Tighter security proofs for generic key encapsulation mechanism in the quantum random oracle model

Haodong Jiang[1,2,4], Zhenfeng Zhang[2,3], and Zhi Ma[1,4]

[1] State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, Henan, China
[2] TCA Laboratory, State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, China
[3] University of Chinese Academy of Sciences, Beijing, China
[4] Henan Key Laboratory of Network Cryptography Technology, Zhengzhou, Henan, China
hdjiang13@gmail.com, zfzhang@tca.iscas.ac.cn, ma_zhi@163.com

Abstract. In (TCC 2017), Hofheinz, Hövelmanns and Kiltz provided a fine-grained and modular toolkit of generic key encapsulation mechanism (KEM) constructions, which were widely used among KEM submissions to NIST Post-Quantum Cryptography Standardization project. The security of these generic constructions in the quantum random oracle model (QROM) has been analyzed by Hofheinz, Hövelmanns and Kiltz (TCC 2017), Saito, Xagawa and Yamakawa (Eurocrypt 2018), and Jiang et al. (Crypto 2018). However, the security proofs from standard assumptions are far from tight. In particular, the factor of security loss is q and the degree of security loss is 2, where q is the total number of adversarial queries to various oracles.

In this paper, using semi-classical oracle technique recently introduced by Ambainis, Hamburg and Unruh (ePrint 2018/904), we improve the results in (Eurocrypt 2018, Crypto 2018) and provide tighter security proofs for generic KEM constructions from standard assumptions. More precisely, the factor of security loss q is reduced to be $\sqrt{q}$. In addition, for transformation T that turns a probabilistic public-key encryption (PKE) into a determined one by derandomization and re-encryption, the degree of security loss 2 is reduced to be 1. Our tighter security proofs can give more confidence to NIST KEM submissions where these generic transformations are used, e.g., CRYSTALS-Kyber etc.

Keywords: quantum random oracle model · key encapsulation mechanism · generic construction

1 Introduction

Indistinguishability against chosen-ciphertext attacks (IND-CCA) [1] is widely accepted as a standard security notion for a key encapsulation mechanism (KEM). Random oracle model (ROM) [2] is an idealized model, where a hash

function is idealized to be a publicly accessible random oracle (RO). Generic constructions of IND-CCA-secure KEMs in the ROM are well studied by Dent [3] and Hofheinz, Hövelmanns and Kiltz [4]. Essentially, these generic constructions are categorized as variants of Fujisaki-Okamoto (FO) transformation (denote these transformations by FO transformations for brevity) [5, 6], including $\mathrm{FO}^{\not\perp}$, $\mathrm{FO}^{\perp}$, $\mathrm{FO}^{\not\perp}_m$, $\mathrm{FO}^{\perp}_m$, $\mathrm{QFO}^{\not\perp}_m$ and $\mathrm{QFO}^{\perp}_m$, where m[5] (without m) means $K = H(m)$ ($K = H(m, c)$), $\not\perp$ ($\perp$) means implicit (explicit) rejection, FO denotes the class of transformations that turn a PKE with standard security (one-wayness against chosen-plaintext attacks (OW-CPA) or indistinguishability against chosen-plaintext attacks (IND-CPA)) into an IND-CCA KEM, Q means an additional Targhi-Unruh hash [7] (a length-preserving hash function that has the same domain and range size) is added into the ciphertext, and variants of REACT/GEM transformation [8, 9] (denote these transformations by modular FO transformations), including $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}^{\not\perp}_m$, $\mathrm{U}^{\perp}_m$, $\mathrm{QU}^{\not\perp}_m$ and $\mathrm{QU}^{\perp}_m$, where U denotes the class of transformations that turn a PKE with non-standard security (e.g., OW-PCA, one-way against plaintext checking attack [8, 9]) or a deterministic PKE (DPKE, where the encryption algorithm is deterministic) into an IND-CCA-secure KEM.

Recently, post-quantum security of these generic transformations has gathered great interest [4, 10–15] due to the widespread adoption [11, Table 1] in KEM submissions to NIST Post-Quantum Cryptography Standardization Project [16], of which the goal is to standardize new public-key cryptographic algorithms with security against quantum adversaries. Quantum adversaries may execute all offline primitives such as hash functions on arbitrary superpositions, which motivated the introduction of quantum random oracle model (QROM) [17]. As Boneh et al. have argued [17], for fully evaluating the post-quantum security, the analysis in the QROM is crucial.

When proving a security of a cryptographic scheme S under a hardness assumption of a problem P, we usually construct a reduction algorithm $\mathcal{A}$ against P that uses an adversary $\mathcal{B}$ against S as a subroutine. Let (T, ϵ) and (T', ϵ') denote the running times and advantages of $\mathcal{A}$ and $\mathcal{B}$, respectively. The reduction is said to be tight if $T \approx T'$ and $\epsilon \approx \epsilon'$. Otherwise, if $T \gg T'$ or $\epsilon \ll \epsilon'$, the reduction is non-tight. Generally, the tightness gap, (informally) defined by $\frac{T\epsilon'}{T'\epsilon}$ [18], is used to measure the quality of a reduction. Tighter reductions with smaller tightness gap are desirable for practice cryptography especially in large-scale scenarios, since the tightness of a reduction determines the strength of the security guarantees provided by the security proof.

In [4, 10, 11] and this work, all the security reductions for (modular) FO transformations in the QROM satisfy (1) T is about T', i.e., $T \approx T'$; (2) $\epsilon' \approx \kappa\epsilon^{\frac{1}{\tau}}$, where κ and τ in the following are respectively denoted as the factor and degree of security loss[6]. Let q be the total number of adversarial queries to various oracles.

[5] The message m here is picked at random from the message space of underlying PKE.

[6] When comparing the tightness of different reductions, we assume perfect correctness of underlying scheme for brevity.

– In [4], Hofheinz et al. presented QROM security reductions for $\mathrm{QFO}_m^{\not\perp}$ and $\mathrm{QFO}_m^{\perp}$ with $\kappa = q^{\frac{3}{2}}$ and $\tau = 4$, for $\mathrm{QU}_m^{\not\perp}$ and $\mathrm{QU}_m^{\perp}$ with $\kappa = q$ and $\tau = 2$.
– In [10], Saito, Xagawa and Yamakawa presented a tight security proof (i.e., $\kappa = 1$ and $\tau = 1$) for $\mathrm{U}_m^{\not\perp}$ under a new (non-standard) security assumption called disjoint simulatability (DS). Moreover, two generic transformation, TPunc and KC, were given to construct a DS-secure DPKE from standard assumptions, with security reductions $\kappa = q$ and $\tau = 2$.
– In [11], Jiang et al. presented security reductions for $\mathrm{FO}^{\not\perp}$, $\mathrm{FO}_m^{\not\perp}$, T, $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}_m^{\not\perp}$ and $\mathrm{U}_m^{\perp}$ with $\kappa = q$ and $\tau = 2$.

As seen above, above security proofs of (modular) FO transformations from standard assumptions are far from tight. Recently, To better assess the security of lattice-based submissions, Ducas and Stehlé [19] suggested 10 questions that NIST should be asking the community. The 10-th question [19, Problem 10] is on this non-tightness in the QROM. To better understand this, they asked that

Can the tightness of those reductions be improved?

1.1 Our Contributions

In this paper, we give a **positive** answer and show that tightness of these reductions can be improved. Specifically, we provide tighter security proofs for these generic transformations in [4, 10] by using semi-classical oracle technique recently introduced by Ambainis, Hamburg and Unruh [20]. The improvements of the factor κ and the degree τ of security loss are summarized in Table 1. The detailed comparison with previous results in [10, 11] is shown in Table 2, where ϵ (ϵ') is the advantage of an adversary against security of underlying (resulting) cryptographic protimitive and δ is the correctness error (the probability of decryption failure in a legitimate execution of a scheme).

Table 1: Improvements of the factor κ and the degree τ of security loss.

(κ, τ)	TPunc, KC	T	$\mathrm{FO}_m^{\not\perp}$, $\mathrm{FO}^{\not\perp}$, $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$,$\mathrm{U}_m^{\not\perp}$, $\mathrm{U}_m^{\perp}$
SXY18 [10]	$(q, 2)$	–	–
JZC+18 [11]	–	$(q, 2)$	$(q, 2)$
Our work	$(\sqrt{q}, 2)$	$(q, 1)$	$(\sqrt{q}, 2)$

1. For $\mathrm{FO}_m^{\not\perp}$ and $\mathrm{FO}^{\not\perp}$, the security loss factor q in [11] is reduced to be $\sqrt{q}$. Specifically, we give a reduction from IND-CPA security of underlying PKE to IND-CCA security of resulting KEM with $\epsilon' \approx \sqrt{q\epsilon} + q\sqrt{\delta}$, which is tighter than $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$ in [11] from OW-CPA security of underlying PKE.

Table 2: Comparisons between previous works [10, 11] and our work.

Transformations	SXY18 [10]	Our results
PKE′=TPunc(PKE, G)	IND-CPA⇒DS $\epsilon' \approx q\sqrt{\epsilon}$	IND-CPA⇒DS $\epsilon' \approx \sqrt{q\epsilon}$
DPKE′=KC(DPKE, H)	OW-CPA⇒DS $\epsilon' \approx q\sqrt{\epsilon}$	OW-CPA⇒DS $\epsilon' \approx \sqrt{q\epsilon}$
Transformations	**JZC$^+$18 [11]**	**Our results**
PKE′=T(PKE, G)	OW-CPA⇒OW-qPCA $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$	IND-CPA⇒OW-qPCA $\epsilon' \approx q\epsilon + q\sqrt{\delta}$
KEM-I=FO$^{\not\perp}_m$(PKE, G, H, f)	OW-CPA⇒IND-CCA $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$	IND-CPA⇒IND-CCA $\epsilon' \approx \sqrt{q\epsilon} + q\sqrt{\delta}$
KEM-II=FO$^{\not\perp}$(PKE, G, H)	OW-CPA⇒IND-CCA $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$	IND-CPA⇒IND-CCA $\epsilon' \approx \sqrt{q\epsilon} + q\sqrt{\delta}$
KEM-III=U$^{\not\perp}$(PKE′, H)	OW-qPCA⇒IND-CCA $\epsilon' \approx q\sqrt{\epsilon}$	OW-qPCA⇒IND-CCA $\epsilon' \approx \sqrt{q\epsilon + q\delta}$
KEM-IV=U$^{\perp}$(PKE′, H)	OW-qPVCA⇒IND-CCA $\epsilon' \approx q\sqrt{\epsilon}$	OW-qPVCA⇒IND-CCA $\epsilon' \approx \sqrt{q\epsilon + q\delta}$
KEM-V=U$^{\not\perp}_m$(DPKE′, H)	OW-CPA⇒IND-CCA $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$	OW-CPA⇒IND-CCA $\epsilon' \approx \sqrt{q\epsilon + q\delta}$
KEM-VI=U$^{\perp}_m$(DPKE′, H)	OW-VA⇒IND-CCA $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$	OW-VA⇒IND-CCA $\epsilon' \approx \sqrt{q\epsilon + q\delta}$

2. For T, the quadratic security loss is reduced to be a linear one. Particularly, we provide a reduction from IND-CPA security of underlying PKE to OW-qPCA security of resulting PKE with $\epsilon' \approx q\epsilon + q\sqrt{\delta}$, while previous reduction in [11] is from OW-CPA security of underlying PKE with $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$.
3. For TPunc and KC, the security loss factor q in [10] is reduced to be $\sqrt{q}$. Both IND-CPA security of underlying PKE and OW-CPA of underlying DPKE can be reduced to DS security of DPKE by TPunc and KC with $\epsilon' \approx \sqrt{q\epsilon}$[7], respectively. While, under the same assumptions, previous reductions [10] are with $\epsilon' \approx q\sqrt{\epsilon}$.
4. For U$^{\not\perp}$, U$^{\perp}$, U$^{\not\perp}_m$ and U$^{\perp}_m$, the security loss factor q in [11] is also reduced to $\sqrt{q}$. Particularly, OW-qPCA (one-way against quantum plaintext checking attacks) security and OW-qPVCA (one-way against quantum plaintext and (classical) validity checking attacks) security of underlying PKE, OW-CPA security and OW-VA (one-way against validity checking attacks) security of underlying DPKE can be reduced to IND-CCA security of resulting KEM with $\epsilon' \approx \sqrt{q\epsilon + q\delta}$. While, under the same assumptions, previous reductions in [11] are with $\epsilon' \approx q\sqrt{\epsilon}$ or $\epsilon' \approx q\sqrt{\epsilon} + q\sqrt{\delta}$.

[7] Here, for TPunc and KC, we just follow [10] and assume the perfect correctness of underlying PKE.

According to [11, Table 1], our results directly apply to the NIST KEM submissions [16], including CRYSTALS-Kyber, LAC, SABER, SIKE and LEDAkem, and provide tighter reductions than previous known [4, 11]. For the submissions [16] where $\text{QFO}_m^{\not\perp}$ and $\text{QFO}^{\not\perp}$ are adopted, including FrodoKEM, KINDI, Lizard, NewHope, OKCN-AKCN-CNKE, Round2, Titanium, BIG QUAKE and LEDAkem, our results also provide tighter reductions than [11] without requiring the additional Targhi-Unruh hash.

1.2 Technique

In security proofs of (modular) FO transformations [4, 11, 10], reprogramming random oracle is an important trick. The security loss in current proofs [4, 11, 10] arises from the reprogramming of quantum random oracle. Here, we focus on the techniques of improving the analysis of quantum random oracle programming.

One way to hiding (OW2H) lemma [21, Lemma 6.2] is a practical tool to prove the indistinguishability between games where the random oracles are reprogrammed. Roughly speaking, OW2H lemma states that the distinguishing advantage $|P_{left} - P_{right}|$ of an oracle algorithm $A^{\mathcal{O}}$ that issuing at most q queries to an oracle $\mathcal{O}$ distinguishes Left ($\mathcal{O}$ is not reprogrammed) from Right ($\mathcal{O}$ is reprogrammed at x^*), can be bounded by $2q\sqrt{P_{guess}}$, that is

$$|P_{left} - P_{right}| \leq 2q\sqrt{P_{guess}}, \tag{1}$$

where P_{guess} is the success probability of another oracle algorithm B guessing x^* by running $A^{\mathcal{O}}$ and measuring one of $A^{\mathcal{O}}$'s query uniformly at random. To apply OW2H lemma to prove the security of some certain cryptographic schemes, [11, 10, 13] generalized the OW2H lemma. However, these generalizations do not give tighter bounds.

Very recently, Ambainis et al. [20] further improved the OW2H lemma by giving higher flexibility as well as tighter bounds. Specifically, a new technique called semi-classical oracle was developed, and semi-classical OW2H lemma was given with tighter bounds. Informally, a semi-classical oracle $\mathcal{O}_{x^*}^{SC}$ measures the output $|f_{x^*}(x)\rangle$ instead of $|x\rangle$, where $f_{x^*}(x) = 1$ if $x = x^*$ and 0 otherwise. Let $\mathcal{O}\backslash x^*$ be an oracle that first queries semi-classical $\mathcal{O}_{x^*}^{SC}$ and then $\mathcal{O}$. Semi-classical OW2H lemma shows that above $|P_{left} - P_{right}|$ can be bounded by $2\sqrt{qP_{find}}$, i.e.,

$$|P_{left} - P_{right}| \leq 2\sqrt{qP_{find}}, \tag{2}$$

where P_{find} is the probability of the event Find that semi-classical oracle $\mathcal{O}_{x^*}^{SC}$ ever outputs 1 during the execution $A^{\mathcal{O}\backslash x^*}$.

Next, we show how to use above semi-classical OW2H lemma to improve the security proofs of (modular) FO transformations [11, 10]. The primal obstacle is the simulation of the semi-classical oracle $\mathcal{O}_{x^*}^{SC}$, which is quantumly accessable. In particular, the key is simulation of f_{x^*}. We overcome this by making the best of specific properties of different FO-like KEM constructions. Specifically, in security proofs of (modular) FO transformations, x^* is instantiated with m^* of which the encryption is exactly challenge ciphertext c^*.

