Measure-Rewind-Extract: Tighter Proofs of One-Way to Hiding and CCA Security in the Quantum Random Oracle Model

Jiangxia Ge[1,2], Heming Liao[1,2], and Rui Xue[1,2](✉)

[1] Key Laboratory of Cyberspace Security Defense, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100084, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
`{gejiangxia, liaoheming, xuerui}@iie.ac.cn`

Abstract. The One-Way to Hiding (O2H) theorem, first given by Unruh (J ACM 2015) and then restated by Ambainis et al. (CRYPTO 2019), is a crucial technique for solving the reprogramming problem in the quantum random oracle model (QROM). It provides an upper bound $d\cdot\sqrt{\epsilon}$ for the distinguisher's advantage, where d is the query depth and ϵ denotes the advantage of a one-wayness attacker. Later, in order to obtain a tighter upper bound, Kuchta et al. (EUROCRYPT 2020) proposed the Measure-Rewind-Measure (MRM) technique and then proved the Measure-Rewind-Measure O2H (MRM-O2H) theorem, which provides the upper bound $d\cdot\epsilon$. They also proposed an open question: Can we combine their MRM technique with Ambainis et al.'s semi-classical oracle technique (CRYPTO 2019) or Zhandry's compressed oracle technique (CRYPTO 2019) to prove a new O2H theorem with an upper bound even tighter than $d\cdot\epsilon$?

In this paper, we give an affirmative answer for the above question. We propose a new technique named Measure-Rewind-Extract (MRE) by combining the MRM technique with the semi-classical oracle technique. By using MRE technique, we prove the Measure-Rewind-Extract O2H (MRE-O2H) theorem, which provides the upper bound $\sqrt{d}\cdot\epsilon$.

As an important application of our MRE-O2H theorem, for the $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}^{\not\perp}_m$, $\mathsf{FO}^{\perp}$ and $\mathsf{FO}^{\perp}_m$ proposed by Hofheinz et al. (TCC 2017), i.e., the key encapsulation mechanism (KEM) variants of the Fujisaki-Okamoto transformation, we prove the following results in the QROM:

- Their IND-CCA security can be reduced to the IND-CPA security of the underlying public key encryption (PKE) scheme without the square-root advantage loss. In particular, compared with the IND-CCA proof of $\mathsf{FO}^{\not\perp}$ given by Kuchta et al. (EUROCRYPT 2020), ours removes the injectivity assumption and has a tighter security bound.
- Under the assumption that the underlying PKE scheme is unique randomness recoverable, we for the first time prove that their IND-CCA security can be reduced to the OW-CPA security of the underlying PKE scheme without the square-root advantage loss.

Keywords: quantum random oracle model · security proof · Fujisaki-Okamoto transformation · key encapsulation mechanism.

1 Introduction

The Fujisaki-Okamoto (FO) transformation [13] is used to construct a public key encryption (PKE) scheme that is secure against the indistinguishability under chosen-ciphertext attacks ($\mathsf{IND\text{-}CCA}$) in the random oracle model (ROM) [3]. The PKE scheme constructed by FO is based on a weakly secure PKE scheme, which can only be secure against the indistinguishability under chosen-plaintext attacks ($\mathsf{IND\text{-}CPA}$) or the one-wayness under one-way attacks ($\mathsf{OW\text{-}CPA}$). Compared to directly constructing an $\mathsf{IND\text{-}CCA}$-secure PKE scheme, it is considered easier and more efficient to first construct an $\mathsf{IND\text{-}CCA}$-secure key encapsulation mechanism (KEM) scheme and then derive an $\mathsf{IND\text{-}CCA}$-secure PKE scheme via the KEM-DEM paradigm [7]. Following this fact, Dent [9] designed the first KEM variant of FO, which can be used to construct $\mathsf{IND\text{-}CCA}$-secure KEM schemes in the ROM. Further, Hofheinz et al. [15] designed some KEM variants of FO including $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}^{\not\perp}_m$, $\mathsf{FO}^{\perp}$ and $\mathsf{FO}^{\perp}_m$. They proved that the KEM schemes constructed by these variants are $\mathsf{IND\text{-}CCA}$-secure in the ROM. Indeed, these variants are also called the FO-like transformations, the $\not\perp$ (resp. $\perp$) indicates that the variant is implicit (resp. explicit) rejection type, in which a pseudorandom value (resp. an abort symbol $\perp$) is returned if the ciphertext fails to decapsulate.

The FO-like transformations are widely adopted in the NIST post-quantum cryptography standardisation process [29], and hence their post-quantum security has received much attention. As argued by Boneh et al. [5], to fully assess post-quantum security, the ROM should be lifted to the quantum random oracle model (QROM). This means that having only ROM security proof of FO-like transformations is not enough, and we also need QROM security proof.

Up to now, a long sequence of works [30,19,20,4,25,16] have provided the QROM security proofs of FO-like transformations, they all focused on the widely accepted $\mathsf{IND\text{-}CCA}$ security and gave different security bounds. Simultaneously, all those works used the original One-Way to Hiding (O2H) theorem [31,1] (or its variant) to solve the reprogramming problem in the QROM. Here the reprogramming problem can be described informally as follows.

- **The reprogramming problem**. To reprogram a random function $G : X \to Y$ at a subset $S \subseteq X$ is to replace G with a new function H, where $H(x)$ is resampled on $x \in S$ and $H(x) = G(x)$ on $x \notin S$, i.e., G and H only differ on S. The reprogramming problem is: for any distinguisher $\mathcal{A}$ making parallel queries with depth d[3], bound its distinguishing advantage

$$\mathrm{Adv}(\mathcal{A}) := |\Pr[b = 1 : b \leftarrow \mathcal{A}^G] - \Pr[b = 1 : b \leftarrow \mathcal{A}^H]|. \tag{1}$$

This problem is said to be in the QROM if $\mathcal{A}$ has quantum access to its oracle.

The original O2H theorem [31,1] designs a one-wayness attacker $\mathcal{B}_{\mathrm{ow}}$, which has oracle access to H and generates its output x by measuring $\mathcal{A}$'s oracle query. And $\mathcal{B}_{\mathrm{ow}}$'s one-wayness advantage $\mathrm{Adv}(\mathcal{B}_{\mathrm{ow}}) := \Pr[x \in S : x \leftarrow \mathcal{B}^H_{\mathrm{ow}}]$ satisfies $\mathrm{Adv}(\mathcal{A}) \le 2d \cdot \sqrt{\mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})}$, where d is the query depth of $\mathcal{A}$.

[3] See Supplementary Material A.1 for the introduction of parallel query.

From the proof strategies of the long sequence of works [30,19,20,4,25,16], one can find that the upper bound of $\mathrm{Adv}(\mathcal{A})$ influences the tightness of their IND-CCA security proofs. Roughly speaking, the tighter the upper bound of $\mathrm{Adv}(\mathcal{A})$, the tighter their IND-CCA security proofs. Since a tighter security proof means more freedom in the parameter selection, many tighter O2H variants have been proved and used in those long sequence of works, and three representative variants are shown in Table 1. As we can see, these variants are all proved by using some novel techniques and giving more "power" (i.e. oracle 1_S or G) to the one-wayness attacker $\mathcal{B}_{\mathrm{ow}}$, and their upper bounds of $\mathrm{Adv}(\mathcal{A})$ are all indeed tighter than the $2d \cdot \sqrt{\mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})}$ proved by the original O2H theorem [31,1].

Table 1. Three O2H variants. Here $\mathcal{A}$ makes parallel queries to its oracle with query depth d. The $|S|$ denotes the number of elements in set S. The 1_S denotes the indicator function of set S, i.e., $1_S(x) = 1$ if $x \in S$ and 0 otherwise.

O2H theorem	Proved by	$\lvert S\rvert$	$\mathrm{Adv}(\mathcal{A}) \le$	$\mathcal{B}_{\mathrm{ow}}$'s oracle
Original O2H [31,1]	\	Arbitrary	$2d \cdot \sqrt{\mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})}$	H
SC-O2H [1]	semi-classical oracle technique [1]	Arbitrary	$2\sqrt{d \cdot \mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})}$	H and 1_S[a]
DS-O2H [4]	compressed oracle technique [34]	One	$2\sqrt{\mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})}$	H and G
MRM-O2H[b] [25]	Measure-Rewind-Measure (MRM) technique [25]	Arbitrary	$4d \cdot \mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})$	H and G

[a] The SC-O2H theorem actually requires that $\mathcal{B}_{\mathrm{ow}}$ has oracle access to $H\backslash S$. Since $H\backslash S$ can be implemented knowing H and 1_S, we just write H and 1_S here for simplicity.

[b] The MRM-O2H theorem additionally requires that the event used by $\mathcal{A}$ to distinguish G and H is efficiently checkable by itself. In fact, as shown in Eq. (1), the distinguisher $\mathcal{A}$ considered in our paper uses the event $b = 1$ to distinguish G and H, which must be efficiently checkable by $\mathcal{A}$. So, we omit this requirement in this table for simplicity.

Although the three O2H variants shown in Table 1 all have tighter upper bounds, there are extra restrictions during their application, respectively. In more detail, the SC-O2H theorem needs the $\mathcal{B}_{\mathrm{ow}}$ to have oracle access to both H and 1_S, which means that when we try to use $\mathcal{B}_{\mathrm{ow}}$ to attack the underlying hard problem, we need to find a way to simulate the additional 1_S for $\mathcal{B}_{\mathrm{ow}}$. In order to achieve this, it seems that we need to clearly know the set S or at least some values related to S[4]. For the DS-O2H and MRM-O2H theorem, the situation is even worse, as their $\mathcal{B}_{\mathrm{ow}}$ even requires oracle access to both H and G. Indeed, since G and H only differ on the set S, requiring oracle access to both H and G seems stronger than to H and 1_S, in the way that one can determine whether $x \in S$ by testing if $G(x) = H(x)$.

