+q_h)}{\sqrt{|\mathcal{M}^{asy}|}}. \end{aligned}$$

From the discussion above, we obtain

$$|\Pr[W_5] - \Pr[W_6]| \leq 2n\sqrt{(q_g+q_h)\cdot \mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathsf{D}}(\lambda)} + \frac{8n(q_g+q_h)}{\sqrt{|\mathcal{M}^{asy}|}},$$

and the proof is completed. □

4 Conclusion

We presented two $\mathsf{SIM\text{-}SO\text{-}CCA}$ secure PKE schemes constructed from KEM schemes in the quantum random oracle model or quantum ideal cipher model. The first one PKE_1^{hy} meets the security in the quantum ideal cipher model. It is constructed from an $\mathsf{IND\text{-}CCA}$ secure KEM and a simulatable DEM with $\mathsf{OT\text{-}INT\text{-}CTXT}$ security. On the other hand, the second one PKE_2^{hy} meets the security in the quantum random oracle model. It is constructed from an FO-based KEM $\mathsf{FO}^{\not\perp}$ and an $\mathsf{sUF\text{-}OT\text{-}CMA}$ secure MAC. The differences between these schemes are as follows: It is possible to apply any $\mathsf{IND\text{-}CCA}$ secure KEM scheme to PKE_1^{hy}, while PKE_2^{hy} applies a particular KEM scheme $\mathsf{FO}^{\not\perp}$ to PKE_2^{hy}. In addition, it is possible to apply any deterministic MAC scheme to PKE_2^{hy}, while the underlying DEM scheme of PKE_1^{hy} needs to meet not only integrity but also simulatability (in the quantum ideal cipher model).

Acknowledgements. This work was in part supported by JSPS KAKENHI under Grant number JP18H03238. This research was in part conducted under a contract of "Research and development on new generation cryptography for secure wireless communication services" among "Research and Development for Expansion of Radio Wave Resources (JPJ000254)", which was supported by the Ministry of Internal Affairs and Communications, Japan. The authors would like to thank the anonymous referees of IMACC 2019 for their helpful comments.

References

[1] A. Ambainis, M. Hamburg, and D. Unruh. Quantum security proofs using semi-classical oracles. In *CRYPTO (2)*, volume 11693 of *LNCS*, pages 269–295. Springer, 2019.

[2] A. Ambainis, A. Rosmanis, and D. Unruh. Quantum attacks on classical proof systems: The hardness of quantum rewinding. In *FOCS*, pages 474–483. IEEE Computer Society, 2014.

[3] M. Bellare, R. Dowsley, B. Waters, and S. Yilek. Standard security does not imply security against selective-opening. In *EUROCRYPT*, volume 7237 of *LNCS*, pages 645–662. Springer, 2012.

[4] M. Bellare, D. Hofheinz, and S. Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In *EUROCRYPT*, volume 5479 of *LNCS*, pages 1–35. Springer, 2009.

[5] M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. *J. Cryptology*, 21(4):469–491, 2008.

[6] M. Bellare and P. Rogaway. Optimal asymmetric encryption. In *EUROCRYPT '94*, volume 950 of *LNCS*, pages 92–111. Springer, 1994.

[7] M. Bellare, B. Waters, and S. Yilek. Identity-based encryption secure against selective opening attack. In *TCC*, volume 6597 of *LNCS*, pages 235–252. Springer, 2011.

[8] M. Bellare and S. Yilek. Encryption schemes secure under selective opening attack. *IACR Cryptology ePrint Archive*, 2009:101, 2009.

[9] D. Boneh, Ö. Dagdelen, M. Fischlin, A. Lehmann, C. Schaffner, and M. Zhandry. Random oracles in a quantum world. In *ASIACRYPT*, volume 7073 of *LNCS*, pages 41–69. Springer, 2011.

[10] D. Boneh and M. Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. In *CRYPTO (2)*, volume 8043 of *LNCS*, pages 361–379. Springer, 2013.

[11] X. Boyen and Q. Li. All-but-many lossy trapdoor functions from lattices and applications. In *CRYPTO (3)*, volume 10403 of *LNCS*, pages 298–331. Springer, 2017.

[12] R. Cramer and V. Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *IACR Cryptology ePrint Archive*, 2001:108, 2001.

[13] J. Don, S. Fehr, C. Majenz, and C. Schaffner. Security of the fiat-shamir transformation in the quantum random-oracle model. In *CRYPTO (2)*, volume 11693 of *LNCS*, pages 356–383. Springer, 2019.

[14] S. Fehr, D. Hofheinz, E. Kiltz, and H. Wee. Encryption schemes secure against chosen-ciphertext selective opening attacks. In *EUROCRYPT*, volume 6110 of *LNCS*, pages 381–402. Springer, 2010.

[15] E. Fujisaki and T. Okamoto. Secure integration of asymmetric and symmetric encryption schemes. *J. Cryptology*, 26(1):80–101, 2013.