- For KC, $\mathrm{U}_m^{\not\perp}$ and $\mathrm{U}_m^{\perp}$, underlying PKE is deterministic. $f_{m^*}(m)$ can be simulated by verifying whether the encryption of m is c^*.
- For $U^{\not\perp}$ and $U^{\perp}$, underlying PKE satisfies OW-qPCA security or OW-qPVCA security. $f_{m^*}(m)$ can be simulated by verifying whether $\mathrm{PCO}(m, c^*) = 1$, where $\textsc{Pco}(m, c)$ is the plaintext checking oracle that returns 1 iff decryption of ciphertext c yields message m.
- For TPunc, underlying PKE satisfies IND-CPA security. We note that in IND-CPA security game, $m^* \in \{m_0, m_1\}$, where m_0 and m_1 are chosen by the adversary. Thus, the simulator can simulate f_{m^*} by setting $m^* = m_0$ or $m^* = m_1$. This trick comes from [20, Sec. 4.2], where Ambainis et al. argued the hardness of inverting a random oracle with leakage.
- For T, $\mathrm{FO}_m^{\not\perp}$, $\mathrm{FO}^{\not\perp}$, OW-CPA security of underlying PKE is assumed in previous security proofs in [4, 11], where OW2H lemma is used. When using semi-classical OW2H lemma, we need to a stronger assumption of underlying PKE, IND-CPA security, to follow above mentioned trick to simulate f_{m^*}.

Directly utilizing semi-classical OW2H lemma with bound (2) instead of OW2H lemma with bound (1), we improve the security reductions for $\mathrm{FO}_m^{\not\perp}$, $\mathrm{FO}^{\not\perp}$, TPunc, KC, $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}_m^{\not\perp}$ and $\mathrm{U}_m^{\perp}$, and reduce security loss factor from q to $\sqrt{q}$.

By introducing Bures distance, Ambainis et al. [20] also gave another tighter bound,

$$\left|\sqrt{P_{left}} - \sqrt{P_{right}}\right| \leq 2\sqrt{qP_{find}}. \tag{3}$$

Apparently, as pointed by [20], if P_{right} is negligible, i.e., $P_{right} \approx 0$, we can *approximately* have $|P_{left}| \lessapprox 4qP_{find}$. In the security proofs of (modular) FO transformations, roughly speaking, P_{left} is the success probability of an adversary against resulting cryptographic scheme, P_{right} is the corresponding "target probability" (typically, 0 or 1/2) specified by concrete security definition, and P_{find} is the success probability of another adversary against underlying primitive. Note that for OW-qPCA security, the "target probability" $P_{right} = 0$. Thus, using semi-classical OW2H lemma with bound (3), we can further reduce quadratic security loss in the proof of T in [11] to a linear one.

2 Preliminaries

Symbol description λ is denoted as a security parameter. $\mathcal{K}$, $\mathcal{M}$, $\mathcal{C}$ and $\mathcal{R}$ are denoted as key space, message space, ciphertext space and randomness space, respectively. Denote the sampling of a uniformly random element x in a finite set X by $x \xleftarrow{\$} X$. Denote the sampling from some distribution D by $x \leftarrow D$. $x =? y$ is denoted as an integer that is 1 if $x = y$, and otherwise 0. $\Pr[P : G]$ is the probability that the predicate P holds true where free variables in P are assigned according to the program in G. Denote deterministic (probabilistic) computation of an algorithm A on input x by $y := A(x)$ ($y \leftarrow A(x)$). Let $|X|$ be the cardinality of set X. A^H means that the algorithm A gets access to the oracle

H. $f \circ g(\cdot)$ means $f(g(\cdot))$. Following the work [4], we also make the convention that the number q_H of the adversarial queries to an oracle H counts the total number of times H is executed in the experiment.

Note: All cryptographic primitives and corresponding security and correctness definitions used in this paper are presented in Appendix A.

2.1 Quantum Random Oracle Model

In this section, we will present several existing lemmas that we need in our security proofs.

Lemma 1 (Simulating the random oracle [22, Theorem 6.1]). *Let H be an oracle drawn from the set of $2q$-wise independent functions uniformly at random. Then the advantage any quantum algorithm making at most q queries to H has in distinguishing H from a truly random function is identically 0.*

Lemma 2 (Generic search problem [23, 24, 11]). *Let $\gamma \in [0,1]$. Let Z be a finite set. $F : Z \to \{0,1\}$ is the following function: For each z, $F(z) = 1$ with probability p_z ($p_z \leq \gamma$), and $F(z) = 0$ else. Let N be the function with $\forall z : N(z) = 0$. If an oracle algorithm A makes at most q quantum queries to F (or N), then $\left|\Pr[b = 1 : b \leftarrow A^F] - \Pr[b = 1 : b \leftarrow A^N]\right| \leq 2q\sqrt{\gamma}$.*

Semi-classical oracle. Roughly speaking, semi-classical oracle O_S^{SC} only measures the output $|f_S(x)\rangle$ but not the input $|x\rangle$, where f_S is the indicator function such that $f_S(x) = 1$ if $x \in S$ and 0 otherwise. Formally, for a query to O_S^{SC} with $\sum_{x,z} a_{x,z}|x\rangle|z\rangle$, $\mathcal{O}_S^{SC}$ does the following

1. initialize a single qubit L with $|0\rangle$,
2. transform $\sum_{x,z} a_{x,z}|x\rangle|z\rangle|0\rangle$ into $\sum_{x,z} a_{x,z}|x\rangle|z\rangle|f_S(x)\rangle$,
3. measure L.

Then, after performing this semi-classical measurement, the query state will become $\sum_{x,z:f_S(x)=y} a_{x,z}|x\rangle|z\rangle$ (non-normalized) if the measurement outputs y ($y \in 0,1$).

Lemma 3 (Semi-classical OW2H [20, Theorem 1]). *Let $S \subseteq X$ be random. Let $\mathcal{O}_1$, $\mathcal{O}_2$ be oracles with domain X and codomain Y such that $\mathcal{O}_1(x) = \mathcal{O}_2(x)$ for any $x \notin S$. Let z be a random bitstring. ($\mathcal{O}_1$, $\mathcal{O}_2$, S and z may have arbitrary joint distribution D.) Let $\mathcal{O}_S^{SC}$ be an oracle that performs the semi-classical measurements corresponding to the projectors M_y, where $M_y := \sum_{x\in X:f_S(x)=y} |x\rangle\langle x|$ ($y \in 0,1$). Let $\mathcal{O}_2\backslash S$ ("$\mathcal{O}_2$ punctured on S") be an oracle that first queries $\mathcal{O}_S^{SC}$ and then $\mathcal{O}_2$. Let $A^{\mathcal{O}_1}(z)$ be an oracle algorithm with query depth q. Denote* Find *as the event that in the execution of $A^{\mathcal{O}_2\backslash S}(z)$, $\mathcal{O}_S^{SC}$ ever outputs 1 during semi-classical measurements. Let*

$$
\begin{aligned}
P_{left} &:= \Pr[b = 1 : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, b \leftarrow A^{\mathcal{O}_1}(z)] \\
P_{right} &:= \Pr[b = 1 : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, b \leftarrow A^{\mathcal{O}_2}(z)] \\
P_{find} &:= \Pr[\mathsf{Find} : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, A^{\mathcal{O}_2\backslash S}(z)].
\end{aligned}
$$

Then $|P_{left} - P_{right}| \leq 2\sqrt{(q+1)P_{find}}$ *and* $\left|\sqrt{P_{left}} - \sqrt{P_{right}}\right| \leq 2\sqrt{(q+1)P_{find}}$. *The lemma also holds with bound* $\sqrt{(q+1)P_{find}}$ *for alternative definition of* $P_{right} = \Pr[b = 1 \wedge \neg \mathsf{Find} : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, b \leftarrow A^{\mathcal{O}_2 \setminus S}(z)]$.

Lemma 4 (Search in semi-classical oracle [20, Corollary 1]). *Suppose that* S *and* z *are independent, and that* A *is a* q*-query algorithm. Let* $P_{max} := \max_{x \in X} \Pr[x \in S]$. *Then* $\Pr[\mathsf{Find} : A^{O_S^{SC}}(z)] \leq 4q \cdot P_{max}$.

3 Improved security proofs for (modular) FO transformations

In [4], Hofheinz et al. proposed several (modular) FO transformations, including T, $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}^{\not\perp}_m$, $\mathrm{U}^{\perp}_m$, $\mathrm{FO}^{\not\perp}_m$ and $\mathrm{FO}^{\not\perp}$, of which the security in the QROM was proven by [11, 10]. However, except the one for $\mathrm{U}^{\not\perp}_m$ from DS security of underlying DPKE to IND-CCA security of resulting KEM [10], all the reductions are non-tight due to the usage of OW2H lemma. To achieve a DS-secure DPKE, [10] also gave two transformations, TPunc and KC, from an IND-CPA-secure PKE and a OW-CPA-secure DPKE, respectively. But, the security reductions for TPunc and KC are also non-tight due to the utilization of OW2H lemma.

In this section, we will show that if the underlying PKE is assumed to be IND-CPA-secure, tighter reductions for $\mathrm{FO}^{\not\perp}_m$ and T can be achieved by using semi-classical oracle technique in [20]. As discussed in Sec. 1.2, we can also use semi-classical oracle technique to obtain tighter security reductions for $\mathrm{FO}^{\not\perp}$, TPunc, KC, $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}^{\not\perp}_m$ and $\mathrm{U}^{\perp}_m$. We present them in Appendix D.

To a public-key encryption scheme PKE = (*Gen*, *Enc*, *Dec*) with message space $\mathcal{M}$ and randomness space $\mathcal{R}$, hash functions $G : \mathcal{M} \to \mathcal{R}$, $H : \mathcal{M} \to \mathcal{K}$ and a pseudorandom function (PRF) f with key space $\mathcal{K}^{prf}$, we associate KEM-I=$\mathrm{FO}^{\not\perp}_m$[PKE,$G$,$H$,$f$], see Fig. 1.

Gen'	$Encaps(pk)$	$Decaps(sk', c)$
1 : $(pk, sk) \leftarrow Gen$	1 : $m \xleftarrow{\$} \mathcal{M}$	1 : Parse $sk' = (sk, k)$
2 : $k \xleftarrow{\$} \mathcal{K}^{prf}$	2 : $c = Enc(pk, m; G(m))$	2 : $m' := Dec(sk, c)$
3 : $sk' := (sk, k)$	3 : $K := H(m)$	3 : **if** $Enc(pk, m'; G(m')) = c$
4 : **return** (pk, sk')	4 : **return** (K, c)	4 : **return** $K := H(m')$
		5 : **else return**
		6 : $K := f(k, c)$

Fig. 1: IND-CCA-secure KEM-I=$\mathrm{FO}^{\not\perp}_m$[PKE,$G$,$H$,$f$]

Theorem 1 (PKE IND-CPA $\overset{QROM}{\Rightarrow}$ KEM-I IND-CCA). *If* PKE *is* δ-correct*, for any IND-CCA* $\mathcal{B}$ *against KEM-I, issuing at most* q_D *queries to the*

decapsulation oracle DECAPS*, at most q_G (q_H) queries to the random oracle G (H) ($(q_G + q_H) \geq 1$), there exist an IND-CPA adversary $\mathcal{A}$ against* PKE *and an adversary $\mathcal{A}'$ against the security of* PRF *with at most q_D classical queries such that* $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-I}}(\mathcal{B}) \leq 2\sqrt{(q_G + q_H + 1)\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}) + 2\frac{(q_G+q_H+1)^2}{|\mathcal{M}|}} + \mathtt{Adv}_{\text{PRF}}(\mathcal{A}') + 4q_G\sqrt{\delta}$ *and the running time of $\mathcal{A}$ is about that of $\mathcal{B}$.*

Proof. Here, we follow the proof skeleton of [11, Theorem 2]. Let $\mathcal{B}$ be an adversary against the IND-CCA security of KEM-I, issuing at most q_D queries to the decapsulation oracle DECAPS, at most q_G (q_H) queries to the random oracle G (H). Denote Ω_G, Ω_H and Ω_{H_q} as the sets of all functions $G : \mathcal{M} \to \mathcal{R}$, $H : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ and $H_q : \mathcal{C} \to \mathcal{K}$, respectively. Consider the games $G_0 - G_9$ in Fig. 2. Although the games $G_0 - G_5$ are essentially the same with the games $G_0 - G_5$ in prior proof of [11, Theorem 2], we still outline them here for readability and completeness. In particular, to apply the semi-classical oracle techniques in [20], we introduce games $G_6 - G_9$, which are different from the proof of [11, Theorem 2], and essential for the improvement of tightness in this paper.

GAME G_0. Since game G_0 is exactly the IND-CCA game,

$$\left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - 1/2\right| = \mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-I}}(\mathcal{B}).$$

GAME G_1. In game G_1, the DECAPS oracle is changed that the pseudorandom function f is replaced by a random function H'_q. Obviously, any distinguisher between G_0 and G_1 can be converted into a distinguisher $\mathcal{A}'$ between f and H'_q with at most q_D classical queries. Thus,

$$\left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - \Pr[G_1^{\mathcal{B}} \Rightarrow 1]\right| \leq \mathtt{Adv}_{\text{PRF}}(\mathcal{A}').$$

Let G' be a random function such that $G'(m)$ is sampled according to the uniform distribution over $\mathcal{R}_{\text{good}}(pk, sk, m) := \{r \in \mathcal{R} : Dec(sk, Enc(pk, m; r)) = m\}$. Let $\Omega_{G'}$ be the set of all functions G'. Define $\delta(pk, sk, m) = \frac{|\mathcal{R} \setminus \mathcal{R}_{\text{good}}(pk,sk,m)|}{|\mathcal{R}|}$ as the fraction of bad randomness and $\delta(pk, sk) = \max_{m \in \mathcal{M}} \delta(pk, sk, m)$. With this notation $\delta = \mathbf{E}[\delta(pk, sk)]$, where the expectation is taken over $(pk, sk) \leftarrow Gen$.