[4] E.g. $S = \{w\}$ and we can get $f(w)$, where f is a public one-way injective function. At this point, we can compute $1_S(x)$ as: $1_S(x) = 1$ if $f(x) = f(w)$ and 0 otherwise.

Fortunately, these restrictions of $\mathcal{B}_{\mathrm{ow}}$ are not completely unattainable, at least they can be achieved when proving the IND-CCA security of FO-like transformations in the QROM. Roughly speaking, in the security proof, due to the special properties of the underlying PKE scheme and the structure of FO-like transformations, one can successfully simulate (H and 1_S)/(H and G) for $\mathcal{B}_{\mathrm{ow}}$. In fact, the (QROM) IND-CCA security proof of $\mathsf{FO}^{\not\perp}$ provided by Kuchta et al. [25] is done in that way: firstly use the MRM-O2H theorem to obtain the corresponding $\mathcal{B}_{\mathrm{ow}}$, then simulate H and G for $\mathcal{B}_{\mathrm{ow}}$, and finally use $\mathcal{B}_{\mathrm{ow}}$ to attack the OW-CPA security of the underlying PKE scheme. One thing we would like to stress is that, since the upper bound provided by the MRM-O2H theorem avoids the square-root advantage loss (see Table 1), Kuchta et al.'s security proof also avoids the square-root advantage loss.

In short, after the long sequence of works [30,19,20,4,25,16], a tighter O2H theorem seems necessary if we want to give tighter QROM security proofs of the FO-like transformations. However, it is quite challenging to prove a tighter O2H theorem, Kuchta et al. also proposed the following question in [25]:

Can we combine their MRM technique with the semi-classical oracle technique or the compressed oracle technique to prove a new O2H theorem that is tighter than their MRM-O2H theorem? And can we use this new O2H theorem to give tighter IND-CCA security proofs of the FO-like transformations in the QROM?

1.1 Our Contribution

Our answer to the above question is yes. We propose a new technique named

Measure-Rewind-Extract (MRE)

by combining the MRM technique with the semi-classical oracle technique. Then, by using our MRE technique, we prove a new O2H theorem (Theorem 4) named

Measure-Rewind-Extract O2H (MRE-O2H).

It shows that $\mathrm{Adv}(\mathcal{A}) \leq 4\sqrt{d} \cdot \mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})$, where d is $\mathcal{A}$'s query depth and $\mathcal{B}_{\mathrm{ow}}$ has oracle access to H, G and 1_S. Note that this upper bound is tighter than the $4d \cdot \mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})$ proved by the MRM-O2H theorem (see Table 1).

Remark 1. Compared with the MRM-O2H theorem, which only requires that $\mathcal{B}_{\mathrm{ow}}$ has oracle access to H and G, our MRE-O2H theorem additionally requires the oracle access to 1_S. In fact, this additional requirement is not essential, as we can simulate 1_S by querying H and G: $1_S(x) = 1$ if $H(x) \neq G(x)$ and 0 otherwise. Intuitively speaking, this simulation is not problematic because G and H only differ on the set S. Here we point out that our MRE-O2H theorem still remains 1_S because directly providing 1_S would make the proof of this theorem more concise and understandable. For completeness, in Supplementary Material B, we (roughly) show that every previous work using the MRM-O2H theorem also works with our MRE-O2H theorem. Therefore, compared with the MRM-O2H theorem, there seems to be no more restrictions on the applicability of our MRE-O2H theorem.

In addition, by using our MRE-O2H theorem, we give tighter $\mathsf{IND\text{-}CCA}$ security proofs of the FO-like transformations $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{FO}^{\perp}$ and $\mathsf{FO}_m^{\perp}$ in the QROM, and the detailed security bounds are shown in Table 2.

Table 2. Security bounds of FO-like transformations in the QROM. Here q is the total number of queries to random oracles, d is the query depth of random oracles, and q_D is the total number of queries to the decapsulation oracle. The "Assumption" column shows the property that needs to be satisfied by the underlying PKE scheme. δ and ϵ respectively represent the correctness error and the security bound of the underlying PKE scheme. Here we abbreviate unique randomness recoverable as URR for simplicity.

Transformation	Underlying security	Assumption	Achieved security	Security bound
$\mathsf{FO}^{\not\perp}, \mathsf{FO}_m^{\not\perp}$ [21]	IND-CPA	\	IND-CCA	$\sqrt{q \cdot \epsilon} + q \cdot \sqrt{\delta}$
$\mathsf{FO}^{\not\perp}$ [25]	IND-CPA	η-injective	IND-CCA	$d^2 \cdot \epsilon + dq \cdot \delta + q\sqrt{\eta}$
$\mathsf{FO}_m^{\perp}$ [18]	IND-CPA	γ-spread	IND-CCA	$\sqrt{(d + q_D) \cdot \epsilon} + q^2 \cdot \delta + qq_D \cdot \sqrt{2^{-\gamma}}$
$\mathsf{FO}^{\not\perp}, \mathsf{FO}_m^{\not\perp}$ Cor. 1	IND-CPA	\	IND-CCA	$d^{1.5} \cdot \epsilon + q^2 \cdot \delta$
$\mathsf{FO}^{\perp}, \mathsf{FO}_m^{\perp}$ Cor. 2	IND-CPA	γ-spread	IND-CCA	$(d + q_D)^{1.5} \cdot \epsilon + q \cdot \sqrt{\delta} + q_D \cdot \sqrt{2^{-\gamma}}$
$\mathsf{FO}^{\not\perp}, \mathsf{FO}_m^{\not\perp}$ Cor. 1	OW-CPA	URR	IND-CCA	$d^{0.5} \cdot \epsilon + q^2 \cdot \delta$
$\mathsf{FO}^{\perp}, \mathsf{FO}_m^{\perp}$ Cor. 2	OW-CPA	URR	IND-CCA	$(d + q_D)^{0.5} \cdot \epsilon + q \cdot \sqrt{\delta}$

In more detail, our $\mathsf{IND\text{-}CCA}$ security proofs all avoid the square-root advantage loss incurred in [21,18]. For the $\mathsf{FO}^{\not\perp}$, when the underlying security is $\mathsf{IND\text{-}CPA}$, our security proof removes the η-injective assumption used in [25] and achieves a tighter security bound. Moreover, we for the first time prove that the $\mathsf{IND\text{-}CCA}$ security of $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{FO}^{\perp}$ and $\mathsf{FO}_m^{\perp}$ can be reduced to the $\mathsf{OW\text{-}CPA}$ security of the underlying PKE scheme without the square-root advantage loss. At this point, we introduce an additional assumption of unique randomness recoverable. Roughly speaking, for a public key pk, a plaintext m and a ciphertext c, this assumption assumes that there exists an efficient algorithm Rec such that $\mathsf{Rec}(pk, m, c) = r$ and the encryption of m with the randomness r is exactly c.

Remark 2. As shown in Table 2, compared with [21,18], although our bounds avoid the square-root advantage loss, the loss related to the query times still exists. For example, if the underlying security is $\mathsf{IND\text{-}CPA}$ and $d = q$ (i.e. each parallel invoking only makes one query), the security bound of $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$ achieved in [21] is $\sqrt{q \cdot \epsilon}$ while ours is $d^{1.5}\epsilon = q^{1.5}\epsilon$ (Cor. 1). At this point, it seems that determining which bound is tighter depends on the actual query times and the concrete underlying problem. Nevertheless, for the massively parallelized attacks, which have low query depth and are the typical methods to deal with high computation costs in practical cryptanalyses, our bound $d^{1.5}\epsilon$ is nearly tight.

1.2 Technique Overview

In this section, for the sake of clarity and understandability, we explain our technique by the following three steps:

- We first introduce a simple distinguisher $\mathcal{A}$ as an example and define some notations that will be used in later explanations.
- Then, based on $\mathcal{A}$, we give a high-level explanation of our Measure-Rewind-Extract (MRE) technique and how we used it to prove our MRE-O2H theorem.
- Finally, we explain how we use the MRE-O2H theorem to give tighter IND-CCA security proofs of the FO-like transformations in the QROM.

A simple distinguisher with query depth 2 Recall that $G, H : X \to Y$ are random functions such that $G(x) = H(x)$ for all $x \notin S$. For the sake of simplicity, we let $S = \{m^* \in X\}$ in the following analysis. That is, there is only one point m^* where G and H differ.

Consider the following simple distinguisher $\mathcal{A}^O$ $(O \in \{G, H\})$ that is aimed to distinguish whether O is G or H:

$$\mathcal{A}^G : \mathbb{M}_{\mathcal{A}} \circ \mathrm{U}_2\mathrm{O}_G\mathrm{U}_1\mathrm{O}_G|\psi\rangle, \quad \mathcal{A}^H : \mathbb{M}_{\mathcal{A}} \circ \mathrm{U}_2\mathrm{O}_H\mathrm{U}_1\mathrm{O}_H|\psi\rangle.$$

Here $|\psi\rangle$ is the initial state of $\mathcal{A}$, U_1 and U_2 are the unitary operations performed by $\mathcal{A}$ between its oracle queries $\mathrm{O}_G/\mathrm{O}_H$, where $\mathrm{O}_G|x, y\rangle = |x, y \oplus G(x)\rangle$ and $\mathrm{O}_H|x, y\rangle = |x, y \oplus H(x)\rangle$. $\mathbb{M}_{\mathcal{A}} := \{\mathrm{M}_0^{\mathcal{A}}, \mathrm{M}_1^{\mathcal{A}}\}$ is the final projective measurement performed by $\mathcal{A}$, and its measurement result b (0 or 1) is $\mathcal{A}$'s final output. Indeed, $\mathcal{A}^O$ considered here is a unitary quantum oracle algorithm that makes parallel queries to O with query depth 2 and query width 1[5].