[16] B. Hemenway, B. Libert, R. Ostrovsky, and D. Vergnaud. Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In *ASIACRYPT*, volume 7073 of *LNCS*, pages 70–88. Springer, 2011.

[17] F. Heuer, T. Jager, E. Kiltz, and S. Schäge. On the selective opening security of practical public-key encryption schemes. In *Public Key Cryptography*, volume 9020 of *LNCS*, pages 27–51. Springer, 2015.

[18] F. Heuer and B. Poettering. Selective opening security from simulatable data encapsulation. In *ASIACRYPT (2)*, volume 10032 of *LNCS*, pages 248–277, 2016.

[19] R. Hiromasa. Digital signatures from the middle-product LWE. In *ProvSec*, volume 11192 of *LNCS*, pages 239–257. Springer, 2018.

[20] D. Hofheinz, K. Hövelmanns, and E. Kiltz. A modular analysis of the fujisaki-okamoto transformation. In *TCC (1)*, volume 10677 of *LNCS*, pages 341–371. Springer, 2017.

[21] D. Hofheinz, T. Jager, and A. Rupp. Public-key encryption with simulation-based selective-opening security and compact ciphertexts. In *TCC (B2)*, volume 9986 of *LNCS*, pages 146–168, 2016.

[22] D. Hofheinz, V. Rao, and D. Wichs. Standard security does not imply indistinguishability under selective opening. In *TCC (B2)*, volume 9986 of *LNCS*, pages 121–145, 2016.

[23] D. Hofheinz and A. Rupp. Standard versus selective opening security: Separation and equivalence results. In *TCC*, volume 8349 of *LNCS*, pages 591–615. Springer, 2014.

[24] A. Hosoyamada and K. Yasuda. Building quantum-one-way functions from block ciphers: Davies-meyer and merkle-damgård constructions. In *ASIACRYPT (1)*, volume 11272 of *LNCS*, pages 275–304. Springer, 2018.

[25] K. Hövelmanns, E. Kiltz, S. Schäge, and D. Unruh. Generic authenticated key exchange in the quantum random oracle model. *IACR Cryptology ePrint Archive*, 2018:928, 2018.

[26] A. Hülsing, J. Rijneveld, and F. Song. Mitigating multi-target attacks in hash-based signatures. In *Public Key Cryptography (1)*, volume 9614 of *LNCS*, pages 387–416. Springer, 2016.

[27] H. Jiang, Z. Zhang, L. Chen, H. Wang, and Z. Ma. IND-CCA-secure key encapsulation mechanism in the quantum random oracle model, revisited. In *CRYPTO (3)*, volume 10993 of *LNCS*, pages 96–125. Springer, 2018.

[28] H. Jiang, Z. Zhang, and Z. Ma. Key encapsulation mechanism with explicit rejection in the quantum random oracle model. In *Public Key Cryptography (2)*, volume 11443 of *LNCS*, pages 618–645. Springer, 2019.

[29] H. Jiang, Z. Zhang, and Z. Ma. Tighter security proofs for generic key encapsulation mechanism in the quantum random oracle model. In *PQCrypto*, volume 11505 of *LNCS*, pages 227–248. Springer, 2019.

[30] E. Kiltz, V. Lyubashevsky, and C. Schaffner. A concrete treatment of Fiat-Shamir signatures in the quantum random-oracle model. In *EUROCRYPT (3)*, volume 10822 of *LNCS*, pages 552–586. Springer, 2018.

[31] J. Lai, R. H. Deng, S. Liu, J. Weng, and Y. Zhao. Identity-based encryption secure against selective opening chosen-ciphertext attack. In *EUROCRYPT*, volume 8441 of *LNCS*, pages 77–92. Springer, 2014.

[32] B. Libert, A. Sakzad, D. Stehlé, and R. Steinfeld. All-but-many lossy trapdoor functions and selective opening chosen-ciphertext security from LWE. In *CRYPTO (3)*, volume 10403 of *LNCS*, pages 332–364. Springer, 2017.

[33] S. Liu and K. G. Paterson. Simulation-based selective opening CCA security for PKE from key encapsulation mechanisms. In *Public Key Cryptography*, volume 9020 of *LNCS*, pages 3–26. Springer, 2015.

[34] L. Lyu, S. Liu, S. Han, and D. Gu. Tightly SIM-SO-CCA secure public key encryption from standard assumptions. In *Public Key Cryptography (1)*, volume 10769 of *LNCS*, pages 62–92. Springer, 2018.

[35] T. Saito, K. Xagawa, and T. Yamakawa. Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In *EUROCRYPT (3)*, volume 10822 of *LNCS*, pages 520–551. Springer, 2018.

[36] E. E. Targhi and D. Unruh. Post-quantum security of the Fujisaki-Okamoto and OAEP transforms. In *TCC (B2)*, volume 9986 of *LNCS*, pages 192–216, 2016.

[37] D. Unruh. Revocable quantum timed-release encryption. In *EUROCRYPT*, volume 8441 of *LNCS*, pages 129–146. Springer, 2014.