GAME G_2. In game G_2, we replace G by G' that uniformly samples from "good" randomness at random, i.e., $G' \xleftarrow{\$} \Omega_{G'}$. Following the same analysis as in the proof of [11, Theorem 1], we can show that the distinguishing problem between G_1 and G_2 is essentially the distinguishing problem between G and G', which can be converted into a distinguishing problem between F_1 and F_2, where F_1 is a function such that $F_1(m)$ is sampled according to Bernoulli distribution $B_{\delta(pk,sk,m)}$, i.e., $\Pr[F_1(m) = 1] = \delta(pk, sk, m)$ and $\Pr[F_1(m) = 0] = 1 - \delta(pk, sk, m)$, and F_2 is a constant function that always outputs 0 for any input. Thus, conditioned on a fixed (pk, sk) we obtain by Lemma 2, $\left|\Pr[G_1^{\mathcal{B}} \Rightarrow 1 : (pk, sk)] - \Pr[G_2^{\mathcal{B}} \Rightarrow 1 : (pk, sk)]\right| \leq 2q_G\sqrt{\delta(pk, sk)}$. By averaging over $(pk, sk) \leftarrow Gen$ we finally obtain

$$\left|\Pr[G_1^{\mathcal{B}} \Rightarrow 1] - \Pr[G_2^{\mathcal{B}} \Rightarrow 1]\right| \leq 2q_G\mathbf{E}[\sqrt{\delta(pk, sk)}] \leq 2q_G\sqrt{\delta}.$$

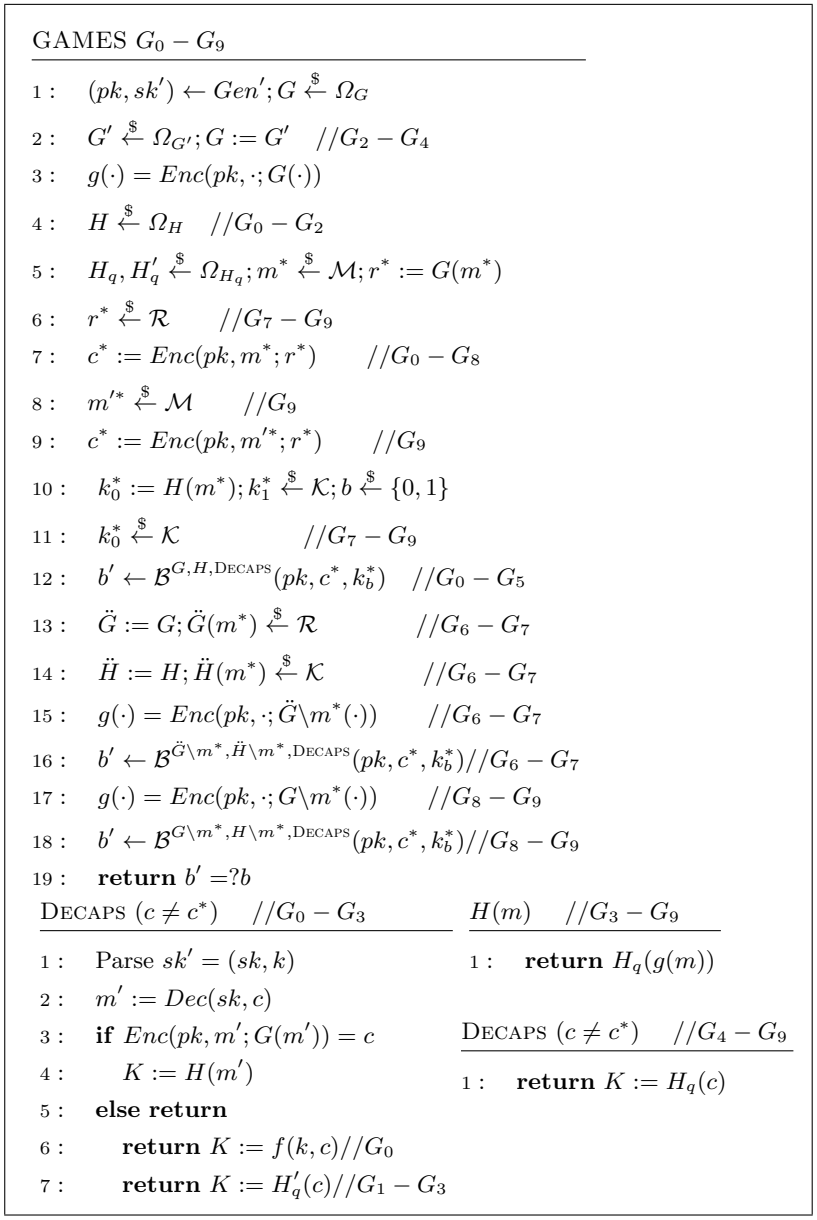

Fig. 2: Games G_0-G_9 for the proof of Theorem 1

GAME G_3. In G_3, H is substituted with $H_q \circ g$, where $g(\cdot) = Enc(pk, \cdot; G(\cdot))$. Since the G in this game only samples "good" randomness, the function g is injective. Thus, $H_q \circ g$ is a perfect random function. Therefore, G_2 and G_3 are statistically indistinguishable and we have $\Pr[G_2^{\mathcal{B}} \Rightarrow 1] = \Pr[G_3^{\mathcal{B}} \Rightarrow 1]$.

GAME G_4. In game G_4, the DECAPS oracle is changed that it makes no use of the secret key sk' any more. When $\mathcal{B}$ queries the DECAPS oracle on c ($c \neq c^*$), $K := H_q(c)$ is returned as the response. Let $m' := Dec(sk, c)$ and consider the following two cases.

Case 1: $Enc(pk, m'; G(m')) = c$. In this case, $H(m') = H_q(c)$ and both DECAPS oracles in G_3 and G_4 return the same value.

Case 2: $Enc(pk, m'; G(m')) \neq c$. In this case, $H'_q(c)$ and $H_q(c)$ are respectively returned in G_3 and G_4. In G_3, $H'_q(c)$ is uniformly random and independent of the oracles G and H in $\mathcal{B}$'s view. In G_4, queries to H can only reveal $H_q(\hat{c})$, where $\hat{c}$ satisfies $g(\hat{m}) = \hat{c}$ for some $\hat{m}$. If there exists a $\hat{m}$ such that $Enc(pk, \hat{m}; G(\hat{m})) = c$, $\hat{m} = m'$ since G in this game only samples from "good" randomness. Thus, $Enc(pk, m'; G(m')) = c$ will contradict the condition $Enc(pk, m'; G(m')) \neq c$. Consequently, $H_q(c)$ is also a fresh random key just like $H'_q(c)$ in $\mathcal{B}$'s view. Hence, in this case, the output distributions of the DECAPS oracles in G_3 and G_4 are identical in $\mathcal{B}$'s view.

As a result, the output distributions of G_3 and G_4 are statistically indistinguishable and we have $\Pr[G_3^{\mathcal{B}} \Rightarrow 1] = \Pr[G_4^{\mathcal{B}} \Rightarrow 1]$.

GAME G_5. In game G_5, we replace G' by G, that is, G in this game is reset to be an ideal random oracle. Then, following the same analysis as in bounding the difference between G_1 and G_2, we can have

$$\left|\Pr[G_4^{\mathcal{B}} \Rightarrow 1] - \Pr[G_5^{\mathcal{B}} \Rightarrow 1]\right| \leq 2q_G\sqrt{\delta}.$$

Let $\ddot{G}$ ($\ddot{H}$) be the function such that $\ddot{G}(m^*)$ ($\ddot{H}(m^*)$) is picked uniformly at random from $\mathcal{R}$ ($\mathcal{K}$), and $\ddot{G} = G$ ($\ddot{H} = H$) everywhere else. In the proof of [11, Theorem 2], G and H in game G_5 are directly reprogrammed to $\ddot{G}$ and $\ddot{H}$, respectively, and then the OW2H lemma is used to argue the indistinguishability.

Here, in order to use the semi-classical OW2H lemma, we reprogram G and H in game G_5 with an additional semi-classical oracle. Thereby, we need to consider the simulation of such a semi-classical oracle, which is unnecessary in the proof of [11, Theorem 2]. As discussed in Sec. 1.2, this semi-classical oracle can be simulated under the IND-CPA security assumption. Thus, we present following gamehops from G_6 to G_9.

GAME G_6. In game G_6, replace G and H by $\ddot{G}\backslash m^*$ and $\ddot{H}\backslash m^*$ respectively. For $\mathcal{B}$'s query to $\ddot{G}\backslash m^*$ ($\ddot{H}\backslash m^*$), $\ddot{G}\backslash m^*$ ($\ddot{H}\backslash m^*$) will first query a semi-classical oracle $\mathcal{O}^{SC}_{m^*}$, i.e., perform a semi-classical measurement, and then query $\ddot{G}$ ($\ddot{H}$). Let Find be the event that $\mathcal{O}^{SC}_{m^*}$ ever outputs 1 during semi-classical measurements of $\mathcal{B}'s$ queries to $\ddot{G}\backslash m^*$ and $\ddot{H}\backslash m^*$. Note that if the event $\neg\mathsf{Find}$ that $\mathcal{O}^{SC}_{m^*}$ always outputs 0 happens, $\mathcal{B}$ never learns the values of $G(m^*)$ and $H(m^*)$ and bit b is independent of $\mathcal{B}$'s view. That is, $\Pr[G_6^{\mathcal{B}} \Rightarrow 1 : \neg\mathsf{Find}] = 1/2$. Hence,

$$\Pr[G_6^{\mathcal{B}} \Rightarrow 1 \wedge \neg\mathsf{Find} : G_6] = 1/2\Pr[\neg\mathsf{Find} \ : G_6] = 1/2(1 - \Pr[\mathsf{Find} \ : G_6]).$$

Let $(G\times H)(\cdot) = (G(\cdot), H(\cdot))$, $(\ddot{G}\times\ddot{H})(\cdot) = (\ddot{G}(\cdot), \ddot{H}(\cdot))$, and $(\ddot{G}\times\ddot{H})\backslash m^*(\cdot) = (\ddot{G}\backslash m^*(\cdot), \ddot{H}\backslash m^*(\cdot))$. If one wants to make queries to G (or H) by accessing to $G \times H$, he just needs to prepare a uniform superposition of all states in the output register responding to H (or G). The number of total queries to $G \times H$ is at most $q_G + q_H$.

$A^{G\times H}(pk, c^*, H(m^*), H_q)$	DECAPS $(c \neq c^*)$
1 : $k_0^* = H(m^*); k_1^* \xleftarrow{\$} \mathcal{K}; b \xleftarrow{\$} \{0,1\}$	1 : **return** $K := H_q(c)$
2 : $b' \leftarrow \mathcal{B}^{G,H,\text{DECAPS}}(pk, c^*, k_b^*)$	
3 : **return** $b' =? b$	

Fig. 3: $A^{G\times H}$ for the proof of Theorem 1.

Let $A^{G\times H}$ be an oracle algorithm on input $(pk, c^*, H(m^*), H_q)$[8] in Fig. 3. Sample pk, m^*, G, H_q, H and c^* in the same way as G_5 and G_6, i.e., $(pk, sk) \leftarrow Gen$, $m^* \xleftarrow{\$} \mathcal{M}$, $G \xleftarrow{\$} \Omega_G$, $H_q \xleftarrow{\$} \Omega_{H_q}$, $H := H_q \circ g$ and $c^* = Enc(pk, m^*; G(m^*))$. Then, $A^{G\times H}(pk, c^*, H(m^*), H_q)$ perfectly simulates G_5, and $A^{(\ddot{G}\times\ddot{H})\backslash m^*}(pk, c^*, H(m^*), H_q)$ perfectly simulates G_6. Applying Lemma 3 with $X = \mathcal{M}$, $Y = (\mathcal{R}, \mathcal{K})$, $S = \{m^*\}$, $\mathcal{O}_1 = G \times H$, $\mathcal{O}_2 = \ddot{G} \times \ddot{H}$ and $z = (pk, c^*, H(m^*), H_q)$ and A, we can have

$$\left|\Pr[G_5^{\mathcal{B}} \Rightarrow 1] - \Pr[G_6^{\mathcal{B}} \Rightarrow 1 \wedge \neg\mathsf{Find} : G_6]\right| \leq \sqrt{(q_G + q_H + 1)\Pr[\mathsf{Find} : G_6]}.$$

GAME G_7. In game G_7, replace $r^* := G(m^*)$ and $k_0^* := H(m^*)$ by $r^* \xleftarrow{\$} \mathcal{R}$ and $k_0^* \xleftarrow{\$} \mathcal{K}$. We do not care about $\mathcal{B}$'s output, but only whether the event Find happens. Note that in G_6 and G_7, there is no information of $(G(m^*), H(m^*))$ in the oracle $\ddot{G} \times \ddot{H}$. Thus, apparently, $\Pr[\mathsf{Find} : G_6] = \Pr[\mathsf{Find} : G_7]]$.

GAME G_8. In game G_8, replace $\ddot{G}$ and $\ddot{H}$ by G and H. Since $G(m^*)$ and $H(m^*)$ are never used in simulating $\mathcal{B}$'s view, such a replacement causes no difference from $\mathcal{B}$'s view and we have $\Pr[\mathsf{Find} : G_7] = \Pr[\mathsf{Find} : G_8]$.

GAME G_9. In game G_9, replace m^* by m'^*. Note that the information of m^* in this game only exists in the oracles $G\backslash m^*$ and $H\backslash m^*$. By Lemma 4,

$$\Pr[\mathsf{Find} : G_9] \leq 4(q_G + q_H/|\mathcal{M}|).$$

[8] Although H_q here is the whole truth table of H_q, it is just taken as an oracle to make queries (with at most q_H times) in algorithm A. Thus, we can also take H_q as an accessible oracle instead of a whole truth table.

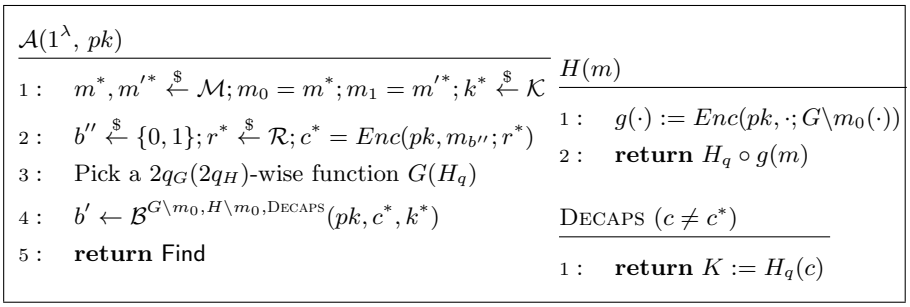

Fig. 4: Adversary $\mathcal{A}$ for the proof of Theorem 1

Next, we show that any adversary distinguishing G_8 from G_9 can be converted into an adversary against the IND-CPA security of underlying PKE. Construct an adversary $\mathcal{A}$ on input $(1^\lambda, pk)$ as in Fig. 4. Then, according to Lemma 1, if $b'' = 0$, $\mathcal{A}$ perfectly simulates G_8 and $\Pr[\mathsf{Find} : G_8] = \Pr[1 \leftarrow \mathcal{A} : b'' = 0]$. If $b'' = 1$, $\mathcal{A}$ perfectly simulates G_9 and $\Pr[\mathsf{Find} : G_9] = \Pr[1 \leftarrow \mathcal{A} : b'' = 1]$. Since $\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}) = 1/2\,|\Pr[1 \leftarrow \mathcal{A} : b'' = 0] - \Pr[1 \leftarrow \mathcal{A} : b'' = 1]|$,

$$|\Pr[\mathsf{Find} : G_8] - \Pr[\mathsf{Find} : G_9]| = 2\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}).$$

Finally, combing this with the bounds derived above, we have $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-I}}(\mathcal{B})$

$$\begin{aligned}
&\leq \mathtt{Adv}_{\text{PRF}}(\mathcal{A}') + 4q_G\sqrt{\delta} + 1/2\Pr[\mathsf{Find} : G_6] + \sqrt{(q_G + q_H + 1)\Pr[\mathsf{Find} : G_6]} \\
&\leq \mathtt{Adv}_{\text{PRF}}(\mathcal{A}') + 4q_G\sqrt{\delta} + \sqrt{2(q_G + q_H + 1)\Pr[\mathsf{Find} : G_6]} \\
&\leq \mathtt{Adv}_{\text{PRF}}(\mathcal{A}') + 4q_G\sqrt{\delta} + 2\sqrt{(q_G + q_H + 1)\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}) + 2\frac{(q_G + q_H + 1)^2}{|\mathcal{M}|}}.
\end{aligned}$$

□

The transformation T [25, 4] turns a probabilistic PKE into a determined one by derandomization and re-encryption [25, 26]. To a PKE=(Gen, Enc, Dec) with message space $\mathcal{M}$ and randomness space R, and a random oracle $G : \mathcal{M} \to \mathcal{R}$, we associate $\text{PKE}' = (Gen', Enc', Dec') = \text{T}[\text{PKE}, G]$, see Fig. 5. As discussed in Sec. 1.2, for T, using semi-classical OW2H lemma with bound (3), we can improve the reduction in [11] and reduce the quadratic security loss to a linear one. The complete proof of Theorem 2 is presented in Appendix B.

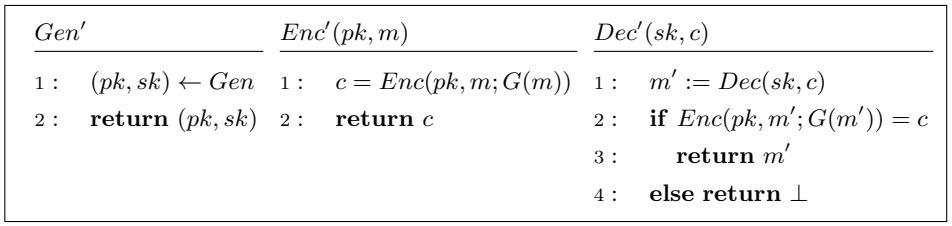

Fig. 5: OW-qPCA-secure $\text{PKE}' = \text{T}[\text{PKE}, G]$

Theorem 2 (PKE IND-CPA $\overset{QROM}{\Rightarrow}$ PKE′ OW-qPCA). *If* PKE *is* δ-correct*, for any OW-qPCA* $\mathcal{B}$ *against* PKE′*, issuing at most* q_G *quantum queries to the random oracle* G *and at most* q_P *quantum queries to the plaintext checking oracle* Pco*, there exists an IND-CPA adversary* $\mathcal{A}$ *against PKE such that* $\mathtt{Adv}^{\mathrm{OW}-q\mathrm{PCA}}_{\mathrm{PKE}'}(\mathcal{B}) \leq 4q_G\sqrt{\delta} + 2(q_G+2)\mathtt{Adv}^{\text{IND-CPA}}_{\mathrm{PKE}}(\mathcal{A}) + 4\frac{(q_G+2)^2}{|\mathcal{M}|}$ *and the running time of* $\mathcal{A}$ *is about that of* $\mathcal{B}$*.*

Acknowledgements. We are grateful to Dominique Unruh for interesting discussions on the one way to hiding lemma. We also thank Rainer Steinwandt, Fang Song, and anonymous reviewers of PQCrypto 2019 for their comments and suggestions. This work is supported by the National Key Research and Development Program of China (No. 2017YFB0802000), the National Natural Science Foundation of China (No. U1536205, 61472446, 61701539), and the National Cryptography Development Fund (mmjj20180107, mmjj20180212).