Before giving our explanation, we first perform some pretreatment. Define two states

$$|\psi_H\rangle := \mathrm{U}_2\mathrm{O}_H\mathrm{U}_1\mathrm{O}_H|\psi\rangle \text{ and } |\psi_G\rangle := \mathrm{U}_2\mathrm{O}_G\mathrm{U}_1\mathrm{O}_G|\psi\rangle.$$

Then, the distinguishing advantage of $\mathcal{A}$ can be computed as follows.

$$\begin{aligned}
\mathrm{Adv}(\mathcal{A}) &= \left|\Pr[b = 1 : b \leftarrow \mathcal{A}^H] - \Pr[b = 1 : b \leftarrow \mathcal{A}^G]\right| \\
&= \left|\|\mathrm{M}_1^{\mathcal{A}}|\psi_H\rangle\|^2 - \|\mathrm{M}_1^{\mathcal{A}}|\psi_G\rangle\|^2\right| \\
&\leq \left|\left(\mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle - |\psi_G\rangle), \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\right)\right| && \text{(By Lemma 3)} \\
&= \left|\Big(|\psi_H\rangle - |\psi_G\rangle, \left(\mathrm{M}_1^{\mathcal{A}}\right)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\Big)\right| && \begin{pmatrix}\text{Basic property} \\ \text{of inner product}\end{pmatrix} \\
&= \left|\left(|\psi_H\rangle - |\psi_G\rangle, \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\right)\right|. && \begin{pmatrix}\mathrm{M}_1^{\mathcal{A}} \text{ is hermitian} \\ \text{and idempotent}\end{pmatrix}
\end{aligned} \tag{2}$$

Let $\mathrm{M}_{m^*} := |m^*\rangle\langle m^*|$ be a projector on the oracle's input register, and let I denotes the identity operator.

[5] See Supplementary Material A.1 for the introduction of unitary quantum oracle algorithm and parallel query.

High-level explanation of our MRE technique Essentially, in order to compute the upper bound of $\mathrm{Adv}(\mathcal{A})$, our MRE technique performs the following three steps:

- **MRE-Step-1**: In this step, we use the projector $\mathrm{M}_{m^*} = |m^*\rangle\langle m^*|$ to divide the state $|\psi_H\rangle - |\psi_G\rangle$. For the state $|\psi_H\rangle$, we have

$$\begin{aligned}
|\psi_H\rangle &= \mathrm{U}_2\mathrm{O}_H\mathrm{U}_1\mathrm{O}_H|\psi\rangle \\
&= \mathrm{U}_2\mathrm{O}_H(\mathrm{M}_{m^*} + \mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H|\psi\rangle \\
&= \mathrm{U}_2\mathrm{O}_H\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_H|\psi\rangle + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H|\psi\rangle \\
&= \mathrm{U}_2\mathrm{O}_H\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_H|\psi\rangle + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H(\mathrm{M}_{m^*} + \mathrm{I} - \mathrm{M}_{m^*})|\psi\rangle \\
&= \mathrm{U}_2\mathrm{O}_H\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_H|\psi\rangle + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H\mathrm{M}_{m^*}|\psi\rangle \\
&\qquad + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})|\psi\rangle.
\end{aligned}$$

One can see that the main idea of the above partition is to sequentially insert $(\mathrm{M}_{m^*} + \mathrm{I} - \mathrm{M}_{m^*})$ before the query O_H, and then divide the entire state into two parts "$\cdots \mathrm{M}_{m^*} \cdots |\psi\rangle$" and "$\cdots (\mathrm{I} - \mathrm{M}_{m^*}) \cdots |\psi\rangle$" by the distributive law. For the first part, we keep it unchanged, and for the second part, we divide it again by inserting $(\mathrm{M}_{m^*} + \mathrm{I} - \mathrm{M}_{m^*})$ before the query O_H. Similarly, for the state $|\psi_G\rangle$, we have

$$\begin{aligned}
|\psi_G\rangle = \mathrm{U}_2\mathrm{O}_G\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_G|\psi\rangle &+ \mathrm{U}_2\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_G\mathrm{M}_{m^*}|\psi\rangle \\
&+ \mathrm{U}_2\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})|\psi\rangle.
\end{aligned}$$

Since G and H only differ on the set $S = \{m^*\}$, the operation $\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})$ must be identical with the operation $\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})$. Based on this property,

$$\begin{aligned}
|\psi_H\rangle - |\psi_G\rangle &= \mathrm{U}_2\mathrm{O}_H\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_H|\psi\rangle + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H\mathrm{M}_{m^*}|\psi\rangle \\
&\qquad + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})|\psi\rangle \\
&\quad - \mathrm{U}_2\mathrm{O}_G\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_G|\psi\rangle - \mathrm{U}_2\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_G\mathrm{M}_{m^*}|\psi\rangle \\
&\qquad - \mathrm{U}_2\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})|\psi\rangle \\
&= \mathrm{U}_2\mathrm{O}_H\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_H|\psi\rangle - \mathrm{U}_2\mathrm{O}_G\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_G|\psi\rangle \\
&\quad + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_H\mathrm{M}_{m^*}|\psi\rangle - \mathrm{U}_2\mathrm{O}_G(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{O}_G\mathrm{M}_{m^*}|\psi\rangle \\
&= \mathrm{U}_2\left(\mathrm{O}_H\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_H|\psi\rangle - \mathrm{O}_G\mathrm{M}_{m^*}\mathrm{U}_1\mathrm{O}_G|\psi\rangle\right) \\
&\quad + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\left(\mathrm{O}_H\mathrm{M}_{m^*}|\psi\rangle - \mathrm{O}_G\mathrm{M}_{m^*}|\psi\rangle\right) \\
&\overset{(a)}{=} \mathrm{U}_2\mathrm{M}_{m^*}\left(\mathrm{O}_H\mathrm{U}_1\mathrm{O}_H|\psi\rangle - \mathrm{O}_G\mathrm{U}_1\mathrm{O}_G|\psi\rangle\right) \\
&\quad + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{M}_{m^*}\left(\mathrm{O}_H|\psi\rangle - \mathrm{O}_G|\psi\rangle\right).
\end{aligned}$$

Here (a) uses the fact that O_G and O_H commute with M_{m^*}, which is obvious since O_G and O_H do not change the state on the oracle's input register.

- **MRE-Step-2**: Define two states $|\psi_1\rangle := \mathrm{O}_H\mathrm{U}_1\mathrm{O}_H|\psi\rangle - \mathrm{O}_G\mathrm{U}_1\mathrm{O}_G|\psi\rangle$ and $|\psi_0\rangle := \mathrm{O}_H|\psi\rangle - \mathrm{O}_G|\psi\rangle$. One can see that the first step of our MRE technique actually shows that $|\psi_H\rangle - |\psi_G\rangle = \mathrm{U}_2\mathrm{M}_{m^*}|\psi_1\rangle + \mathrm{U}_2\mathrm{O}_H(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{U}_1\mathrm{M}_{m^*}|\psi_0\rangle$. Combine this equation with Eq. (2), we get

$$
\begin{aligned}
\mathrm{Adv}(\mathcal{A}) &= \left| \big(|\psi_H\rangle - |\psi_G\rangle, \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle) \big) \right| \\
&= \left| \begin{aligned} &\big(\mathrm{U}_2 \mathrm{M}_{m^*} |\psi_1\rangle, \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle) \big) \\ &+ \big(\mathrm{U}_2 \mathrm{O}_H (\mathrm{I} - \mathrm{M}_{m^*}) \mathrm{U}_1 \mathrm{M}_{m^*} |\psi_0\rangle, \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle) \big) \end{aligned} \right| \\
&\overset{(a)}{=} \left| \begin{aligned} &\big(\mathrm{M}_{m^*} |\psi_1\rangle, \mathrm{M}_{m^*} (\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle) \big) \\ &+ \big(\mathrm{M}_{m^*} |\psi_0\rangle, \mathrm{M}_{m^*} (\mathrm{U}_1)^\dagger (\mathrm{I} - \mathrm{M}_{m^*}) \mathrm{O}_H (\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle) \big) \end{aligned} \right| .
\end{aligned}
$$

Here (a) follows from the basic property of inner product and the fact that M_{m^*} is hermitian and idempotent.

Then, by applying Lemma 4, which guarantees that $|(|\alpha\rangle, |\beta\rangle) + (|\gamma\rangle, |\delta\rangle)| \leq \sqrt{\||\alpha\rangle\|^2 + \||\gamma\rangle\|^2} \cdot \sqrt{\||\beta\rangle\|^2 + \||\delta\rangle\|^2}$ for any states $|\alpha\rangle, |\beta\rangle, |\gamma\rangle, |\delta\rangle$, we rewrite $\mathrm{Adv}(\mathcal{A})$ shown in the above equation into

$$
\begin{aligned}
\mathrm{Adv}(\mathcal{A}) \leq & \sqrt{\|\mathrm{M}_{m^*}|\psi_1\rangle\|^2 + \|\mathrm{M}_{m^*}|\psi_0\rangle\|^2} \\
& \cdot \sqrt{\begin{aligned} &\left\| \mathrm{M}_{m^*} (\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H + |\psi_G\rangle) \right\|^2 \\ &+ \left\| \mathrm{M}_{m^*} (\mathrm{U}_1)^\dagger (\mathrm{I} - \mathrm{M}_{m^*}) \mathrm{O}_H (\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle) \right\|^2 \end{aligned}} . \\
\overset{(a)}{=} & \sqrt{\|\mathrm{M}_{m^*}|\psi_1\rangle\|^2 + \|\mathrm{M}_{m^*}|\psi_0\rangle\|^2} \\
& \cdot \sqrt{\begin{aligned} &\left\| \mathrm{M}_{m^*} \mathrm{O}_H (\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H + |\psi_G\rangle) \right\|^2 \\ &+ \left\| \mathrm{M}_{m^*} \mathrm{O}_H (\mathrm{U}_1)^\dagger (\mathrm{I} - \mathrm{M}_{m^*}) \mathrm{O}_H (\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle) \right\|^2 \end{aligned}} .
\end{aligned} \tag{3}
$$

Here (a) uses the fact that O_H is a unitary operation and it commutes with M_{m^*}.