References

1. Rackoff, C., Simon, D.: Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In Feigenbaum, J., ed.: Advances in Cryptology – CRYPTO 1991. Volume 576 of LNCS., Springer (1992) 433–444
2. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V., eds.: Proceedings of the 1st ACM Conference on Computer and Communications Security – CCS 1993, ACM (1993) 62–73
3. Dent, A.W.: A designer's guide to KEMs. In Paterson, K.G., ed.: Cryptography and Coding: 9th IMA International Conference. Volume 2898 of LNCS., Springer-Verlag (2003) 133–151
4. Hofheinz, D., Hövelmanns, K., Kiltz, E.: A modular analysis of the Fujisaki-Okamoto transformation. In Kalai, Y., Reyzin, L., eds.: Theory of Cryptography - 15th International Conference – TCC 2017. Volume 10677 of LNCS., Springer (2017) 341–371
5. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. In Wiener, M.J., ed.: Advances in Cryptology – CRYPTO 1999. Volume 99 of LNCS., Springer (1999) 537–554
6. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. Journal of cryptology **26**(1) (2013) 1–22
7. Targhi, E.E., Unruh, D.: Post-quantum security of the Fujisaki-Okamoto and OAEP transforms. In Hirt, M., Smith, A.D., eds.: Theory of Cryptography Conference – TCC 2016-B. Volume 9986 of LNCS., Springer (2016) 192–216
8. Okamoto, T., Pointcheval, D.: REACT: Rapid enhanced-security asymmetric cryptosystem transform. In Naccache, D., ed.: Topics in Cryptology – CT-RSA 2001. Volume 2020 of LNCS., Springer (2001) 159–174
9. Jean-Sébastien, C., Handschuh, H., Joye, M., Paillier, P., Pointcheval, D., Tymen, C.: GEM: A generic chosen-ciphertext secure encryption method. In Preneel, B., ed.: Topics in Cryptology – CT-RSA 2002. Volume 2271 of LNCS., Springer (2002) 263–276

10. Saito, T., Xagawa, K., Yamakawa, T.: Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In Nielsen, J.B., Rijmen, V., eds.: Advances in Cryptology – EUROCRYPT 2018. Volume 10822 of LNCS. (2018) 520–551
11. Jiang, H., Zhang, Z., Chen, L., Wang, H., Ma, Z.: IND-CCA-secure key encapsulation mechanism in the quantum random oracle model, revisited. In Shacham, H., Boldyreva, A., eds.: Advances in Cryptology – CRYPTO 2018. Volume 10993 of LNCS. (2018) 96–125 https://eprint.iacr.org/2017/1096.
12. Bernstein, D.J., Persichetti, E.: Towards KEM unification. Cryptology ePrint Archive, Report 2018/526 (2018) https://eprint.iacr.org/2018/526.
13. Szepieniec, A., Reyhanitabar, R., Preneel, B.: Key encapsulation from noisy key agreement in the quantum random oracle model. Cryptology ePrint Archive, Report 2018/884 (2018) https://eprint.iacr.org/2018/884.
14. Hövelmanns, K., Kiltz, E., Schäge, S., Unruh, D.: Generic authenticated key exchange in the quantum random oracle model. Cryptology ePrint Archive, Report 2018/928 (2018) https://eprint.iacr.org/2018/928.
15. Xagawa, K., Yamakawa, T.: (tightly) QCCA-secure key-encapsulation mechanism in the quantum random oracle model. Cryptology ePrint Archive, Report 2018/838 (2018) https://eprint.iacr.org/2018/838.
16. NIST: National institute for standards and technology. Post quantum crypto project (2017) https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.
17. Boneh, D., Dagdelen, O., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In Lee, D.H., Wang, X., eds.: Advances in Cryptology – ASIACRYPT 2011. Volume 7073 of LNCS., Springer (2011) 41–69
18. Menezes, A.: Another look at provable security (2012) Invited Talk at EUROCRYPT 2012, https://www.iacr.org/cryptodb/archive/2012/EUROCRYPT/presentation/24260.pdf.
19. Ducas, L., Stehlé, D.: Assessing the security of lattice-based submissions: the 10 questions that NIST should be asking the community (2018) http://prometheuscrypt.gforge.inria.fr/2018-06-04.assessing-security.html.
20. Ambainis, A., Hamburg, M., Unruh, D.: Quantum security proofs using semi-classical oracles. Cryptology ePrint Archive, Report 2018/904 (2018) https://eprint.iacr.org/2018/904.
21. Unruh, D.: Revocable quantum timed-release encryption. Journal of the ACM **62**(6) (2015) 49:1–49:76
22. Zhandry, M.: Secure identity-based encryption in the quantum random oracle model. In Safavi-Naini, R., Canetti, R., eds.: Advances in Cryptology – CRYPTO 2012. Volume 7417 of LNCS., Springer (2012) 758–775
23. Ambainis, A., Rosmanis, A., Unruh, D.: Quantum attacks on classical proof systems: The hardness of quantum rewinding. In: 55th IEEE Annual Symposium on Foundations of Computer Science – FOCS 2014, IEEE (2014) 474–483
24. Hülsing, A., Rijneveld, J., Song, F.: Mitigating multi-target attacks in hash-based signatures. In Cheng, C., Chung, K., Persiano, G., Yang, B., eds.: Public-Key Cryptography – PKC 2016. Volume 9614 of LNCS., Springer (2016) 387–416
25. Bellare, M., Boldyreva, A., O'Neill, A.: Deterministic and efficiently searchable encryption. In Menezes, A., ed.: Advances in Cryptology – CRYPTO 2007. Volume 4622 of LNCS., Springer (2007) 535–552
26. Bellare, M., Halevi, S., Sahai, A., Vadhan, S.: Many-to-one trapdoor functions and their relation to public-key cryptosystems. In Krawczyk, H., ed.: Advances in Cryptology – CRYPTO 1998. Volume 1462 of LNCS., Springer (1998) 283–298

27. Persichetti, E.: Secure and anonymous hybrid encryption from coding theory. In Gaborit, P., ed.: Post-Quantum Cryptography - 5th International Workshop – PQCrypto 2013. Volume 7932 of LNCS., Springer (2013) 174–187

A Cryptographic Primitives

Definition 1 (Public-key encryption). *A public-key encryption scheme* PKE *consists of three algorithms. The key generation algorithm, Gen, is a probabilistic algorithm which on input* 1^λ *outputs a public/secret key-pair* (pk, sk)*. The encryption algorithm Enc, on input pk and a message* $m \in \mathcal{M}$*, outputs a ciphertext* $c \leftarrow Enc(pk, m)$*. If necessary, we make the used randomness of encryption explicit by writing* $c := Enc(pk, m; r)$*, where* $r \xleftarrow{\$} \mathcal{R}$ *(*$\mathcal{R}$ *is the randomness space). The decryption algorithm Dec, is a deterministic algorithm which on input sk and a ciphertext c outputs a message* $m := Dec(sk, c)$ *or a rejection symbol* $\perp \notin \mathcal{M}$*. A PKE is determined if Enc is deterministic. We denote* DPKE *to stand for a determined PKE.*

Definition 2 (Correctness [4]). *A public-key encryption scheme* PKE *is* δ-correct *if* $E[\max_{m \in \mathcal{M}} \Pr[Dec(sk, c) \neq m : c \leftarrow Enc(pk, m)]] \leq \delta$*, where the expectation is taken over* $(pk, sk) \leftarrow Gen$*. A PKE is perfectly correct if* $\delta = 0$*.*

Definition 3 (DS-secure DPKE [10]). *Let* $D_{\mathcal{M}}$ *denote an efficiently sampleable distribution on a set* $\mathcal{M}$*. A DPKE scheme (Gen,Enc,Dec) with plaintext and ciphertext spaces* $\mathcal{M}$ *and* $\mathcal{C}$ *is* $D_{\mathcal{M}}$*-disjoint simulatable if there exists a PPT algorithm S that satisfies the following,*

(1) Statistical disjointness:

$$\textsc{Disj}_{\text{PKE},S} := \max_{(pk,sk) \in Gen(1^\lambda; \mathcal{R}_{gen})} \Pr[c \in Enc(pk, \mathcal{M}) : c \leftarrow S(pk)]$$

is negligible, where $\mathcal{R}_{gen}$ *denotes a randomness space for Gen.*

(2) Ciphertext indistinguishability: For any PPT adversary $\mathcal{A}$*,*

$$\mathtt{Adv}^{\text{DS-IND}}_{\text{PKE},D_{\mathcal{M}},S}(\mathcal{A}) := \left| \begin{matrix} \Pr\left[\mathcal{A}(pk, c^*) \rightarrow 1 : \begin{matrix}(pk, sk) \leftarrow Gen; m^* \leftarrow D_{\mathcal{M}};\\ c^* = Enc(pk, m^*)]\end{matrix}\right] \\ - \Pr[\mathcal{A}(pk, c^*) \rightarrow 1 : (pk, sk) \leftarrow Gen; c^* \leftarrow S(pk)] \end{matrix} \right|$$

is negligible.

Definition 4 (OW-ATK-secure PKE). *Let* $\text{PKE} = (Gen, Enc, Dec)$ *be a public-key encryption scheme with message space* $\mathcal{M}$*. For* $\text{ATK} \in \{\text{CPA}, \text{VA}, \text{qPCA}, \text{qPVCA}\}$ *[11], we define OW-ATK games as in Fig. 6, where*

$$O_{ATK} := \begin{cases} \perp & \text{ATK} = \text{CPA} \\ \textsc{Val}(\cdot) & \text{ATK} = \text{VA} \\ \textsc{Pco}(\cdot, \cdot) & \text{ATK} = \text{qPCA} \\ \textsc{Pco}(\cdot, \cdot), \textsc{Val}(\cdot) & \text{ATK} = \text{qPVCA}. \end{cases}$$

Define the OW-ATK advantage function of an adversary $\mathcal{A}$ against PKE as $\mathtt{Adv}^{\text{OW-ATK}}_{\text{PKE}}(\mathcal{A}) := \Pr[\text{OW-ATK}^{\mathcal{A}}_{\text{PKE}} = 1]$.

Game OW-ATK	Pco(m,c)	Val(c)
1 : $(pk, sk) \leftarrow Gen$	1 : **if** $m \notin \mathcal{M}$	1 : $m := Dec(sk, c)$
2 : $m^* \xleftarrow{\$} \mathcal{M}$	2 : **return** $\perp$	2 : **if** $m \in \mathcal{M}$
3 : $c^* \leftarrow Enc(pk, m^*)$	3 : **else return**	3 : **return** 1
4 : $m' \leftarrow \mathcal{A}^{O_{\text{ATK}}}(pk, c^*)$	4 : $Dec(sk, c) =? m$	4 : **else return** 0
5 : **return** $m' =? m^*$		

Fig. 6: Games OW-ATK (ATK $\in$ {CPA, VA, qPCA, qPVCA}) for PKE, where O_{ATK} is defined in Definition 4. In games qPCA and qPVCA, the adversary $\mathcal{A}$ can query the Pco oracle with quantum state.

Game IND-CPA for PKE	Game IND-CCA for KEM	Decaps(sk, c)
1 : $(pk, sk) \leftarrow Gen$	1 : $(pk, sk) \leftarrow Gen$	1 : **if** $c = c^*$
2 : $b \leftarrow \{0, 1\}$	2 : $b \xleftarrow{\$} \{0, 1\}$	2 : **return** $\perp$
3 : $(m_0, m_1) \leftarrow \mathcal{A}(pk)$	3 : $(K^*_0, c^*) \leftarrow Encaps(pk)$	3 : **else return**
4 : $c^* \leftarrow Enc(pk, m_b)$	4 : $K^*_1 \xleftarrow{\$} \mathcal{K}$	4 : $K := Decaps(sk, c)$
5 : $b' \leftarrow \mathcal{A}(pk, c^*)$	5 : $b' \leftarrow \mathcal{A}^{\text{Decaps}}(pk, c^*, K^*_b)$	
6 : **return** $b' =? b$	6 : **return** $b' =? b$	

Fig. 7: IND-CPA game for PKE and IND-CCA game for KEM.

Definition 5 (IND-CPA-secure PKE). *Define* $\text{IND}-\text{CPA}$ *game of PKE as in Fig. 7 and the* $\text{IND}-\text{CPA}$ *advantage function of an adversary* $\mathcal{A}$ *against PKE as* $\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}) := \left|\Pr[\text{IND-CPA}^{\mathcal{A}}_{\text{PKE}} = 1] - 1/2\right|$.

Definition 6 (Key encapsulation). *A key encapsulation mechanism KEM consists of three algorithms. The key generation algorithm Gen outputs a key pair* (pk, sk)*. The encapsulation algorithm Encaps, on input* pk*, outputs a tuple* (K, c)*, where* $K \in \mathcal{K}$ *and* c *is said to be an encapsulation of the key* K*. The deterministic decapsulation algorithm Decaps, on input* sk *and an encapsulation* c*, outputs either a key* $K := Decaps(sk, c) \in \mathcal{K}$ *or a rejection symbol* $\perp \notin \mathcal{K}$*.*

Definition 7 (IND-CCA-secure KEM). *We define the* $\text{IND}-\text{CCA}$ *game as in Fig. 7 and the* $\text{IND}-\text{CCA}$ *advantage function of an adversary* $\mathcal{A}$ *against* KEM *as* $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM}}(\mathcal{A}) := \left|\Pr[\text{IND-CCA}^{\mathcal{A}}_{\text{KEM}} = 1] - 1/2\right|$.

B Proof of Theorem 2

Proof. Let $\mathcal{B}$ be an adversary against the OW-qPCA security of PKE′, issuing at most q_{PC} queries to the oracle PCO, at most q_G queries to the random oracle G. Denote Ω_G as the sets of all functions $G : \mathcal{M} \to \mathcal{R}$. Let G' be a random function such that $G'(m)$ is sampled according to the uniform distribution in $\mathcal{R}_{\text{good}}(pk, sk, m)$, where $\mathcal{R}_{\text{good}}(pk, sk, m) := \{r \in \mathcal{R} : Dec(sk, Enc(pk, m; r)) = m\}$. Let $\Omega_{G'}$ be the set of all functions G'. Let $\delta(pk, sk, m) = \frac{|\mathcal{R}_{\text{bad}}(pk,sk,m)|}{|\mathcal{R}|}$ as the fraction of bad randomness, where $\mathcal{R}_{\text{bad}}(pk, sk, m) = \mathcal{R} \setminus \mathcal{R}_{\text{good}}(pk, sk, m)$. $\delta(pk, sk) = \max_{m\in\mathcal{M}} \delta(pk, sk, m)$. $\delta = \mathbf{E}[\delta(pk, sk)]$, where the expectation is taken over $(pk, sk) \leftarrow Gen$. Consider the games in Fig. 8.