• **MRE-Step-3**: Here, we will relate the above two sum of square norms with the success probabilities of two one-wayness attackers, that is, Eq. (4) and Eq. (5). For the $\|\mathrm{M}_{m^*}|\psi_1\rangle\|^2 + \|\mathrm{M}_{m^*}|\psi_0\rangle\|^2$, by the superposition oracle trick[6] given in [4], we can construct two one-wayness attackers $\mathcal{B}_1$ and $\mathcal{B}_2$ such that

$$
\begin{aligned}
4 \cdot \Pr[m^* \leftarrow \mathcal{B}_1^{G,H}] &= \|\mathrm{M}_{m^*}|\psi_1\rangle\|^2 = \|\mathrm{M}_{m^*}(\mathrm{O}_H \mathrm{U}_1 \mathrm{O}_H |\psi\rangle - \mathrm{O}_G \mathrm{U}_1 \mathrm{O}_G |\psi\rangle)\|^2, \\
4 \cdot \Pr[m^* \leftarrow \mathcal{B}_2^{G,H}] &= \|\mathrm{M}_{m^*}|\psi_0\rangle\|^2 = \|\mathrm{M}_{m^*}(\mathrm{O}_H |\psi\rangle - \mathrm{O}_G |\psi\rangle)\|^2.
\end{aligned}
$$

Note that there is an extra constant factor "4" due to the using of superposition oracle trick. Now we can merge $\mathcal{B}_1$ and $\mathcal{B}_2$ into $\mathcal{B}_3$, which uniformly chooses i from $\{1, 2\}$, runs $\mathcal{B}_i$ and outputs its final output. Obviously, $\Pr[m^* \leftarrow \mathcal{B}_3^{G,H}]$ is equal to $1/2 \cdot \Pr[m^* \leftarrow \mathcal{B}_1^{G,H}] + 1/2 \cdot \Pr[m^* \leftarrow \mathcal{B}_2^{G,H}]$. Hence we obtain

$$
4 \cdot 2 \cdot \Pr[m^* \leftarrow \mathcal{B}_3^{G,H}] = \|\mathrm{M}_{m^*}|\psi_1\rangle\|^2 + \|\mathrm{M}_{m^*}|\psi_0\rangle\|^2. \tag{4}
$$

[6] Roughly speaking, this trick first performs $\mathrm{O}_{G,H} := (\mathrm{O}_H \otimes |+\rangle\langle+|) + (\mathrm{O}_G \otimes |-\rangle\langle-|)$ on $|\psi\rangle|0\rangle$ to obtain the state $1/2(\mathrm{O}_H|\psi\rangle - \mathrm{O}_G|\psi\rangle)|1\rangle + 1/2(\mathrm{O}_H|\psi\rangle + \mathrm{O}_G|\psi\rangle)|0\rangle$, and then measures the last qubit with the desired measurement result 1, which makes the whole state to collapse into $\mathrm{O}_H|\psi\rangle - \mathrm{O}_G|\psi\rangle$ (non-normalized).

For the another sum of the square norms $\|\mathrm{M}_{m^*}(U_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2$ and $\|\mathrm{M}_{m^*}(U_1)^\dagger(\mathrm{I}-\mathrm{M}_{m^*})O_H(U_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2$, we first define a one-wayness attacker as:

$\mathcal{B}_4^{G,H,1_S}$: Given oracle access to G, H and 1_S, it works as follows. Here 1_S is the indicator function of $S = \{m^*\}$, that is, $1_S(x) = 1$ if $x = m^*$ and 0 otherwise.

1. Prepare $|\psi_H\rangle + |\psi_G\rangle$ by using the superposition oracle trick [4].
2. Perform the measurement $\mathbb{M}_{\mathcal{A}} = \{\mathrm{M}_0^{\mathcal{A}}, \mathrm{M}_1^{\mathcal{A}}\}$ with the desired measurement result 1, if the measurement result is 0, abort and output $\perp$.
3. Apply $\mathrm{O}_H(\mathrm{U}_2)^\dagger$, then perform projective measurement $\mathbb{M}_{m^*} := \{\chi_0, \chi_1\}$ on the oracle's input register by querying 1_S. Here $\chi_0 = \mathrm{I}-\mathrm{M}_{m^*}$ and $\chi_1 = \mathrm{M}_{m^*}$[7].
   (a) If the measurement result is 1, measure the oracle's input register to get m^*, then abort and output m^*.
   (b) If the measurement result is 0, apply $\mathrm{O}_H(\mathrm{U}_1)^\dagger$ and then perform the measurement $\mathbb{M}_{m^*}$ on the oracle's input register again.
      i. If the second measurement $\mathbb{M}_{m^*}$ has measurement result 1, measure the oracle's input register to get m^*, then abort and output m^*. Otherwise, abort and output $\perp$.

Let E_1 be the classical event that the measurement $\mathbb{M}_{\mathcal{A}}$ has result 1 and the next first measurement $\mathbb{M}_{m^*}$ also has result 1. Let E_2 be the classical event that the measurement $\mathbb{M}_{\mathcal{A}}$ has result 1, then the first measurement $\mathbb{M}_{m^*}$ has result 0 and the next second measurement $\mathbb{M}_{m^*}$ has result 1.

At this point, we have a crucial observation that events E_1 and E_2 are mutually exclusive and

$$\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2 = 4 \cdot \Pr[E_1 : \mathcal{B}_4^{G,H,1_S}],$$
$$\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_1)^\dagger(\mathrm{I}-\mathrm{M}_{m^*})\mathrm{O}_H(\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2 = 4 \cdot \Pr[E_2 : \mathcal{B}_4^{G,H,1_S}].$$

Here we have an extra constant factor "4" since our $\mathcal{B}_4$ uses the superposition oracle trick. Indeed, by the definition, E_1 and E_2 are obviously mutually exclusive. For the $\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2$, the $\mathrm{M}_1^{\mathcal{A}}$ and M_{m^*} actually represent that the measurement $\mathbb{M}_{\mathcal{A}}$ and the first measurement $\mathbb{M}_{m^*}$ both have result 1, i.e. E_1 occurs. For the $\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_1)^\dagger(\mathrm{I}-\mathrm{M}_{m^*})\mathrm{O}_H(\mathrm{U}_2)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2$, the $\mathrm{M}_1^{\mathcal{A}}$ represents that the measurement $\mathbb{M}_{\mathcal{A}}$ has result 1, the subsequent $(\mathrm{I}-\mathrm{M}_{m^*})$ represents that the first measurement $\mathbb{M}_{m^*}$ has result 0, and the final M_{m^*} represents that the second measurement $\mathbb{M}_{m^*}$ has result 1, i.e. E_2 occurs. Consequently, we can compute

[7] Roughly speaking, to perform $\{\mathrm{I}-\mathrm{M}_{m^*}, \mathrm{M}_{m^*}\}$ on a state $|\phi\rangle := \sum_{x,y}|x,y\rangle$, we first query the oracle 1_S to obtain $\sum_y |m^*, y\rangle|1\rangle + \sum_{x\neq m^*,y}|x,y\rangle|0\rangle$, and then measure the last qubit. If the measurement result is 1, the state $|\phi\rangle$ collapses into $\mathrm{M}_{m^*}|\phi\rangle = \sum_y |m^*, y\rangle$(non-normalized), and we can further measure the first register to get m^*.

$$
\begin{aligned}
&4\cdot\Pr[m^*\leftarrow\mathcal{B}_4^{G,H,1_S}]\overset{(a)}{=}4\cdot\Pr[E_1\vee E_2:\mathcal{B}_4^{G,H,1_S}]\\
&\overset{(b)}{=}4\cdot\Pr[E_1:\mathcal{B}_4^{G,H,1_S}]+4\cdot\Pr[E_2:\mathcal{B}_4^{G,H,1_S}]\\
&=\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_2)^\dagger\mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle+|\psi_G\rangle)\|^2+\\
&\quad\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_1)^\dagger(\mathrm{I}-\mathrm{M}_{m^*})\mathrm{O}_H(\mathrm{U}_2)^\dagger\mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle+|\psi_G\rangle)\|^2.
\end{aligned}
\tag{5}
$$

Here (a) follows from the definition of our one-wayness attacker $\mathcal{B}_4$, (b) uses the fact that events E_1 and E_2 are mutually exclusive. Now, by Eq. (4) and Eq. (5), we can rewrite the $\mathrm{Adv}(\mathcal{A})$ shown in Eq. (3) into

$$\mathrm{Adv}(\mathcal{A})\leq\sqrt{4\cdot 2\cdot\Pr[m^*\leftarrow\mathcal{B}_3^{G,H}]}\cdot\sqrt{4\cdot\Pr[m^*\leftarrow\mathcal{B}_4^{G,H,1_S}]}.$$

Let $\mathcal{B}$ be a one-wayness attacker that runs both $\mathcal{B}_3$ and $\mathcal{B}_4$, outputs m^* if either of these two outputs m^*, and outputs $\perp$ otherwise. Obviously, we have $\max\{\Pr[m^*\leftarrow\mathcal{B}_3^{G,H}],\Pr[m^*\leftarrow\mathcal{B}_4^{G,H,1_S}]\}\leq\Pr[m^*\leftarrow\mathcal{B}^{G,H,1_S}]$, thus

$$\mathrm{Adv}(\mathcal{A})\leq 4\cdot\sqrt{2}\cdot\Pr[m^*\leftarrow\mathcal{B}^{G,H,1_S}].$$

That is to say, for the distinguisher $\mathcal{A}$ with query depth $d=2$, our MRE technique provides an upper bound of $\mathrm{Adv}(\mathcal{A})$ as $4\cdot\sqrt{d}\cdot\mathrm{Adv}(\mathcal{B})$, where $\mathrm{Adv}(\mathcal{B})$ is the probability that $\mathcal{B}^{G,H,1_S}$ successfully finds an element in $S=\{m^*\}$.

Note that $\mathcal{B}_4$ constructed above has a special structure that first measures (i.e. performs $\mathbb{M}_{\mathcal{A}}$), then rewinds and extracts (i.e. performs *rewinding* operations $\mathrm{O}_H(\mathrm{U}_2)^\dagger$, $\mathrm{O}_H(\mathrm{U}_1)^\dagger$ and measurement $\mathbb{M}_{m^*}$ to extract m^*). The same structure is inherited by the final $\mathcal{B}$ since it directly runs $\mathcal{B}_4$. Actually, that is precisely why we call our technique described above the Measure-Rewind-Extract (MRE) technique. In addition, when describing our contribution in Section 1.1, we mentioned that MRE technique is a combination of the MRM technique [25] and the semi-classical oracle technique [1]. Here, we explain this statement.

- Firstly, our MRE technique follows the framework of MRM technique, which first divides the state $|\psi_H\rangle-|\psi_G\rangle$, then rewrites $\mathrm{Adv}(\mathcal{A})$ into a product of some square norms like Eq. (3), and finally designs a one-wayness attacker $\mathcal{B}$ based on these square norms such that $\mathrm{Adv}(\mathcal{A})$ can be upper bounded by utilizing $\mathrm{Adv}(\mathcal{B})$. However, different from the MRM technique which uses a hybrid argument to divide $|\psi_H\rangle-|\psi_G\rangle$, our MRE technique uses the projector M_{m^*} to directly divide $|\psi_H\rangle-|\psi_G\rangle$. Note that in the MRM technique, it is this hybrid argument that inevitably introduce a loss of query depth d.
- Secondly, due to using M_{m^*} to divide $|\psi_H\rangle-|\psi_G\rangle$, we have to construct a one-wayness attacker that performs the measurement $\mathbb{M}_{m^*}$ on the oracle's input register, aiming at extracting m^* from $\mathcal{A}$'s oracle query. In fact, $\mathbb{M}_{m^*}$ is the "semi-classical oracle $\mathcal{O}_{\{m^*\}}^{SC}$" designed in [1], and the core idea of the semi-classical oracle technique is exactly to extract m^* from $\mathcal{A}$'s oracle query by performing $\mathcal{O}_{m^*}^{SC}$ (i.e. the $\mathbb{M}_{m^*}$).

Hence, our MRE technique can be viewed as a combination of the MRM technique [25] and the semi-classical oracle technique [1].

Remark 3 (Concern about the measurement $\mathbb{M}_{m^*}$). Roughly speaking, by using the semi-classical oracle technique, [1] constructed a one-wayness attacker $\mathcal{B}$ and proves that $|\Pr[1 \leftarrow \mathcal{A}^{O}] - \Pr[1 \leftarrow \mathcal{A}^{O\backslash S}]| \leq \sqrt{O(d) \cdot \mathrm{Adv}(\mathcal{B})}$. Here $O\backslash S$ is a new oracle that first performs the measurement $\mathbb{M}_{m^*}$ on the oracle's input register and then queries O. This inequality actually shows that using $\mathbb{M}_{m^*}$ to measure $\mathcal{A}$'s oracle query will disrupt $\mathcal{A}$'s computation, resulting in a probability difference $\sqrt{O(d) \cdot \mathrm{Adv}(\mathcal{B})}$. Note that our one-wayness attacker $\mathcal{B}_4$ constructed above rewound $\mathcal{A}$ and performed $\mathbb{M}_{m^*}$, so one might be concerned that $\mathcal{B}_4$ disrupts $\mathcal{A}$'s computation and hence will inevitably introduce the loss $\sqrt{O(d) \cdot \mathrm{Adv}(\mathcal{B})}$. Here, we emphasize that we do not need to concern about this. Firstly, our construction of $\mathcal{B}_4$ does not directly convert $\mathcal{A}^{O}$ into $\mathcal{A}^{O\backslash S}$. Secondly, what our MRE technique does is, first derive the value

$$\begin{aligned} p := &\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_2)^{\dagger}\mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2 + \\ &\|\mathrm{M}_{m^*}\mathrm{O}_H(\mathrm{U}_1)^{\dagger}(\mathrm{I} - \mathrm{M}_{m^*})\mathrm{O}_H(\mathrm{U}_2)^{\dagger}\mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\|^2, \end{aligned}$$

then construct $\mathcal{B}_4$ and clearly prove that its success probability $\Pr[m^* \leftarrow \mathcal{B}_4]$ equals to $1/4 \cdot p$ (i.e. Eq. (5)), and finally use this property to prove $\mathrm{Adv}(\mathcal{A}) \leq 4 \cdot \sqrt{d} \cdot \mathrm{Adv}(\mathcal{B})$. That is to say, for $\mathcal{B}_4$, what we need to care about is only whether it satisfies $\Pr[m^* \leftarrow \mathcal{B}_4] = 1/4 \cdot p$, and we do not need to concern about whether $\mathcal{B}_4$ disrupts the computation of $\mathcal{A}$ and hence introduces an additional loss.

Use MRE technique to prove our MRE-O2H theorem Although the above explanation of the MRE technique only considers the case where the query depth d is 2, we can directly lift it through induction to account for the case with arbitrary query depth d. Hence, the above explanation of the MRE technique has proved the following fixed version of MRE-O2H theorem.

Theorem 1 (Fixed O2H with MRE, informal). *For a fixed tuple (G, H, S) and a quantum distinguisher $\mathcal{A}$ that makes parallel queries with query depth d, we can construct a quantum one-wayness attacker $\mathcal{B}^{G,H,1_S}$ such that*

$$|\Pr[b = 1 : b \leftarrow \mathcal{A}^{G}] - \Pr[b = 1 : b \leftarrow \mathcal{A}^{H}]| \leq 4 \cdot \sqrt{d} \cdot \mathrm{Adv}(\mathcal{B}). \tag{6}$$

Here $\mathrm{Adv}(\mathcal{B})$ is the probability that $\mathcal{B}^{G,H,1_S}$ successfully finds an element in S.

For the random version of MRE-O2H theorem, where (G, H, S) is sampled from an arbitrary joint distribution D, we can directly prove it by averaging over $(G, H, S) \leftarrow D$ in Eq. (6).

QROM security proofs of FO-like transformations Note that [4, Theorem 5] has shown that $\mathsf{FO}^{\not\perp}$ (resp. $\mathsf{FO}^{\perp}$) is as secure as $\mathsf{FO}_m^{\not\perp}$ (resp. $\mathsf{FO}_m^{\perp}$) and vice versa. Hence, in our paper, for the FO-like transformations $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}_m^{\not\perp}$, $\mathsf{FO}^{\perp}$ and $\mathsf{FO}_m^{\perp}$, we only consider the the IND-CCA security of $\mathsf{FO}^{\not\perp}$ and $\mathsf{FO}_m^{\perp}$ in the QROM.

Our proof outline is shown in Fig. 1. In this outline, we utilize the property that $\mathsf{FO}^{\not\perp} = \mathsf{U}^{\not\perp} \circ \mathsf{T}$ and $\mathsf{FO}_m^{\perp} = \mathsf{U}_m^{\perp} \circ \mathsf{T}$ introduced in [15]. Here, T transforms a randomized PKE (rPKE) scheme into a deterministic PKE (dPKE) scheme, $\mathsf{U}^{\not\perp}$ and $\mathsf{U}_m^{\perp}$ both transform a dPKE scheme into a KEM scheme.

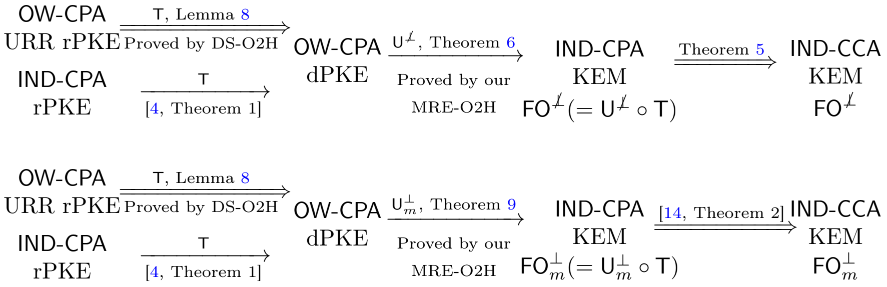

Fig. 1. Proof outline of $\mathsf{FO}^{\not\perp}$ and $\mathsf{FO}^{\perp}_m$ in the QROM. All the security proofs shown in this outline avoid the square-root advantage loss. The double arrow indicates a tight security proof, while the single arrow indicates a non-tight security proof. In this outline, we abbreviate unique randomness recoverable as URR for the sake of simplicity.

As shown in Fig. 1, by using the DS-O2H theorem [4], we prove that T can tightly transform a OW-CPA-secure and unique randomness recoverable rPKE scheme into a OW-CPA-secure dPKE scheme in the QROM. Based on this proof, we give an IND-CCA security proof of $\mathsf{FO}^{\not\perp}$ and $\mathsf{FO}^{\perp}_m$ from the OW-CPA security, while avoiding the square-root advantage loss.

In addition, we emphasize that our security proof of $\mathsf{U}^{\not\perp}$ shown in Fig. 1 (i.e. Theorem 6) does not rely on the η-injective assumption used in [25]. Our observation is that, it is not necessary to require the encryption algorithm dEnc of the underlying dPKE scheme to be nearly injective, we can only need dEnc to satisfy the following weaker property:

For a m^* uniformly sampled from the message space and (pk, sk) generated by the key generation algorithm, there does not exist $m \neq m^*$ such that $\mathsf{dEnc}_{pk}(m) = \mathsf{dEnc}_{pk}(m^*)$.