GAME G_0. Since game G_0 is exactly the OW-qPCA game,

$$\Pr[G_0^{\mathcal{B}} \Rightarrow 1] = \mathtt{Adv}^{OW-qPCA}_{\text{PKE}'}(\mathcal{B}).$$

GAMES $G_0 - G_6$

$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$
$G' \xleftarrow{\$} \Omega_{G'}; G := G' \quad //G_1 - G_2$
$m^* \xleftarrow{\$} \mathcal{M}$
$r^* := G(m^*)$
$r^* \xleftarrow{\$} \mathcal{R} \quad //G_6$
$c^* := Enc(pk, m^*; r^*)//G_0 - G_6$
$g(\cdot) := Enc(pk, \cdot; G(\cdot))//G_0 - G_4$
$m' \leftarrow \mathcal{B}^{G,\text{PCO}}(pk, c^*)//G_0 - G_4$
$\ddot{G} = G; \ddot{G}(m^*) \xleftarrow{\$} \mathcal{R}//G_5 - G_6$
$g(\cdot) := Enc(pk, \cdot; \ddot{G}\backslash m^*(\cdot))//G_5 - G_6$
$m' \leftarrow \mathcal{B}^{\ddot{G}\backslash m^*,\text{PCO}}(pk, c^*)//G_5 - G_6$
Query G with input $m'//G_4 - G_6$
return $m' =?m^*$

PCO(m,c) $//G_0 - G_1$

if $m \notin \mathcal{M}$
  return $\perp$
else return
  $Dec'(sk, c) =?m$

PCO (m, c) $//G_2 - G_6$

if $m \notin \mathcal{M}$
  return $\perp$
else return
  $g(m) =?c$

Fig. 8: Games G_0-G_6 for the proof of Theorem 2

GAME G_1. In game G_1, we replace G by G' that uniformly samples from "good" randomness at random, i.e., $G' \xleftarrow{\$} \Omega_{G'}$. Following the same analysis as in the proof of Theorem 1, we can have

$$\left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - \Pr[G_1^{\mathcal{B}} \Rightarrow 1]\right| \leq 2q_G\sqrt{\delta}.$$

GAME G_2. In game G_2, the PCO oracle is changed that it makes no use of the secret key sk any more. Particularly, when $\mathcal{B}$ queries PCO oracle, $Enc(pk, m; G(m)) = ?c$ is returned instead of $Dec'(sk, c) = ?m$. It is easy to verify that $Dec'(sk, c) = ?m$ is equal to $Dec(sk, c) = ?m \wedge Enc(pk, m; G(m)) = ?c$. Thus, the outputs of the PCO oracles in G_1 and G_2 merely differs for the case of $Dec(sk, c) \neq m$ and $Enc(pk, m; G(m)) = c$. But, such a case does not exist since G in this game only samples from "good" randomness. That is, the PCO oracle in G_2 always has the identical output with the one in G_1. Therefore, we have

$$\Pr[G_1^{\mathcal{B}} \Rightarrow 1] = \Pr[G_2^{\mathcal{B}} \Rightarrow 1].$$

GAME G_3. In game G_3, we switch the G that only samples from "good" randomness back to an ideal random oracle G. Then, similar to the case of G_0 and G_1, the distinguishing problem between G_2 and G_3 can also be converted to the distinguishing problem between G and G'. Using the same analysis method in bounding the difference between G_0 and G_1, we can have

$$\left|\Pr[G_2^{\mathcal{B}} \Rightarrow 1] - \Pr[G_3^{\mathcal{B}} \Rightarrow 1]\right| \leq 2q_G\sqrt{\delta}.$$

GAME G_4. In game G_4, an additional query to G with **classical** state $|m'\rangle|0\rangle$ is performed after $\mathcal{B}$ returns m'. Obviously, G_4 has the same output as G_3 and we have

$$\Pr[G_3^{\mathcal{B}} \Rightarrow 1] = \Pr[G_4^{\mathcal{B}} \Rightarrow 1].$$

Let $\ddot{G}$ be the function that $\ddot{G}(m^*) = \ddot{r}^*$, and $\ddot{G} = G$ everywhere else, where $\ddot{r}^*$ is picked uniformly at random from $\mathcal{R}$.

GAME G_5. In game G_5, we replace G by a semi-classical oracle $\ddot{G}\backslash m^*$. For a query input, $\ddot{G}\backslash m^*$ will first query $\mathcal{O}_{m^*}^{SC}$, i.e., perform a semi-classical measurement, and then query $\ddot{G}$. Let Find be the event that $\mathcal{O}_{m^*}^{SC}$ ever outputs 1 during semi-classical measurements of the queries to $\ddot{G}\backslash m^*$. We note that

$$\Pr[G_5^{\mathcal{B}} \Rightarrow 1 \wedge \neg\mathsf{Find} : G_5] = 0$$

since $G_5^{\mathcal{B}} \Rightarrow 1$ implies that $m' = m^*$ in G_5, and $\ddot{G}$ is **classically** queried at m' in G_5[9]. Applying Lemma 3 with $X = \mathcal{M}$, $Y = \mathcal{R}$, $S = \{m^*\}$, $\mathcal{O}_1 = G$, $\mathcal{O}_2 = \ddot{G}$ and $z = (pk, c^*)$, we can have

$$\left|\sqrt{\Pr[G_4^{\mathcal{B}} \Rightarrow 1]} - \sqrt{\Pr[G_5^{\mathcal{B}} \Rightarrow 1 \wedge \neg\mathsf{Find} : G_5]}\right| \leq \sqrt{(q_G + 2)\Pr[\mathsf{Find} : G_5]}.$$

GAME G_6. In game G_6, we replace $r^* := G(m^*)$ by $r^* \xleftarrow{\$} \mathcal{R}$. Since $G(m^*)$ is only used once and independent of the oracles $\ddot{G}$ and PCO,

$$\Pr[\mathsf{Find} : G_5] = \Pr[\mathsf{Find} : G_6].$$

[9] For a classical query input m^*, $\mathcal{O}_{m^*}^{SC}$ always outputs 1.

Note that $G(m^*)$ is never used in G_6, we can just replace $G \xleftarrow{\$} \Omega_G; \ddot{G} = G; \ddot{G}(m^*) \xleftarrow{\$} \mathcal{R}$ by $\ddot{G} \xleftarrow{\$} \Omega_G$. For brevity and readability, we will substitute the notation $\ddot{G}$ with notation G. Then, game G_6 can be rewritten as in Fig. 9.

GAMES G_6		GAMES G_7	
1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$	1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$
2 :	$m^* \xleftarrow{\$} \mathcal{M}; r^* \xleftarrow{\$} \mathcal{R}$	2 :	$m^*, m_1^* \xleftarrow{\$} \mathcal{M}; r^* \xleftarrow{\$} \mathcal{R}$
3 :	$c^* := Enc(pk, m^*; r^*)$	3 :	$c^* := Enc(pk, m_1^*; r^*)$
4 :	$g(\cdot) := Enc(pk, \cdot; G \backslash m^*(\cdot))$	4 :	$g(\cdot) := Enc(pk, \cdot; G \backslash m^*(\cdot))$
5 :	$m' \leftarrow \mathcal{B}^{G \backslash m^*, \mathrm{PCO}}(pk, c^*)$	5 :	$m' \leftarrow \mathcal{B}^{G \backslash m^*, \mathrm{PCO}}(pk, c^*)$
6 :	Query G with input m'	6 :	Query G with input m'
7 :	**return** $m' =? m^*$	7 :	**return** $m' =? m^*$

PCO (m, c) $\quad //G_6 - G_7$	
1 :	**if** $m \notin \mathcal{M}$ **return** $\perp$
2 :	**else return**
3 :	$g(m) =? c$

Fig. 9: Game G_6 and game G_7 for the proof of Theorem 2

$\mathcal{A}(1^\lambda, pk)$	
1 :	$m^*, m_1^* \xleftarrow{\$} \mathcal{M}; m_0 = m^*; m_1 = m_1^*$
2 :	$b'' \xleftarrow{\$} \{0,1\}; r^* \xleftarrow{\$} \mathcal{R}$
3 :	$c^* = Enc(pk, m_{b''}; r^*)$
4 :	Pick a $2q_G$-wise function G
5 :	$g(\cdot) := Enc(pk, \cdot; G \backslash m_0(\cdot))$
6 :	$m' \leftarrow \mathcal{B}^{G \backslash m_0, \mathrm{PCO}}(pk, c^*)$
7 :	Query G with input m'
8 :	**return** Find

PCO (m, c)	
1 :	**if** $m \notin \mathcal{M}$
2 :	**return** $\perp$
3 :	**else return**
4 :	$g(m) =? c$

Fig. 10: Adversary $\mathcal{A}$ for the proof of Theorem 2

Game G_7. In game G_7, we replace $c^* := Enc(pk, m^*; r^*)$ by $c^* := Enc(pk, m_1^*; r^*)$, where $m_1^* \xleftarrow{\$} \mathcal{M}$. Note that the information of m^* in this game only exists in the oracle $G \backslash m^*$, by Lemma 4 we have

$$\Pr[\mathsf{Find} : G_7] \leq 4\frac{q_G + 1}{|\mathcal{M}|}.$$

Next, we show that any adversary distinguishing G_6 from G_7 can be converted into an adversary against the IND-CPA security of underlying PKE scheme. Construct an adversary $\mathcal{A}$ on input $(1^\lambda, pk)$ as in Fig. 10, where Find is 1 if the event Find that $\mathcal{O}^{SC}_{m_0}$ ever outputs 1 during semi-classical measurements of the queries to $G\backslash m_0$ happens. Then, according to Lemma 1, if $b''=0$, $\mathcal{A}$ perfectly simulates G_6 and $\Pr[\mathsf{Find} : G_6] = \Pr[1 \leftarrow \mathcal{A} : b''=0]$. If $b''=1$, $\mathcal{A}$ perfectly simulates G_7 and $\Pr[\mathsf{Find} : G_7] = \Pr[1 \leftarrow \mathcal{A} : b''=1]$. Since $\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}) = 1/2\,|\Pr[1 \leftarrow \mathcal{A} : b''=0] - \Pr[1 \leftarrow \mathcal{A} : b''=1]|$,

$$|\Pr[\mathsf{Find} : G_6] - \Pr[\mathsf{Find} : G_7]| = 2\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}).$$

Finally, combing this with the bounds derived above, we can conclude that

$$\mathtt{Adv}^{\text{OW}-q\text{PCA}}_{\text{PKE}'}(\mathcal{B}) \leq 4q_G\sqrt{\delta} + 2(q_G+2)\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}) + 4\frac{(q_G+2)^2}{|\mathcal{M}|}.$$

□

C Auxiliary lemmas

We note that semi-classical OW2H lemma can not directly apply to the security proofs where underlying assumption is a *search* one (e.g., One-Wayness) since the semi-classical oracle can only return "Yes" (the event Find happens) or "no" (the event $\neg\mathsf{Find}$ happens). Indeed, intuitively, if Find happens, we can measure the register $|x\rangle$ of the query state $\sum_{x,z:f_S(x)=1} a_{x,z}|x\rangle|z\rangle$ to obtain a x such that $x \in S$.

Here, formally, we focus on the case where the set S is a singleton, i.e., $S = \{x^*\}$ ($x^* \in X$) and slightly modify the semi-classical OW2H lemma to make it applicable for security proofs with a *search* assumption.

Semi-classical oracle with auxiliary extractor. An auxiliary extractor L' with size $\lceil \log |X| \rceil$ is added to record x^*. For a query to O^{SC}_S with $\sum_{x,z} a_{x,z}|x\rangle|z\rangle$, $\mathcal{O}^{SC}_S$ does the following,

1. initialize a single qubit L and an auxiliary extractor L' with $|0\rangle$,
2. transform $\sum_{x,z} a_{x,z}|x\rangle|z\rangle|0\rangle|0\rangle$ into $\sum_{x,z} a_{x,z}|x\rangle|z\rangle|f_S(x)\rangle|f'_S(x)\rangle$, where $f'_S(x) = x \cdot f_S(x)$, that is $f'_S(x) = x$ if $x = x^*$ and 0 otherwise.
3. measure L and L'.

Note that the state of composite system made up of L and L' is a superposition of $|0\rangle|0\rangle$ and $|1\rangle|x^*\rangle$. Thus, if the measurement of L outputs 1 (0), the measurement of L' will output x^* (0). Given $x^* \in X$, we can view the composite system of L and L' as a new system $\tilde{L}$, $|0\rangle|0\rangle$ as $|\tilde{0}\rangle$, $|1\rangle|x^*\rangle$ as $|\tilde{1}\rangle$. Thus, following the proof of [20, Theorem 1], we can directly derive the following lemma.

Lemma 5 (Semi-classical OW2H with auxiliary extractor). *Let $S = \{x^*\}$ ($x^* \in X$) be random. Let $\mathcal{O}_1$, $\mathcal{O}_2$ be oracles with domain X and codomain Y such that $\mathcal{O}_1(x) = \mathcal{O}_2(x)$ for any $x \neq x^*$. Let z be a random bitstring. ($\mathcal{O}_1$, $\mathcal{O}_2$,*

S and z may have arbitrary joint distribution D.) Let $\mathcal{O}_S^{SC}$ be a semi-classical oracle with auxiliary extractor described above. Let $\mathcal{O}_2\backslash S$ ("$\mathcal{O}_2$ punctured on S") be an oracle that first queries $\mathcal{O}_S^{SC}$ and then $\mathcal{O}_2$.

Let $A^{\mathcal{O}_1}(z)$ be an oracle algorithm with query depth q. Denote Find as the event that in the execution of $A^{\mathcal{O}_2\backslash S}(z)$, $\mathcal{O}_S^{SC}$ ever outputs 1 and x^* during semi-classical measurements.

Let

$$\begin{aligned} P_{left} &:= \Pr[b = 1 : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, b \leftarrow A^{\mathcal{O}_1}(z)] \\ P_{right} &:= \Pr[b = 1 : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, b \leftarrow A^{\mathcal{O}_2}(z)] \\ P_{find} &:= \Pr[\mathsf{Find} : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, A^{\mathcal{O}_2\backslash S}(z)] \end{aligned}$$

Then $|P_{left} - P_{right}| \leq 2\sqrt{(q+1)P_{find}}$ and $\left|\sqrt{P_{left}} - \sqrt{P_{right}}\right| \leq 2\sqrt{(q+1)P_{find}}$. The lemma also holds with bound $\sqrt{(q+1)P_{find}}$ for the following alternative definition of P_{right},

$$P_{right} = \Pr[b = 1 \wedge \neg\mathsf{Find} : (\mathcal{O}_1, \mathcal{O}_2, S, z) \leftarrow D, b \leftarrow A^{\mathcal{O}_2\backslash S}(z)].$$

Lemma 6 ([11, Lemma 4][10, Lemma 2.2]). *Let Ω_H ($\Omega_{H'}$) be the set of all functions $H : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}^m$ ($H' : \{0,1\}^{n_2} \to \{0,1\}^m$). Let $H \xleftarrow{\$} \Omega_H$, $H' \xleftarrow{\$} \Omega_{H'}$, $x \xleftarrow{\$} \{0,1\}^{n_1}$. Let $F_0 = H(x, \cdot)$, $F_1 = H'(\cdot)$ Consider an oracle algorithm A^{H,F_i} that makes at most q queries to H and F_i ($i \in \{0,1\}$). If x is independent from the A^{H,F_i}'s view,*

$$\left|\Pr[1 \leftarrow A^{H,F_0}] - \Pr[1 \leftarrow A^{H,F_1}]\right| \leq 2q\frac{1}{\sqrt{2^{n_1}}}.$$

D Improved security proofs for remaining (modular) FO transformations

D.1 FO$^{\not\perp}$: From IND-CPA-secure PKE to IND-CCA-secure KEM

To a public-key encryption scheme PKE = (*Gen*, *Enc*, *Dec*) with message space $\mathcal{M}$, ciphertext space $\mathcal{C}$ and randomness space $\mathcal{R}$, hash functions $G : \mathcal{M} \to \mathcal{R}$ and $H : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$, we associate KEM-II=FO$^{\not\perp}$[PKE,$G$,$H$], see Fig. 11.

Different from the one in FO$^{\not\perp}_m$, the random oracle H in FO$^{\not\perp}$ takes both the plaintext m and the ciphertext c as input. Using the same proof technique in [11, Theorem 1], we can divide the H-inputs (m, c) into two categories, matched inputs and unmatched inputs, by judging whether $c = Enc(pk, m; G(m))$, and replace H by $H_q \circ g$ only for the matched inputs. Then, following the proofs of Theorem 1, we can derive Theorem 3.