Indeed, according to [26, Lemma 4], the probability that the underlying dPKE scheme of $\mathsf{U}^{\not\perp}$ does not satisfy this property is negligible. Hence, we can remove the η-injective assumption in our security proof of $\mathsf{U}^{\not\perp}$.

1.3 Related Work

There are also some O2H variants that involve Zhandry's compressed oracle technique [34]. For example, the [8, Theorem 10], the [23, Theorem C.5], the [16, Theorem 6] and the [14, Theorem 1]. Intuitively speaking, these O2H variants are all obtained by generalizing the SC-O2H theorem [1] to work with the compressed oracle technique. However, they all have a drawback: their final upper bound suffers from the square-root advantage loss.

We note that in [30,32], the authors proved that the $\mathsf{IND\text{-}CCA}$ security of the transformation SXY (also known as $\mathsf{U}_m^{\not\perp}$) can be tightly reduced to the $\mathsf{DS\text{-}IND}$ security of the underlying dPKE scheme in the QROM, where $\mathsf{DS\text{-}IND}$ is a non-standard security assumption. Indeed, although they provided a tight QROM security proof of $\mathsf{U}_m^{\not\perp}$, which is used to construct the FO-like transformation $\mathsf{FO}_m^{\not\perp}(= \mathsf{U}_m^{\not\perp} \circ \mathsf{T})$ [15], the cost is that they used a non-standard security assumption and the underlying PKE scheme is restricted to a dPKE scheme.

In our QROM security proofs of the FO-like transformations, when the security of the underlying PKE scheme is $\mathsf{OW\text{-}CPA}$, we introduce an addition assumption named unique randomness recoverable. This assumption is actually a stronger variant of the assumption named randomness recoverable, which, as far as we know, was first introduced in [27,12] to achieve a tight ROM security proof of the transformation T. According to the definition of unique randomness recoverable given in Definition 6, we find that it is not a security assumption but just a constraint on the encryption algorithm. Meanwhile, we find that the NTRU-based PKE schemes generally satisfy the assumption of unique randomness recoverable, and we also provide a rough explanation in Supplementary Material C for completeness. Hence, the unique randomness recoverable does not seem to be a pretty strong assumption.

In a concurrent work, under the assumption that the underlying PKE scheme is unique randomness recoverable, Bao et al. [2] introduced a variant of the DS-O2H theorem [4] and then used it to give a tight security proof of T in the QROM. Their security bound $4 \cdot \epsilon$ is even tighter than the security bound $10 \cdot \epsilon$ achieved by our security proof of T (Lemma 8) in the QROM. Here ϵ is the security bound of the underlying PKE scheme.

2 Preliminaries

2.1 Notation

By $[\![x = y]\!]$ we denote a bit that is 1 if $x = y$ and otherwise 0. For a finite set S, $x \xleftarrow{\$} S$ denotes that x is an element uniformly sampled from set S. For a distribution D, $x \leftarrow D$ denotes that x is chosen according to distribution D. For a game $\mathbf{G}$ in the security proof, $1 \leftarrow \mathbf{G}$ denotes that $\mathbf{G}$ finally returns 1. $\Pr[A : B, C]$ (or $\Pr_C[A : B]$, $\Pr_{B,C}[A]$ for short) is the probability that the predicate A keeps true where all variables in A are conditioned according to predicates B and C. For an algorithm $\mathcal{A}$, we use $\mathcal{T}_{\mathcal{A}}$ to denote its running time.

2.2 Quantum Background

We refer to [28] for detailed basics of quantum computation and quantum information. In Supplementary Material A.1, we introduce several important quantum notions used in this paper.

2.3 Quantum Random Oracle Model

The random oracle model (ROM) is an ideal model in which a uniformly random function $H : X \to Y$ is selected, and all parties have access to H. In real schemes, the random oracle H is implemented using a suitable hash function. In the quantum setting, since the hash function can be evaluated in superposition, the ROM should be lifted into the quantum random oracle model (QROM) [5], where all parties have quantum access to the random oracle. In the QROM, we take the random oracle H as a unitary operation $O_H : |x, y\rangle \mapsto |x, y \oplus H(x)\rangle$.

Here, we state the following two lemmas that are used throughout this paper.

Lemma 1 (Simulate the QROM [33, Theorem 6.1]). *Let O be a random oracle, H be a function uniformly sampled from the set of $2q$-wise independent functions. For any algorithm $\mathcal{A}$ that makes at most q quantum queries, we have*

$$\Pr[b = 1 : b \leftarrow \mathcal{A}^H] = \Pr[b = 1 : b \leftarrow \mathcal{A}^O].$$

Remark 4. This lemma shows that we can use a $2q$-wise independent function to perfectly simulate a quantum accessible random oracle with query bound q. Indeed, as stated in [30, Section 2.2], this simulation has an $O(q^2)$ running time increase since it has to compute a $2q$-wise independent function for each query.

Lemma 2 (Generic quantum distinguishing problem with bounded probabilities [17, Lemma 2.9]). *Let $\delta \in [0, 1]$ and $\mathcal{M}$ be a finite set. Let $N_1 : \mathcal{M} \to \{0, 1\}$ be a random function such that, for each $m \in \mathcal{M}$, $N_1(m) = 1$ with probability δ_m ($\delta_m \leq \delta$), and $N_1(m) = 0$ with probability $1 - \delta_m$. Let $N_2 : \mathcal{M} \to \{0, 1\}$ be a constant function such that $N_2(m) = 0$ for all $m \in \mathcal{M}$. For any algorithm $\mathcal{A}$ that makes at most q quantum queries, we have*

$$\left|\Pr\left[b = 1 : b \leftarrow \mathcal{A}^{N_1}\right] - \Pr\left[b = 1 : b \leftarrow \mathcal{A}^{N_2}\right]\right| \leq 8(q + 1)^2 \cdot \delta.$$

Now, we recall the Measure-Rewind-Measure One-Way to Hiding (MRM-O2H) theorem introduced in [25].

Theorem 2 (MRM-O2H [25, Lemma 3.3]). *Let $G, H : X \to Y$ be random functions, $S \subseteq X$ be a random set and $z \in Z$ be a random bitstring. The tuple (G, H, S, z) may have arbitrary joint distribution D and satisfies that $\forall x \notin S$, $G(x) = H(x)$. Let $\mathcal{A}^O$ ($O \in \{G, H\}$) be a quantum oracle algorithm that makes parallel queries with query depth d and query width n. Define*

$$P_{\text{left}} := \Pr_{(G,H,S,z) \leftarrow D}[b = 1 : b \leftarrow \mathcal{A}^H(z)], P_{\text{right}} := \Pr_{(G,H,S,z) \leftarrow D}[b = 1 : b \leftarrow \mathcal{A}^G(z)].$$

Then, we can construct an algorithm $\mathcal{D}^{G,H}(z)$ which has the following two properties:

- *Let* $\mathrm{Adv}(\mathcal{D}) := \Pr\left[T_{\mathcal{D}} \cap S \neq \varnothing : T_{\mathcal{D}} \leftarrow \mathcal{D}^{G,H}(z), (G, H, S, z) \leftarrow D\right]$, *then*

$$|P_{\text{left}} - P_{\text{right}}| \leq 4d \cdot \mathrm{Adv}(\mathcal{D}).$$

- *$\mathcal{D}^{G,H}(z)$ makes parallel queries to G and H both with query depth at most $3d$ and query width n. Its running time can be bounded as $\mathcal{T}_{\mathcal{D}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}}$.*

3 O2H with Measure-Rewind-Extract (MRE)

In this section, we focus on the tuple (G, H, S, z), where G, H are functions with domain X and codomain Y, S is a subset of X and G, H, S satisfy that $\forall x \notin S$, $G(x) = H(x)$, $z \in Z$ is a bitstring that can depend on G, H, S. Let 1_S denote the indicator function of the set S, that is, $1_S(x) = 1$ if $x \in S$ and 0 otherwise.

Here we introduce the following two lemmas that will be used in later proofs, and their proofs can be found in Supplementary Material A.3.

Lemma 3 ([25, Lemma 3.1]). *For any states $|\phi_1\rangle$ and $|\phi_2\rangle$, we have*

$$\left| \| |\phi_1\rangle \|^2 - \| |\phi_2\rangle \|^2 \right| \leq |(|\phi_1\rangle - |\phi_2\rangle, |\phi_1\rangle + |\phi_2\rangle)|.$$

Lemma 4. *For any states $|\varphi_1\rangle, \ldots, |\varphi_n\rangle$ and $|\phi_1\rangle, \ldots, |\phi_n\rangle$, we have*

$$\sum_{i=1}^{n} |(|\varphi_i\rangle, |\phi_i\rangle)| \leq \sqrt{\sum_{i=1}^{n} \| |\varphi_i\rangle \|^2} \cdot \sqrt{\sum_{i=1}^{n} \| |\phi_i\rangle \|^2}.$$

Now we prove our new O2H theorem. Same as [25], we first prove the fixed version, where the tuple (G, H, S, z) is fixed. Then, we extend it to the random version, where the tuple (G, H, S, z) can have an arbitrary joint distribution.

Theorem 3 (Fixed O2H with MRE). *Let $G, H : X \to Y$ be fixed functions, $S \subseteq X$ be a fixed set and $z \in Z$ be a fixed bitstring. The tuple (G, H, S, z) satisfies that $\forall x \notin S$, $G(x) = H(x)$. Let $\mathcal{A}^O$ $(O \in \{G, H\})$ be a quantum oracle algorithm that makes parallel queries with query depth d and query width n. Define*

$$P_{\text{left}}^{GHSz} := \Pr[b = 1 : b \leftarrow \mathcal{A}^H(z)],\ P_{\text{right}}^{GHSz} := \Pr[b = 1 : b \leftarrow \mathcal{A}^G(z)].$$

Then, we can construct an algorithm $\mathcal{D}^{G,H,1_S}(z)$ which has the following two properties:

- *Let $\mathrm{Adv}(\mathcal{D}) := \Pr[T_{\mathcal{D}} \cap S \neq \varnothing : T_{\mathcal{D}} \leftarrow \mathcal{D}^{G,H,1_S}(z)]$, then*

$$\left| P_{\text{left}}^{GHSz} - P_{\text{right}}^{GHSz} \right| \leq 4\sqrt{d} \cdot \mathrm{Adv}(\mathcal{D}). \tag{7}$$

- *$\mathcal{D}^{G,H,1_S}(z)$ makes parallel queries to G, H and 1_S all with query depth at most $3d$ and query width n. Its running time can be bounded as $\mathcal{T}_{\mathcal{D}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}}$.*

Proof. Following the proof of [25, Lemma 3.2], we denote $O_G^{\otimes n}$ (resp. $O_H^{\otimes n}$) as the n-width parallel quantum oracle for G (resp. H). Then, we define a new quantum oracle

$$O_{G,H}^{\otimes n} := (O_H^{\otimes n} \otimes |+\rangle\langle+|) + (O_G^{\otimes n} \otimes |-\rangle\langle-|).$$

Here $|+\rangle := (|0\rangle + |1\rangle)/\sqrt{2}$ and $|-\rangle := (|0\rangle - |1\rangle)/\sqrt{2}$. Indeed, the oracle $O_{G,H}^{\otimes n}$ uses an auxiliary single quantum bit as the controlling bit: if the state of this controlling bit is $|+\rangle$ (resp. $|-\rangle$), the oracle $O_H^{\otimes n}$ (resp. $O_G^{\otimes n}$) will be queried. As analyzed in [4,25], $O_{G,H}^{\otimes n}$ can be efficiently implemented by applying a Hadamard gate before and after a conditional operation, which queries $O_H^{\otimes n}$ (resp. $O_G^{\otimes n}$) if the controlling bit is in the state $|0\rangle$ (resp. $|1\rangle$).

Based on the above notations, we introduce the following lemma that will be used later. It shows that we can use $O_{G,H}^{\otimes n}$ to get a uniform superposition of the sum and difference of the state generated by $O_G^{\otimes n}$ and $O_H^{\otimes n}$, and those states are entangled with the controlling bit of $O_{G,H}^{\otimes n}$. This lemma can be easily proved by induction, and we omit the detailed proof for the sake of simplicity.

Lemma 5. *Let $V_1, \ldots, V_t$ $(t \in \mathbb{N}^+)$ be any unitary operation that can be applied between $O_G^{\otimes n}/O_H^{\otimes n}$ queries and $|\phi\rangle$ be any appropriate initial state. Let the controlling bit of $O_{G,H}^{\otimes n}$ be in the initial state $|0\rangle$. Then*

$$\begin{aligned}\prod_{i=1}^{t}[V_i O_{G,H}^{\otimes n}](|\phi\rangle|0\rangle) =& \frac{1}{2}\left(\prod_{i=1}^{t}[V_i O_H^{\otimes n}]|\phi\rangle + \prod_{i=1}^{t}[V_i O_G^{\otimes n}]|\phi\rangle\right) \otimes |0\rangle \\ &+ \frac{1}{2}\left(\prod_{i=1}^{t}[V_i O_H^{\otimes n}]|\phi\rangle - \prod_{i=1}^{t}[V_i O_G^{\otimes n}]|\phi\rangle\right) \otimes |1\rangle.\end{aligned}$$

Here $\prod_{i=1}^{t}[V_i O_{G,H}^{\otimes n}](|\phi\rangle|0\rangle) := V_t O_{G,H}^{\otimes n} V_{t-1} O_{G,H}^{\otimes n} \ldots V_2 O_{G,H}^{\otimes n} V_1 O_{G,H}^{\otimes n}(|\phi\rangle|0\rangle)$, and analogously for $\prod_{i=1}^{t}[V_i O_H^{\otimes n}]|\phi\rangle$ and $\prod_{i=1}^{t}[V_i O_G^{\otimes n}]|\phi\rangle$.

Since any quantum oracle algorithm can be efficiently transformed into a unitary quantum oracle algorithm with the same query time and query depth (i.e. Fact 1 in Supplementary Material A.1), we assume $\mathcal{A}^O(z)$ to be unitary without loss of generality. Now, for $O \in \{G, H\}$, denote $|\psi_z\rangle$ as the initial state of $\mathcal{A}^O(z)$, and denote $U_1, \ldots, U_d$ as the unitary operations performed by $\mathcal{A}^O(z)$ between its (parallel) oracle queries. Then, the joint state of $\mathcal{A}^H(z)$ (resp. $\mathcal{A}^G(z)$) just before performing the final binary projective measurement $\mathbb{M}_{\mathcal{A}} := \{\mathrm{M}_0^{\mathcal{A}}, \mathrm{M}_1^{\mathcal{A}}\}$ can be written as

$$|\psi_H\rangle := \prod_{i=1}^{d}[U_i O_H^{\otimes n}]|\psi_z\rangle \;\; \text{(resp. } |\psi_G\rangle := \prod_{i=1}^{d}[U_i O_G^{\otimes n}]|\psi_z\rangle). \tag{8}$$

Since the measurement result of $\mathbb{M}_{\mathcal{A}}$ is the final output of $\mathcal{A}$, we can compute

$$\begin{aligned}\left|P_{\text{left}}^{GHSz} - P_{\text{right}}^{GHSz}\right| &= \left|\Pr[b=1 : b \leftarrow \mathcal{A}^H(z)] - \Pr[b=1 : b \leftarrow \mathcal{A}^G(z)]\right| \\ &= \left|\|\mathrm{M}_1^{\mathcal{A}}|\psi_H\rangle\|^2 - \|\mathrm{M}_1^{\mathcal{A}}|\psi_G\rangle\|^2\right| \\ &\overset{(a)}{\leq} \left|\left(\mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle - |\psi_G\rangle), \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\right)\right| \\ &\overset{(b)}{=} \left|\left(|\psi_H\rangle - |\psi_G\rangle, \left(\mathrm{M}_1^{\mathcal{A}}\right)^\dagger \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\right)\right| \\ &\overset{(c)}{=} \left|\left(|\psi_H\rangle - |\psi_G\rangle, \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\right)\right|.\end{aligned} \tag{9}$$

Here (a) is obtained by using Lemma 3. (b) uses the fact that $(A|\phi_1\rangle, B|\phi_2\rangle) = (|\phi_1\rangle, A^\dagger B|\phi_2\rangle)$ for any operators A, B and states $|\phi_1\rangle$, $|\phi_2\rangle$. (c) uses the fact that the projector $\mathrm{M}_1^{\mathcal{A}}$ is Hermitian and idempotent.

Next, we focus on the states $|\psi_H\rangle$ and $|\psi_G\rangle$. We will give them a decomposition (i.e., the following Eq. (13) and Eq. (14)) according to a projector on the oracle's input register. In the following, we first define this projector which we denote as $\mathrm{M}_{S^{\oplus n}}$, and then introduce some properties of it.

- **The definition of projector** $\mathrm{M}_{S^{\oplus n}}$. Since $\mathcal{A}$ makes parallel queries with query width n, the query state of $\mathcal{A}$ can be written as

$$|query\rangle := \sum_{\mathrm{in},\mathrm{out},aux} \alpha^{\mathrm{out}}_{\mathrm{in},aux} |in_1\rangle|out_1\rangle \cdots |in_n\rangle|out_n\rangle|aux\rangle.$$

Here $\mathrm{in} := (in_1, \ldots, in_n) \in X^{\otimes n}$, $\mathrm{out} := (out_1, \ldots, out_n) \in Y^{\otimes n}$ and $aux \in \{0,1\}^*$. $|in_1\rangle \cdots |in_n\rangle$ (resp. $|out_1\rangle \cdots |out_n\rangle$) is the basis state of the oracle's input register IN (resp. oracle's output register OUT), $|aux\rangle$ is the basis state of some auxiliary registers that may be entangled with IN and OUT. Furthermore, we have

$$O_G^{\otimes n}|query\rangle = \sum_{\mathrm{in},\mathrm{out},aux} \alpha^{\mathrm{out}}_{\mathrm{in},aux} |in_1\rangle|out_1 \oplus G(in_1)\rangle \cdots |in_n\rangle|out_n \oplus G(in_n)\rangle|aux\rangle,$$

$$O_H^{\otimes n}|query\rangle = \sum_{\mathrm{in},\mathrm{out},aux} \alpha^{\mathrm{out}}_{\mathrm{in},aux} |in_1\rangle|out_1 \oplus H(in_1)\rangle \cdots |in_n\rangle|out_n \oplus H(in_n)\rangle|aux\rangle.$$

Define set

$$S^{\oplus n} := \{(in_1, \ldots, in_n) | in_1, \ldots, in_n \in X, \exists i \in \{1, \ldots, n\} \text{ s.t. } in_i \in S\}.$$

Then, we define a projector on the oracle's input register IN as

$$\mathrm{M}_{S^{\oplus n}} := \sum_{(in_1,\ldots,in_n)\in S^{\oplus n}} |in_1\rangle \cdots |in_n\rangle\langle in_1| \cdots \langle in_n|. \tag{10}$$

Let I_{IN} be the identity operator on the oracle's input register IN, we have

$$\mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}} = \sum_{(in_1,\ldots,in_n)\notin S^{\oplus n}} |in_1\rangle \cdots |in_n\rangle\langle in_1| \cdots \langle in_n|.$$

- **Properties satisfied by** $\mathrm{M}_{S^{\oplus n}}$**,** $O_G^{\otimes n}$ **and** $O_H^{\otimes n}$. Using the fact that $G(x) = H(x)$ for all $x \notin S$, it is easy to see that

$$O_H^{\otimes n}(\mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}})|query\rangle = O_G^{\otimes n}(\mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}})|query\rangle. \tag{11}$$

Note that querying the oracles $O_G^{\otimes n}$ and $O_H^{\otimes n}$ does not change the state on the oracle's input register IN, we also have

$$O_G^{\otimes n}\mathrm{M}_{S^{\oplus n}} = \mathrm{M}_{S^{\oplus n}}O_G^{\otimes n},\ O_H^{\otimes n}\mathrm{M}_{S^{\oplus n}} = \mathrm{M}_{S^{\oplus n}}O_H^{\otimes n}. \tag{12}$$

That is, $O_G^{\otimes n}$ and $O_H^{\otimes n}$ both commute with $\mathrm{M}_{S^{\oplus n}}$.