Theorem 3 (PKE IND-CPA $\overset{QROM}{\Rightarrow}$ KEM-II IND-CCA). *If* PKE *is* δ-correct*, for any IND-CCA $\mathcal{B}$ against KEM-II, issuing at most q_D queries to the decapsulation oracle* DECAPS*, q_G (q_H) queries to the random oracle G (H),*

there exists an IND-CPA adversary $\mathcal{A}$ against PKE *such that* $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-II}}(\mathcal{B}) \leq 2q_H\frac{1}{\sqrt{|\mathcal{M}|}}+4q_G\sqrt{\delta}+2\sqrt{(q_G+q_H+1)\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A})+2\frac{(q_G+q_H+1)^2}{|\mathcal{M}|}}$ *and the running time of $\mathcal{A}$ is about that of $\mathcal{B}$.*

Gen'	$Encaps(pk)$	$Decaps(sk', c)$
1 : $(pk, sk) \leftarrow Gen$	1 : $m \xleftarrow{\$} \mathcal{M}$	1 : Parse $sk' = (sk, s)$
2 : $s \xleftarrow{\$} \mathcal{M}$	2 : $c = Enc(pk, m; G(m))$	2 : $m' := Dec(sk, c)$
3 : $sk' := (sk, s)$	3 : $K := H(m, c)$	3 : **if** $Enc(pk, m'; G(m')) = c$
4 : **return** (pk, sk')	4 : **return** (K, c)	4 : **return** $K := H(m', c)$
		5 : **else return**
		6 : $K := H(s, c)$

Fig. 11: IND-CCA-secure KEM-II=FO$^{\not\perp}$[PKE,G,H]

D.2 TPunc: From IND-CPA-secure PKE to DS-secure DPKE

Gen'	$Dec'(sk, c)$	$S(pk)$
1 : $(pk, sk) \leftarrow Gen$	1 : $m' := Dec(sk, c)$	1 : $r \xleftarrow{\$} \mathcal{R}$
2 : **return** (pk, sk)	2 : **if** $m' \notin \mathcal{M}'$	2 : $c = Enc(pk, 0; r)$
	3 : **return** $\perp$	3 : **return** c
$Enc'(pk, m)$, where $m \in \mathcal{M}'$	4 : **else return** m'	
1 : $c = Enc(pk, m; G(m))$		
2 : **return** c		

Fig. 12: DS-secure PKE$'$ = TPunc[PKE, G] with simulation S

The transformation TPunc that converts a perfectly-correct IND-CPA-secure PKE into a DS-secure DPKE, is a variant of T and proposed by [10]. The security of TPunc is proven in the QROM with non-tight reduction due to the usage of OW2H lemma. Here, we will improve the reduction using semi-classical OW2H lemma.

To a PKE=(Gen, Enc, Dec) with message space $\mathcal{M}$ and randomness space R, and a random oracle $G : \mathcal{M} \to \mathcal{R}$, we associate PKE$' = (Gen', Enc', Dec') =$ TPunc[PKE, G], where $\mathcal{M}' = \mathcal{M}\backslash\{0\}$, see Fig. 12.

Theorem 4 (PKE IND-CPA $\overset{QROM}{\Rightarrow}$ PKE$'$ DS). *Let S be the algorithm described in Fig. 12. Let $U_{\mathcal{M}'}$ be the uniform distribution over $\mathcal{M}'$. If PKE is perfectly correct,* $\textsc{Disj}_{\text{PKE}',S} = 0$. *Moreover, for any $\mathcal{B}$ against PKE$'$ issuing at*

most q_G quantum queries to G, there exist adversaries $\mathcal{A}_1$ and $\mathcal{A}_2$ against the IND-CPA security of PKE such that $\mathtt{Adv}^{\text{DS-IND}}_{\text{PKE}',U_{\mathcal{M}'},S}(\mathcal{B}) \leq 2\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}_2) + 2\sqrt{2(q_G+1)\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}_1) + 4\frac{(q_G+1)^2}{|\mathcal{M}|}}$*, and the running time of $\mathcal{A}_1$ ($\mathcal{A}_2$) is about that of $\mathcal{B}$.*

Remark: In [14], TPunc is modularized into two transformations, Punc (reduce IND-CPA security to probabilistic DS security tightly) and T (reduce probabilistic DS security to deterministic DS security non-tightly). The proof technique in Theorem 4 can also be trivially used to get a tighter reduction for T from probabilistic DS security to deterministic DS security.

Proof. $\text{DISJ}_{\text{PKE}',S} = 0$ has been proven in [10, Theorem 3.3]. Here, we focus on the upper bound of $\mathtt{Adv}^{\text{DS-IND}}_{\text{PKE}',U_{\mathcal{M}'},S}(\mathcal{B})$. Let $\mathcal{B}$ be an adversary against PKE′, issuing at most q_G queries to the random oracle G. Denote Ω_G as the sets of all functions $G : \mathcal{M} \to \mathcal{R}$.

Define games G_0 and G_1 as in Fig. 13. According to the definition of DS security, we have

$$\mathtt{Adv}^{\text{DS-IND}}_{\text{PKE}',U_{\mathcal{M}'},S}(\mathcal{B}) = \left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - \Pr[G_1^{\mathcal{B}} \Rightarrow 1]\right|.$$

GAME G_2. Replace $r^* := G(m^*)$ by $r^* \xleftarrow{\$} \mathcal{R}$. First, as [10, Theorem 3.3] has showed, we can construct an adversary $\mathcal{A}_2$ against the IND-CPA security of PKE as in Fig. 14. Apparently,

$$\left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - \Pr[G_2^{\mathcal{B}} \Rightarrow 1]\right| = 2\mathtt{Adv}^{\text{IND-CPA}}_{\text{PKE}}(\mathcal{A}_2).$$

GAMES G_0		GAMES $G_1 - G_2$	
1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$	1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$
2 :	$r^* \xleftarrow{\$} \mathcal{R}$	2 :	$m^* \xleftarrow{\$} \mathcal{M}'$
3 :	$c^* := Enc(pk, 0; r^*)$	3 :	$r^* := G(m^*)//G_1$
4 :	$b' \leftarrow \mathcal{B}^G(pk, c^*)$	4 :	$r^* \xleftarrow{\$} \mathcal{R}//G_2$
5 :	**return** b'	5 :	$c^* := Enc(pk, m^*; r^*)$
		6 :	$b' \leftarrow \mathcal{B}^G(pk, c^*)$
		7 :	**return** b'

Fig. 13: Games G_0-G_2 for the proof of Theorem 4

$\mathcal{A}_2$ against the IND-CPA game	
1 :	$(pk, sk) \leftarrow Gen; m_0 \xleftarrow{\$} \mathcal{M}'; m_1 := 0; r^* \xleftarrow{\$} \mathcal{R}; b \xleftarrow{\$} \{0,1\}$
2 :	$c^* := Enc(pk, m_b; r^*); b' \leftarrow \mathcal{A}(pk, c^*);$ **return** $b' =? b$

Fig. 14: $\mathcal{A}_2$ for the proof of Theorem 4

Then, we will use semi-classical OW2H lemma to bound $\left|\Pr[G_1^{\mathcal{B}} \Rightarrow 1] - \Pr[G_2^{\mathcal{B}} \Rightarrow 1]\right|$. Let $\ddot{G}$ be the function such that $\ddot{G}(m) = G(m)$ for $m \neq m^*$ and $\ddot{G}(m^*) \xleftarrow{\$} \mathcal{R}$. Then, it's easy to see that G_2 can be rewritten as G_3 in Fig. 15 and we have

$$\Pr[G_2^{\mathcal{B}} \Rightarrow 1] = \Pr[G_3^{\mathcal{B}} \Rightarrow 1]$$

GAME G_4. G_4 is the same as G_3 except that $\ddot{G}$ is replaced by a semi-classical oracle $\ddot{G}\backslash m^*$. For a query input, $\ddot{G}\backslash m^*$ will first query $\mathcal{O}^{SC}_{m^*}$, i.e., perform a semi-classical measurement, and then query $\ddot{G}$. Let Find be the event that $\mathcal{O}^{SC}_{m^*}$ ever outputs 1 during semi-classical measurements of the queries to $\ddot{G}\backslash m^*$.

GAMES G_3		GAMES G_4	
1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$	1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$
2 :	$m^* \xleftarrow{\$} \mathcal{M}'$	2 :	$m^* \xleftarrow{\$} \mathcal{M}'$
3 :	$r^* := G(m^*)$	3 :	$r^* := G(m^*)$
4 :	$\ddot{G} = G$	4 :	$\ddot{G} = G$
5 :	$\ddot{G}(m^*) \xleftarrow{\$} \mathcal{R}$	5 :	$\ddot{G}(m^*) \xleftarrow{\$} \mathcal{R}$
6 :	$c^* := Enc(pk, m^*; r^*)$	6 :	$c^* := Enc(pk, m^*; r^*)$
7 :	$b' \leftarrow \mathcal{B}^{\ddot{G}}(pk, c^*)$	7 :	$b' \leftarrow \mathcal{B}^{\ddot{G}\backslash m^*}(pk, c^*)$
8 :	**return** b'	8 :	**return** b'

Fig. 15: Games G_3-G_4 for the proof of Theorem 4

Applying Lemma 3 with $X = \mathcal{M}$, $Y = \mathcal{R}$, $S = \{m^*\}$, $\mathcal{O}_1 = G$, $\mathcal{O}_2 = \ddot{G}$ and $z = (pk, c^*)$, we can have

$$\left|\Pr[G_1^{\mathcal{B}} \Rightarrow 1] - \Pr[G_3^{\mathcal{B}} \Rightarrow 1]\right| \leq 2\sqrt{(q_G + 1)\Pr[\mathsf{Find} : G_4]}.$$

We note that in game G_4, $G(m^*)$ is only used in evaluating c^* and independent of $\ddot{G}$, we can replace $r^* := G(m^*)$ by $r^* \xleftarrow{\$} \mathcal{R}$ and then simplify G_4 as G_5 in Fig. 16. Since the output distributions of G_4 and G_5 are totally identical, we have

$$\Pr[G_4^{\mathcal{B}} \Rightarrow 1] = \Pr[G_5^{\mathcal{B}} \Rightarrow 1].$$

GAME G_6. In game G_6, we replace $m^* \xleftarrow{\$} \mathcal{M}'$ by $m^* \xleftarrow{\$} \mathcal{M}$. Since the statistical distance between uniform distributions on $\mathcal{M}'$ and $\mathcal{M}$ is $\frac{1}{|\mathcal{M}|}$, we have

$$\left|\Pr[G_5^{\mathcal{B}} \Rightarrow 1] - \Pr[G_6^{\mathcal{B}} \Rightarrow 1]\right| \leq \frac{1}{|\mathcal{M}|}.$$

GAMES $G_5 - G_6$		GAMES G_7	
1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$	1 :	$(pk, sk) \leftarrow Gen; G \xleftarrow{\$} \Omega_G$
2 :	$m^* \xleftarrow{\$} \mathcal{M}'//G_5$	2 :	$m^* \xleftarrow{\$} \mathcal{M}$
3 :	$m^* \xleftarrow{\$} \mathcal{M}//G_6$	3 :	$m_1^* \xleftarrow{\$} \mathcal{M}$
4 :	$r^* \xleftarrow{\$} \mathcal{R}$	4 :	$r^* \xleftarrow{\$} \mathcal{R}$
5 :	$c^* := Enc(pk, m^*; r^*)$	5 :	$c^* := Enc(pk, m_1^*; r^*)$
6 :	$b' \leftarrow \mathcal{B}^{G \backslash m^*}(pk, c^*)$	6 :	$b' \leftarrow \mathcal{B}^{G \backslash m^*}(pk, c^*)$
7 :	**return** b'	7 :	**return** b'

Fig. 16: Games G_5-G_7 for the proof of Theorem 4

$\mathcal{A}(1^\lambda, pk)$	
1 :	$m^*, m_1^* \xleftarrow{\$} \mathcal{M}; m_0 = m^*; m_1 = m_1^*$
2 :	$b'' \xleftarrow{\$} \{0,1\}; r^* \xleftarrow{\$} \mathcal{R}; c^* = Enc(pk, m_{b''}; r^*)$
3 :	Pick a $2q_G$-wise function G
4 :	$m' \leftarrow \mathcal{B}^{G \backslash m_0}(pk, c^*); pcreturn\mathsf{Find}$

Fig. 17: Adversary $\mathcal{A}_1$ for the proof of Theorem 4

GAME G_7. In game G_7, we replace $c^* := Enc(pk, m^*; r^*)$ by $c^* := Enc(pk, m_1^*; r^*)$, where $m_1^* \xleftarrow{\$} \mathcal{M}$. Note that the information of m^* in this game only exists in the oracle $G \backslash m^*$, by Lemma 4 we have

$$\Pr[\mathsf{Find} \ : G_7] \leq 4\frac{q_G}{|\mathcal{M}|}.$$

Next, we show that any adversary distinguishing G_6 from G_7 can be converted into an adversary against the IND-CPA security of underlying PKE. Construct an adversary $\mathcal{A}_1$ on input $(1^\lambda, pk)$ as in Fig. 17, where Find returns 1 iff the event Find that $\mathcal{O}_{m_0}^{SC}$ ever outputs 1 during semi-classical measurements happens. Then, according to Lemma 1, if $b'' = 0$, $\mathcal{A}_1$ perfectly simulates G_6 and $\Pr[\mathsf{Find} \ : G_6] = \Pr[1 \leftarrow \mathcal{A}_1 : b'' = 0]$. If $b'' = 1$, $\mathcal{A}$ perfectly simulates G_7 and $\Pr[\mathsf{Find} \ : G_7] = \Pr[1 \leftarrow \mathcal{A}_1 : b'' = 1]$. Since $\mathtt{Adv}_{\mathrm{PKE}}^{\text{IND-CPA}}(\mathcal{A}) = 1/2\,|\Pr[1 \leftarrow \mathcal{A}_1 : b'' = 0] - \Pr[1 \leftarrow \mathcal{A}_1 : b'' = 1]|$,

$$|\Pr[\mathsf{Find} \ : G_6] - \Pr[\mathsf{Find} \ : G_7]| = 2\mathtt{Adv}_{\mathrm{PKE}}^{\text{IND-CPA}}(\mathcal{A}_1).$$

Finally, combing this with the bounds derived above, we have $\mathtt{Adv}_{\mathrm{PKE}',U_{\mathcal{M}'},S}^{\text{DS-IND}}(\mathcal{B})$

$$\leq 2\mathtt{Adv}_{\mathrm{PKE}}^{\text{IND-CPA}}(\mathcal{A}_2) + 2\sqrt{2(q_G+1)\mathtt{Adv}_{\mathrm{PKE}}^{\text{IND-CPA}}(\mathcal{A}_1) + 4\frac{(q_G+1)q_G}{|\mathcal{M}|} + \frac{q_G+1}{|\mathcal{M}|}}$$

$$\leq 2\mathtt{Adv}_{\mathrm{PKE}}^{\text{IND-CPA}}(\mathcal{A}_2) + 2\sqrt{2(q_G+1)\mathtt{Adv}_{\mathrm{PKE}}^{\text{IND-CPA}}(\mathcal{A}_1) + 4\frac{(q_G+1)^2}{|\mathcal{M}|}}$$

□

D.3 KC: from OW-CPA-secure DPKE to DS-secure DPKE

Following the idea of "plaintext confirmation" that attaches an additional hash of plaintext to ciphertext, [10] proposed a transformation KC from OW-CPA-Secure DPKE to DS-Secure DPKE. The security reduction of KC in the QROM is non-tight due to the usage of OW2H lemma. Here, we will use semi-classical OW2H lemma to improve the tightness in [10].

$\underline{Gen'}$

1 : $(pk, sk) \leftarrow Gen$
2 : **return** (pk, sk)

$\underline{Enc'(pk, m)}$

1 : $c_1 = Enc(pk, m)$
2 : $c_2 = H(m)$
3 : **return** $c = (c_1, c_2)$

$\underline{Dec'(sk, c)}$

1 : Parse $c = (c_1, c_2)$
2 : $m' := Dec(sk, c_1)$
3 : **if** $m' \notin \mathcal{M} \vee H(m') \neq c_2$
4 : **return** $\perp$
5 : **else return** m'

$\underline{S(pk)}$

1 : $m \xleftarrow{\$} \mathcal{M}$
2 : $c_1 = Enc(pk, m)$
3 : $c_2 \xleftarrow{\$} \{0,1\}^n$
4 : **return** $c = (c_1, c_2)$

Fig. 18: DS-secure $\text{DPKE}' = \text{KC}[\text{DPKE}, H]$ with simulation S

To a public-key encryption DPKE=(Gen, Enc, Dec) with message space $\mathcal{M}$, and a random oracle $H : \mathcal{M} \to \{0,1\}^n$, we associate $\text{DPKE}' = (Gen', Enc', Dec') = \text{KC}[\text{DPKE}, H]$, see Fig. 18.

Theorem 5 (DPKE OW-CPA $\overset{QROM}{\Rightarrow}$ DPKE′ DS). *Let S be the algorithm described in Fig. 18. If DPKE is perfectly correct, $\text{DISJ}_{\text{DPKE}',S} = 2^{-n}$. Moreover, for any $\mathcal{B}$ against DPKE′ issuing at most q_H quantum queries to H, there exists an adversary $\mathcal{A}$ against the OW-CPA security of DPKE such that $\mathtt{Adv}^{\text{DS-IND}}_{\text{DPKE}',U_{\mathcal{M}},S}(\mathcal{B}) \leq 2\sqrt{(q_H+1)\mathtt{Adv}^{\text{OW-CPA}}_{\text{DPKE}}(\mathcal{A})}$ and the running time of $\mathcal{A}$ is about that of $\mathcal{B}$.*

Proof. Since H is a random oracle, it's obvious that $\text{DISJ}_{\text{DPKE}',S} = 2^{-n}$.

Let Ω_H be the sets of all functions $H : \mathcal{M} \to \{0,1\}^n$. Define game G_0 and game G_1 as in Fig. 19. Then, we have

$$\mathtt{Adv}^{\text{DS-IND}}_{\text{DPKE}',U_{\mathcal{M}},S}(\mathcal{B}) = \left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - \Pr[G_1^{\mathcal{B}} \Rightarrow 1]\right|.$$

Let $\ddot{H}$ be the function such that $\ddot{H}(m) = H(m)$ for $m \neq m^*$ and $\ddot{H}(m^*) \xleftarrow{\$} \mathcal{R}$. Then, it's easy to see that G_1 can be rewritten as G_2 in Fig. 20 and we have

$$\Pr[G_1^{\mathcal{B}} \Rightarrow 1] = \Pr[G_2^{\mathcal{B}} \Rightarrow 1]$$

GAME G_3. G_3 is the same as G_2 except that $\ddot{H}$ is replaced by a semi-classical oracle $\ddot{H}\backslash m^*$. For a query input, $H\backslash m^*$ will first query $\mathcal{O}^{SC}_{m^*}$, i.e., perform a semi-classical measurement, and then query $\ddot{H}$. Particularly, $\mathcal{O}^{SC}_{m^*}$ here will be equipped with an auxiliary extractor to record m^*. Let Find be the event that $\mathcal{O}^{SC}_{m^*}$ ever outputs 1 and m^* during semi-classical measurements of the queries to $\ddot{H}\backslash m^*$.

GAMES G_0		GAMES G_1	
1 :	$(pk, sk) \leftarrow Gen'; H \xleftarrow{\$} \Omega_H$	1 :	$(pk, sk) \leftarrow Gen'; H \xleftarrow{\$} \Omega_H$
2 :	$m^* \xleftarrow{\$} \mathcal{M}$	2 :	$m^* \xleftarrow{\$} \mathcal{M}$
3 :	$c_1^* := Enc(pk, m^*)$	3 :	$c_1^* := Enc(pk, m^*)$
4 :	$c_2^* := H(m^*)$	4 :	$c_2^* \xleftarrow{\$} \{0,1\}^n$
5 :	$c^* = (c_1^*, c_2^*)$	5 :	$c^* = (c_1^*, c_2^*)$
6 :	$b' \leftarrow \mathcal{B}^H(pk, c^*)$	6 :	$b' \leftarrow \mathcal{B}^H(pk, c^*)$
7 :	**return** b'	7 :	**return** b'

Fig. 19: Games G_0-G_1 for the proof of Theorem 5

GAMES G_2		GAMES G_3	
1 :	$(pk, sk) \leftarrow Gen'; H \xleftarrow{\$} \Omega_H$	1 :	$(pk, sk) \leftarrow Gen'; H \xleftarrow{\$} \Omega_H$
2 :	$m^* \xleftarrow{\$} \mathcal{M}$	2 :	$m^* \xleftarrow{\$} \mathcal{M}$
3 :	$c_1^* := Enc(pk, m^*)$	3 :	$c_1^* := Enc(pk, m^*)$
4 :	$c_2^* := H(m^*)$	4 :	$c_2^* := H(m^*)$
5 :	$c^* = (c_1^*, c_2^*)$	5 :	$c^* = (c_1^*, c_2^*)$
6 :	$\ddot{H} = H; \ddot{H}(m^*) \xleftarrow{\$} \{0,1\}^n$	6 :	$\ddot{H} = H; \ddot{H}(m^*) \xleftarrow{\$} \{0,1\}^n$
7 :	$b' \leftarrow \mathcal{B}^{\ddot{H}}(pk, c^*)$	7 :	$b' \leftarrow \mathcal{B}^{\ddot{H}\backslash m^*}(pk, c^*)$
8 :	**return** b'	8 :	**return** b'

Fig. 20: Games G_2-G_3 for the proof of Theorem 5

Applying Lemma 5 with $X = \mathcal{M}$, $Y = \mathcal{R}$, $S = \{m^*\}$, $\mathcal{O}_1 = H$, $\mathcal{O}_2 = \ddot{H}$ and $z = (pk, c^*)$, we can have

$$\left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - \Pr[G_2^{\mathcal{B}} \Rightarrow 1]\right| \leq 2\sqrt{(q_G + 1)\Pr[\mathsf{Find} \ : G_3]}.$$

Note that in game G_3, $H(m^*)$ is only used in evaluating c_2^* and independent of $\ddot{H}$, we can replace $c_2^* := H(m^*)$ by $c_2^* \leftarrow \{0,1\}^n$ and then simplify G_3 as G_4 in Fig. 21. Since the output distributions of G_3 and G_4 are totally identical, we

have

$$\Pr[\mathsf{Find} : G_3] = \Pr[\mathsf{Find} : G_4].$$

Next, construct an adversary $\mathcal{A}(pk, c_1^*)$ against the OW-CPA security of underlying DPKE as in Fig. 21, where $\hat{H}\backslash m^*$ is the same as $H\backslash m^*$ except that the indication function $f_{m^*}(m)$ which outputs 1 if $m = m^*$ and 0 otherwise, is replaced by $\hat{f}_{m^*}(m)$,

$$\hat{f}_{m^*}(m) = \begin{cases} 1 & Enc(pk, m) = c^* \\ 0 & \text{otherwise.} \end{cases}$$

Note that $Enc(pk, m) = c^*$ iff $m = m^*$ since PKE is perfectly correct. Thus, $\hat{H}\backslash m^*$ is totally identical with $H\backslash m^*$. According to Lemma 1,

$$\Pr[\mathsf{Find} : G_4] = \mathtt{Adv}^{\text{OW-CPA}}_{\text{DPKE}}(\mathcal{A}).$$

GAMES G_4		$\mathcal{A}(pk, c_1^*)$	
1 :	$(pk, sk) \leftarrow Gen'; H \xleftarrow{\$} \Omega_H$	1 :	$c_2^* \xleftarrow{\$} \{0,1\}^n$
2 :	$m^* \xleftarrow{\$} \mathcal{M}$	2 :	$c^* = (c_1^*, c_2^*)$
3 :	$c_1^* := Enc(pk, m^*)$	3 :	Pick a $2q_H$-wise function H
4 :	$c_2^* \xleftarrow{\$} \{0,1\}^n$	4 :	$b' \leftarrow \mathcal{B}^{\hat{H}\backslash m^*}(pk, c^*)$
5 :	$c^* = (c_1^*, c_2^*)$	5 :	**return** Find
6 :	$b' \leftarrow \mathcal{B}^{H\backslash m^*}(pk, c^*)$		
7 :	**return** b'		

Fig. 21: Game G_4 and $\mathcal{A}$ for the proof of Theorem 5

Collecting all above bounds, we have

$$\mathtt{Adv}^{\text{DS-IND}}_{\text{DPKE}',U_{\mathcal{M}},S}(\mathcal{B}) \leq 2\sqrt{(q_H+1)\mathtt{Adv}^{\text{OW-CPA}}_{\text{DPKE}}(\mathcal{A})}.$$

□

Gen		$Encaps(pk)$		$Decaps(sk', c)$	
1 :	$(pk, sk) \leftarrow Gen'$	1 :	$m \xleftarrow{\$} \mathcal{M}$	1 :	Parse $sk' = (sk, s)$
2 :	$s \xleftarrow{\$} \mathcal{M}$	2 :	$c \leftarrow Enc'(pk, m)$	2 :	$m' := Dec'(sk, c)$
3 :	$sk' := (sk, s)$	3 :	$K := H(m, c)$	3 :	**if** $m' = \perp$
4 :	**return** (pk, sk')	4 :	**return** (K, c)	4 :	**return** $K := H(s, c)$
				5 :	**else return**
				6 :	$K := H(m', c)$

Fig. 22: IND-CCA-secure KEM-III = $U^{\not\perp}[\text{PKE}', H]$

D.4 $U^{\not\perp}$: from OW-qPCA-secure PKE to IND-CCA-secure KEM

To a public-key encryption PKE′=(Gen', Enc', Dec') and a random oracle H ($H : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$), we associate KEM-III = $U^{\not\perp}[\mathrm{PKE}', H]$. The algorithms of KEM-III=(Gen,$Encaps$,$Decaps$) are defined in Fig. 22.

Theorem 6 (PKE′ OW-qPCA $\overset{QROM}{\Rightarrow}$ KEM IND-CCA). *If PKE′ is δ-*correct*, for any IND-CCA $\mathcal{B}$ against KEM-III, issuing at most q_D (classical) queries to the decapsulation oracle* Decaps *and at most q_H queries to the quantum random oracle H, there exists a OW-qPCA adversary $\mathcal{A}$ against PKE′ such that* $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-III}}(\mathcal{B}) \leq 2q_H\frac{1}{\sqrt{|\mathcal{M}|}} + 2\sqrt{(q_H+1)(2\delta + \mathtt{Adv}^{\text{OW}-q\text{PCA}}_{\text{PKE}'}(\mathcal{A}))}$ *and the running time of $\mathcal{A}$ is about that of $\mathcal{B}$.*

Proof. Let $\mathcal{B}$ be an adversary against the IND-CCA security of KEM-III, issuing at most q_D queries to Decaps and at most q_H queries to H. Let Ω_H and Ω_{H_q} be the sets of all functions $H : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ and $H_q : \mathcal{C} \to \mathcal{K}$, respectively. Consider the games in Fig. 23 and Fig. 25.

Game G_0. Since game G_0 is exactly the IND-CCA game,

$$\left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - 1/2\right| = \mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-III}}(\mathcal{B}).$$

Game G_1. In game G_1, the Decaps oracle is changed that $H'_q(c)$ is returned instead of $H(s, c)$ for an invalid encapsulation c. Considering that $\mathcal{B}$'s view is independent of (the uniform secret) s, we can use Lemma 6 to obtain

$$\left|\Pr[G_0^{\mathcal{B}} \Rightarrow 1] - \Pr[G_1^{\mathcal{B}} \Rightarrow 1]\right| \leq 2q_H \cdot \frac{1}{\sqrt{\mathcal{M}}}.$$

Game G_2. In game G_2, H is changed that $H_q(c)$ is returned instead of $H_1(m, c)$ when $(m,\ c)$ satisfies $\textsc{Pco}(m, c) = 1$ (i.e., $Dec'(sk, c) = m$). Note that it is impossible that $\textsc{Pco}(m_1, c) = \textsc{Pco}(m_2, c) = 1$ for $m_1 \neq m_2$ because Dec' is a deterministic algorithm. Further, as H_q is a random function independent of H_1, H in game G_2 is also a uniformly random function like the one in game G_1. Thus,

$$\Pr[G_1^{\mathcal{B}} \Rightarrow 1] = \Pr[G_2^{\mathcal{B}} \Rightarrow 1].$$

Game G_3. In game G_3, the Decaps oracle is changed that it makes no use of the secret key sk' any more. When $\mathcal{B}$ queries the Decaps oracle on c ($c \neq c^*$), $K := H_q(c)$ is returned as the response. In order to show that the output distributions of Decaps are identical in G_2 and G_3, we consider the following cases for a fixed ciphertext c and $m' := Dec'(sk, c)$.

Case 1: $m' \neq \perp$. Note that $H(m', c) = H_q(c)$ on account of $\textsc{Pco}(m', c) = 1$. Therefore, the two Decaps oracles in games G_2 and G_3 return the same value.

Case 2: $m' = \perp$. Random values $H'_q(c)$ and $H_q(c)$ in $\mathcal{K}$ are returned in G_2 and G_3, respectively. In G_2, H'_q is a random function independent of H. In G_3, $\mathcal{B}$'s queries to H can only help him get access to H_q at c such that $Dec'(sk, c) = \hat{m}$ for some $\hat{m} \neq \perp$. Therefore, $\mathcal{B}$ never sees $H_q(c)$ by querying H. Hence, in $\mathcal{B}$'s view, $H_q(c)$ is totally uniform at random like $H'_q(c)$. As a result, the DECAPS oracle in G_3 has the same output distribution as the one in G_2.

Thus, $\mathcal{B}$'s views are identical in G_2 and G_3 and we have

$$\Pr[G_2^{\mathcal{B}} \Rightarrow 1] = \Pr[G_3^{\mathcal{B}} \Rightarrow 1].$$

GAMES $G_0 - G_5$

1 : $(pk, sk') \leftarrow Gen'$
2 : $H_q, H'_q \xleftarrow{\$} \Omega_{H_q}; H_1 \xleftarrow{\$} \Omega_H$
3 : $m^* \xleftarrow{\$} \mathcal{M}$
4 : $r^* \xleftarrow{\$} \mathcal{R}$
5 : $c^* := Enc(pk, m^*; r^*)$
6 : $k_0^* := H(m^*, c^*)$
7 : $k_1^* \xleftarrow{\$} \mathcal{K}$
8 : $b \xleftarrow{\$} \{0, 1\}$
9 : $b' \leftarrow B^{H, \text{DECAPS}}(pk, c^*, k_b^*) // G_0 - G_3$
10 : $\ddot{H} = H; \ddot{H}(m^*, c^*) \xleftarrow{\$} \mathcal{R}$ $// G_4 - G_5$
11 : $b' \leftarrow B^{\ddot{H}, \text{DECAPS}}(pk, c^*, k_b^*) // G_4$
12 : $b' \leftarrow B^{\ddot{H} \backslash (m^*, c^*), \text{DECAPS}}(pk, c^*, k_b^*) // G_5$
13 : **return** $b' =? b$

$H(m, c)$

1 : **if** $\text{PCO}(m, c) = 1$ $// G_2 - G_5$
2 : **return** $H_q(c)$ $// G_2 - G_5$
3 : **return** $H_1(m, c)$

DECAPS $(c \neq c^*)$ $// G_0 - G_2$

1 : Parse $sk' = (sk, s)$
2 : $m' := Dec'(sk, c)$
3 : **if** $m' \neq \perp$ **return** $K := H(m', c)$
4 : **else return**
5 : $K := H(s, c)$ $// G_0$
6 : $K := H'_q(c)$ $// G_1 - G_2$

DECAPS $(c \neq c^*)$ $// G_3 - G_5$

1 : **return** $K := H_q(c)$

Fig. 23: Games G_0-G_5 for the proof of Theorem 6

Let $\ddot{H}$ be an oracle such that $\ddot{H}(m, c) = H(m, c)$ for $(m, c) \neq (m^*, c^*)$ and $\ddot{H}(m^*, c^*) \xleftarrow{\$} \mathcal{K}$. Let S be a singleton $\{(m^*, c^*)\}$.

GAME G_4. In game G_4, replace H by $\ddot{H}$. We note that in this game b is independent of $\mathcal{B}$'s view since queries to $\ddot{H}$ and DECAPS can not reveal $H(m^*, c^*)$. Thus,

$$\Pr[G_4^{\mathcal{B}} \Rightarrow 1] = 1/2.$$

GAME G_5. In game G_5, replace $\ddot{H}$ by $\ddot{H} \backslash S$. Given any query input, $\ddot{H} \backslash S$ will first query semi-classical oracle $\mathcal{O}_S^{SC}$, i.e., perform a semi-classical measurement,

and then query $\ddot{H}$. Particularly, $\mathcal{O}_S^{SC}$ here will be equipped with an auxiliary extractor to record (m^*, c^*). Let Find be the event that $\mathcal{O}_S^{SC}$ ever outputs 1 and (m^*, c^*) during semi-classical measurements of the queries to $\ddot{H}\backslash S$.

Let A^H be an oracle algorithm on input $(pk,\, c^*,\, k_0^*,\, H_q)$[10] in Fig. 24. Sample pk, c^*, k_0^*, H_q and H in the same way as G_3 and G_4, i.e., $(pk, sk) \leftarrow Gen$, $m^* \xleftarrow{\$} \mathcal{M}$, $r^* \xleftarrow{\$} \mathcal{R}$, $c^* = Enc(pk, m^*; r^*)$, $H_q \xleftarrow{\$} \Omega_{H_q}$, $H_1 \xleftarrow{\$} \Omega_H$, simulate H in the same way as G_3 and G_4, set $k_0^* = H(m^*, c^*)$, $\ddot{H}(m, c) = H(m, c)$ for $(m, c) \neq (m^*, c^*)$ and $\ddot{H}(m^*, c^*) \xleftarrow{\$} \mathcal{K}$. Then, A^H and $A^{\ddot{H}}$ on input $(pk,\, c^*,\, k_0^*,\, H_q)$ perfectly simulate G_3 and G_4, respectively.

$A^H(pk, c^*, k_0^*, H_q)$	DECAPS $(c \neq c^*)$
$k_1^* \xleftarrow{\$} \mathcal{K}; b \xleftarrow{\$} \{0,1\}$	**return** $K := H_q(c)$
$b' \leftarrow \mathcal{B}^{H,\text{DECAPS}}(pk, c^*, k_b^*)$	
return $b' =? b$	

Fig. 24: A^H for the proof of Theorem 6.

GAMES $G_6 - G_9$	$H(m, c)$
$(pk, sk') \leftarrow Gen'$	**if** PCO$(m, c) = 1$
$H_q \xleftarrow{\$} \Omega_{H_q}; H_1 \xleftarrow{\$} \Omega_H$	**return** $H_q(c)$
$m^* \xleftarrow{\$} \mathcal{M}; r^* \xleftarrow{\$} \mathcal{R}$	**else return** $H_1(m, c)$
$r^* \xleftarrow{\$} \mathcal{R}_{good}(pk, sk, m) // G_7 - G_8$	DECAPS $(c \neq c^*)$
$c^* := Enc(pk, m^*; r^*)$	**return** $K := H_q(c)$
$k_0^*, k_1^* \xleftarrow{\$} \mathcal{K}; b \xleftarrow{\$} \{0,1\}$	
$b' \leftarrow B^{H\backslash(m^*,c^*),\text{DECAPS}}(pk, c^*, k_b^*) // G_6 - G_7$	
$b' \leftarrow B^{\hat{H}\backslash(m^*,c^*),\text{DECAPS}}(pk, c^*, k_b^*) // G_8 - G_9$	
return $b' =? b$	

Fig. 25: Games $G_6 - G_9$ for the proof of Theorem 6

Applying Lemma 5 with $X = (\mathcal{M}, \mathcal{C})$, $Y = \mathcal{K}$, $S = \{(m^*, c^*)\}$, $\mathcal{O}_1 = H$, $\mathcal{O}_2 = \ddot{H}$ and $z = (pk, c^*, k_0^*, H_q)$, we can have

$$\left|\Pr[G_3^{\mathcal{B}} \Rightarrow 1] - \Pr[G_4^{\mathcal{B}} \Rightarrow 1]\right| \leq 2\sqrt{(q_H + 1)\Pr[\mathsf{Find} : G_5]}.$$

Since $H(m^*, c^*)$ in G_5 is used only once and independent of the oracles $\ddot{H}$ and DECAPS, we can replace $k_0^* = H(m^*, c^*)$ by $k_0^* \xleftarrow{\$} \mathcal{K}$ and simplify G_5 into G_6 as

[10] As in the proof of Theorem 1, H_q here can be either the whole truth table of H_q or an accessible oracle.

in Fig. 25, and have

$$\Pr[\mathsf{Find} \ : G_5] = \Pr[\mathsf{Find} \ : G_6]$$

GAME G_7. In game G_7, replace $r^* \xleftarrow{\$} \mathcal{R}$ by $r^* \xleftarrow{\$} \mathcal{R}_{good}(pk, sk, m^*)$, where $\mathcal{R}_{\text{good}}(pk, sk, m^*) := \{r \in \mathcal{R} : Dec(sk, Enc(pk, m^*; r)) = m^*\}$.

We note that the statistical distance between uniform distributions on $\mathcal{R}$ and $\mathcal{R}_{\text{good}}(pk, sk, m^*)$ is

$$\delta(pk, sk, m^*) = \frac{|\mathcal{R}_{\text{bad}}(pk, sk, m^*)|}{|\mathcal{R}|},$$

where $\mathcal{R}_{\text{bad}}(pk, sk, m^*) = \mathcal{R} \backslash \mathcal{R}_{\text{good}}(pk, sk, m^*)$. Let $\delta(pk, sk) = \max_{m \in \mathcal{M}} \delta(pk, sk, m)$. Then $\delta = \mathbf{E}[\delta(pk, sk)]$, where the expectation is taken over $(pk, sk) \leftarrow Gen$.

Conditioned on a fixed (pk, sk) we have $|\Pr[\mathsf{Find} \ : G_6] - \Pr[\mathsf{Find} \ : G_7]| \leq \delta(pk, sk)$. By averaging over $(pk, sk) \leftarrow Gen$ we finally obtain

$$|\Pr[\mathsf{Find} \ : G_6] - \Pr[\mathsf{Find} \ : G_7]| \leq \delta.$$

GAME G_8. In game G_8, replace $H \backslash S$ by $\hat{H} \backslash S$, where $\hat{H} \backslash S$ is the same as $H \backslash S$ except the indicator function $f_S(m, c)$ in semi-classical oracle $\mathcal{O}_S^{SC}$ is replaced by $\hat{f}_S(m, c)$. $\hat{f}_S(m, c)$ is defined as

$$\hat{f}_S(m, c) = \begin{cases} 1 & \text{PCO}(m, c^*) = 1 \text{ and } c = c^* \\ 0 & \text{otherwise.} \end{cases}$$

We note that $f_S(m, c) = 1$ iff $(m, c) = (m^*, c^*)$. Since $c^* = Enc(pk, m^*; r^*)$ and the r^* in this game is sampled from "good" randomness, $\text{PCO}(m, c^*) = 1$ iff $m = m^*$. Therefore, $f_S(m, c)$ is identical with $\hat{f}_S(m, c)$, and we have

$$\Pr[\mathsf{Find} \ : G_7] = \Pr[\mathsf{Find} \ : G_8].$$

$\mathcal{A}(1^\lambda, pk, c^*)$

1 : $k_0^*, k_1^* \xleftarrow{\$} \mathcal{K}$
2 : $b \xleftarrow{\$} \{0, 1\}$
3 : Pick a q_H-wise functions H_q, H_1
4 : $m' \leftarrow \mathcal{B}^{\hat{H} \backslash S, \text{DECAPS}}(pk, c^*)$
5 : **return** Find

$H(m, c)$

1 : **if** $\text{PCO}(m, c) = 1$
2 : **return** $H_q(c)$
3 : **else return** $H_1(m, c)$

DECAPS $(c \neq c^*)$

1 : **return** $K := H_q(c)$

Fig. 26: Adversary $\mathcal{A}$ for the proof of Theorem 6

GAME G_9. In game G_9, switch $r^* \xleftarrow{\$} \mathcal{R}_{good}(pk, sk, m^*)$ back to $r^* \xleftarrow{\$} \mathcal{R}$. Similar to the case of bounding the difference between G_6 and G_7, we can have

$$|\Pr[\mathsf{Find} : G_8] - \Pr[\mathsf{Find} : G_9]| \leq \delta.$$

Then, we construct an adversary $\mathcal{A}$ against the OW-qPCA security of the PKE′ scheme as in Fig. 26. According to Lemma 1,

$$\mathtt{Adv}^{\mathrm{OW}-q\mathrm{PCA}}_{\mathrm{PKE}'}(\mathcal{A}) = \Pr[\mathsf{Find} : G_9].$$

Finally, combing this with the bounds derived above, we can conclude that

$$\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-III}}(\mathcal{B}) \leq 2q_H \frac{1}{\sqrt{\mathcal{M}}} + 2\sqrt{(q_H+1)(2\delta + \mathtt{Adv}^{\mathrm{OW}-q\mathrm{PCA}}_{\mathrm{PKE}'}(\mathcal{A}))}.$$

□

D.5 U$^{\perp}$: from OW-qPVCA-secure PKE to IND-CCA-secure KEM

U$^{\perp}$, a KEM variant of the REACT/GEM transformations [8, 9], was first given by [3, Table 2]. To a public-key encryption PKE′=(Gen', Enc', Dec') and a hash function H, we associate KEM-IV = U$^{\perp}$[PKE′, H] as in Fig. 27. The security of U$^{\perp}$ in the QROM was analyzed by [11]. However, the security proof is non-tight due to utilization of OW2H lemma. Here, using the semi-classical OW2H lemma in the same way as in Theorem 6, we improve the tightness and obtain the following theorem.

Gen	$Encaps(pk)$	$Decaps^{\perp}(sk, c)$
1 : $(pk, sk) \leftarrow Gen'$	1 : $m \xleftarrow{\$} \mathcal{M}$	1 : $m' := Dec'(sk, c)$
2 : **return** (pk, sk)	2 : $c \leftarrow Enc'(pk, m)$	2 : **if** $m' = \perp$
	3 : $K := H(m, c)$	3 : **return** $\perp$
	4 : **return** (K, c)	4 : **else return**
		5 : $K := H(m', c)$

Fig. 27: IND-CCA-secure KEM-IV = $U^{\perp}$[PKE′, H]

Theorem 7 (PKE′ OW-qPVCA $\overset{QROM}{\Rightarrow}$ KEM-IV IND-CCA). *If* PKE′ *is* δ-correct*, for any IND-CCA* $\mathcal{B}$ *against KEM-IV, issuing at most* q_D *(classical) queries to the decapsulation oracle* DECAPS *and at most* q_H *queries to the quantum random oracle* H*, there exists a OW-qPVCA adversary* $\mathcal{A}$ *against PKE′ that makes at most* q_H *queries to the* PCO *oracle and at most* q_D *(classical) queries to the* VAL *oracle such that* $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-IV}}(\mathcal{B}) \leq 2\sqrt{(q_H+1)(2\delta + \mathtt{Adv}^{\mathrm{OW}-q\mathrm{PVCA}}_{\mathrm{PKE}'}(\mathcal{A}))}$ *and the running time of* $\mathcal{A}$ *is about that of* $\mathcal{B}$*.*

D.6 $\mathrm{U}^{\not\perp}_{m}/U^{\perp}_{m}$: from OW-CPA-secure/OW-VA-secure DPKE to IND-CCA KEM

The transformation $\mathrm{U}^{\not\perp}_{m}$ ($\mathrm{U}^{\perp}_{m}$) [27, 4] is a variant of $\mathrm{U}^{\not\perp}$ ($\mathrm{U}^{\perp}$) that derives the KEM key as $K = H(m)$ instead of $K = H(m, c)$. To a deterministic public-key encryption scheme $\mathrm{PKE}' = (Gen', Enc', Dec')$ with message space $\mathcal{M}$, a hash function $H : \mathcal{M} \to \mathcal{K}$, and a pseudorandom function f with key space $\mathcal{K}^{prf}$, we associate KEM-V=$\mathrm{U}^{\not\perp}_{m}$[DPKE′,$H$,$f$] and KEM-VI=$\mathrm{U}^{\perp}_{m}$[DPKE′,$H$] shown in Fig. 28 and Fig. 29, respectively.

Gen		$Encaps(pk)$		$Decaps(sk', c)$	
1 :	$(pk, sk) \leftarrow Gen'$	1 :	$m \xleftarrow{\$} \mathcal{M}$	1 :	Parse $sk' = (sk, k)$
2 :	$k \xleftarrow{\$} \mathcal{K}^{prf}$	2 :	$c := Enc'(pk, m)$	2 :	$m' := Dec'(sk, c)$
3 :	$sk' := (sk, k)$	3 :	$K := H(m)$	3 :	**if** $Enc'(pk, m') = c$
4 :	**return** (pk, sk')	4 :	**return** (K, c)	4 :	**return** $K := H(m')$
				5 :	**else return**
				6 :	$K := f(k, c)$

Fig. 28: IND-CCA-secure KEM-V=$\mathrm{U}^{\not\perp}_{m}$[DPKE′,$H$,$f$]

Gen		$Encaps(pk)$		$Decaps(sk, c)$	
1 :	$(pk, sk) \leftarrow Gen'$	1 :	$m \xleftarrow{\$} \mathcal{M}$	1 :	$m' := Dec'(sk, c)$
2 :	**return** (pk, sk)	2 :	$c := Enc'(pk, m)$	2 :	**if** $Enc'(pk, m') = c$
		3 :	$K := H(m)$	3 :	**return** $K := H(m')$
		4 :	**return** (K, c)	4 :	**else return** $\perp$

Fig. 29: IND-CCA-secure KEM-VI=$\mathrm{U}^{\perp}_{m}$[DPKE′,H]

We note that for a deterministic PKE scheme the OW-PCA security is equivalent to the OW-CPA security as we can simulate the Pco oracle via re-encryption during the proof. Thus, combing the proofs of Theorem 6, Theorem 7, we can easily obtain the following two theorems.

Theorem 8 (PKE′ OW-CPA $\overset{QROM}{\Rightarrow}$ KEM-V IND-CCA). *If* PKE′ *is* δ-correct *and deterministic, for any IND-CCA* $\mathcal{B}$ *against KEM-V, issuing at most* q_D *(classical) queries to the decapsulation oracle* Decaps *and at most* q_H *quantum queries to the random oracle* H*, there exist a OW-CPA adversary* $\mathcal{A}$ *against* PKE′ *and an adversary* $\mathcal{A}'$ *against the security of* PRF *with at most* q_D *classical queries such that* $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-V}}(\mathcal{B}) \leq 2\sqrt{(q_H+1)(\mathtt{Adv}^{\text{OW-CPA}}_{\text{PKE}'}(\mathcal{A}) + \delta)} + \delta + \mathtt{Adv}_{\text{PRF}}(\mathcal{A}')$*, and the running time of* $\mathcal{A}$ *is about that of* $\mathcal{B}$*.*

Theorem 9 (PKE′ OW-VA $\overset{QROM}{\Rightarrow}$ KEM-VI IND-CCA). *If* PKE′ *is* δ-correct *and deterministic, for any IND-CCA* $\mathcal{B}$ *against KEM-VI, issuing at most* q_D *(classical) queries to the decapsulation oracle* DECAPS *and at most* q_H *quantum queries to the random oracle* H*, there exists a OW-VA adversary* $\mathcal{A}$ *against* PKE′ *who makes at most* q_D *(classical) queries to the* VAL *oracle such that* $\mathtt{Adv}^{\text{IND-CCA}}_{\text{KEM-VI}}(\mathcal{B}) \leq 2\sqrt{(q_H+1)(\mathtt{Adv}^{\text{OW-VA}}_{\text{PKE}'}(\mathcal{A})+\delta)}+\delta$ *and the running time of* $\mathcal{A}$ *is about that of* $\mathcal{B}$*.*