In particular, we introduce the following lemma about $\mathrm{M}_{S^{\oplus n}}$. It shows that we can implement the projective measurement $\{\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}}, \mathrm{M}_{S^{\oplus n}}\}$ on the oracle's input register IN by quantum querying the 1_S. The proof of this lemma is very simple and is given in Supplementary Material A.4.

Lemma 6. *Recall that 1_S is the indicator function of the set S, that is, $1_S(x)=1$ if $x\in S$ and 0 otherwise. Let $\chi_0 := \mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}}$ and $\chi_1 := \mathrm{M}_{S^{\oplus n}}$. Then, the binary projective measurement $\mathbb{M}_{S^{\oplus n}} := \{\chi_0,\chi_1\}$ on the oracle's input register IN can be implemented by making two parallel queries to 1_S with query width n.*

Now, we define the following states

$$|\psi_H^j\rangle := \prod_{i=1}^{j}[U_i O_H^{\otimes n}]|\psi_z\rangle \ (1\le j\le d) \text{ and } |\psi_H^0\rangle := |\psi_z\rangle.$$

Let $\prod_{i=1}^{0}[U_i O_H^{\otimes n}] := \mathrm{I}_{\mathcal{A}}$, where $\mathrm{I}_{\mathcal{A}}$ denotes the identity operator on $\mathcal{A}$'s whole register. Then, for $1\le j\le d$, we can compute

$$\begin{aligned}|\psi_H^j\rangle &= \prod_{i=1}^{j}[U_i O_H^{\otimes n}]|\psi_z\rangle = U_j O_H^{\otimes n}(\mathrm{M}_{S^{\oplus n}}+\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}})\prod_{i=1}^{j-1}[U_i O_H^{\otimes n}]|\psi_z\rangle \\ &= U_j O_H^{\otimes n}(\mathrm{M}_{S^{\oplus n}}+\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}})|\psi_H^{j-1}\rangle \\ &= U_j O_H^{\otimes n}\mathrm{M}_{S^{\oplus n}}|\psi_H^{j-1}\rangle + U_j O_H^{\otimes n}(\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}})|\psi_H^{j-1}\rangle.\end{aligned}$$

Hence, by induction, it is not hard to obtain

$$\begin{aligned}|\psi_H\rangle = |\psi_H^d\rangle = &\prod_{i=1}^{d}\left[U_i O_H^{\otimes n}(\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}})\right]|\psi_z\rangle + U_d O_H^{\otimes n}\mathrm{M}_{S^{\oplus n}}|\psi_H^{d-1}\rangle \\ &+\sum_{k=1}^{d-1}\prod_{j=k+1}^{d}\left[U_j O_H^{\otimes n}(\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}})\right]\left(U_k O_H^{\otimes n}\mathrm{M}_{S^{\oplus n}}\right)|\psi_H^{k-1}\rangle.\end{aligned} \tag{13}$$

Similarly, for $|\psi_G\rangle$, we can derive the following equation using the definitions $|\psi_G^j\rangle := \prod_{i=1}^{j}[U_j O_G^{\otimes n}]|\psi_z\rangle$ $(1\le j\le d)$ and $|\psi_G^0\rangle := |\psi_z\rangle$.

$$\begin{aligned}|\psi_G\rangle = |\psi_G^d\rangle = &\prod_{i=1}^{d}\left[U_i O_G^{\otimes n}(\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}})\right]|\psi_z\rangle + U_d O_G^{\otimes n}\mathrm{M}_{S^{\oplus n}}|\psi_G^{d-1}\rangle \\ &+\sum_{k=1}^{d-1}\prod_{j=k+1}^{d}\left[U_j O_G^{\otimes n}(\mathrm{I}_{IN}-\mathrm{M}_{S^{\oplus n}})\right]\left(U_k O_G^{\otimes n}\mathrm{M}_{S^{\oplus n}}\right)|\psi_G^{k-1}\rangle.\end{aligned} \tag{14}$$

Note that I_{IN} denotes the identity operator on the oracle's input register IN, and $\mathrm{I}_{\mathcal{A}}$ denotes the identity operator on $\mathcal{A}$'s whole register. Then, based on the states $|\psi_H^j\rangle$ and $|\psi_G^j\rangle$ $(0\le j\le d)$ defined above, we define the following notations that will be used later:

$$
\begin{aligned}
\chi_0 &:= \mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}}, \chi_1 := \mathrm{M}_{S^{\oplus n}}, \\
|\psi_{H+G}\rangle &:= \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle), \\
|\psi^i_{H-G}\rangle &:= O_H^{\otimes n}|\psi_H^{i-1}\rangle - O_G^{\otimes n}|\psi_G^{i-1}\rangle \ (1 \le i \le d), \\
|\psi^{S,i}_{H-G}\rangle &:= O_H^{\otimes n}\mathrm{M}_{S^{\oplus n}}|\psi_H^{i-1}\rangle - O_G^{\otimes n}\mathrm{M}_{S^{\oplus n}}|\psi_G^{i-1}\rangle \\
&\overset{(a)}{=} \mathrm{M}_{S^{\oplus n}}O_H^{\otimes n}|\psi_H^{i-1}\rangle - \mathrm{M}_{S^{\oplus n}}O_G^{\otimes n}|\psi_G^{i-1}\rangle \ (1 \le i \le d), \\
U^{\text{non-}S}_{d\leftarrow k+1} &:= \prod_{j=k+1}^{d} \left[U_j O_H^{\otimes n}(\mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}})\right] \ (1 \le k \le d-1), \\
U^{\text{non-}S}_{d\leftarrow d+1} &:= \mathrm{I}_{\mathcal{A}}.
\end{aligned} \tag{15}
$$

Here (a) follows from Eq. (12).

By using Eq. (11), it is not hard to check that

$$
\prod_{i=1}^{d} \left[U_i O_H^{\otimes n}(\mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}})\right] |\psi_z\rangle = \prod_{i=1}^{d} \left[U_i O_G^{\otimes n}(\mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}})\right] |\psi_z\rangle. \tag{16}
$$

Then, combining Eq. (11) with Eq. (13) to Eq. (16), we obtain $|\psi_H\rangle - |\psi_G\rangle = \sum_{k=1}^{d} U^{\text{non-}S}_{d\leftarrow k+1} U_k |\psi^{S,k}_{H-G}\rangle$. Combine this equation with Eq. (9), we can compute

$$
\begin{aligned}
&\left|P^{GHSz}_{\text{left}} - P^{GHSz}_{\text{right}}\right| \\
&\le \left|\left(|\psi_H\rangle - |\psi_G\rangle, \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)\right)\right| \overset{(a)}{=} |(|\psi_H\rangle - |\psi_G\rangle, |\psi_{H+G}\rangle)| \\
&= \left|\sum_{k=1}^{d} \left(U^{\text{non-}S}_{d\leftarrow k+1} U_k |\psi^{S,k}_{H-G}\rangle, |\psi_{H+G}\rangle\right)\right| \\
&\overset{(b)}{\le} \sum_{k=1}^{d} \left|\left(U^{\text{non-}S}_{d\leftarrow k+1} U_k |\psi^{S,k}_{H-G}\rangle, |\psi_{H+G}\rangle\right)\right| \\
&\overset{(c)}{=} \sum_{k=1}^{d} \left|\left(|\psi^{S,k}_{H-G}\rangle, (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle\right)\right| \\
&\overset{(d)}{=} \sum_{k=1}^{d} \left|\left(\chi_1 |\psi^{S,k}_{H-G}\rangle, (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle\right)\right| \\
&\overset{(e)}{=} \sum_{k=1}^{d} \left|\left(\chi_1 |\psi^{S,k}_{H-G}\rangle, \chi_1 (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle\right)\right| \\
&\overset{(f)}{=} \sum_{k=1}^{d} \left|\left(|\psi^{S,k}_{H-G}\rangle, \chi_1 (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle\right)\right| \\
&\overset{(g)}{\le} \sqrt{\sum_{k=1}^{d} \||\psi^{S,k}_{H-G}\rangle\|^2} \cdot \sqrt{\sum_{k=1}^{d} \left\|\chi_1 (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle\right\|^2}.
\end{aligned} \tag{17}
$$

Here (a) follows the definition of $|\psi_{H+G}\rangle$ given in Eq. (15). (b) uses the triangle inequality. (c) uses the basic property of inner product. (d) and (f) use the fact that $|\psi^{S,i}_{H-G}\rangle$ defined in Eq. (15) satisfies $|\psi^{S,i}_{H-G}\rangle = \chi_1|\psi^{S,i}_{H-G}\rangle$ for $1 \le i \le d$. (e) uses the fact that χ_1 is Hermitian and idempotent. (g) uses Lemma 4.

Recall that $|\psi_z\rangle$ is the initial state of $\mathcal{A}^O(z)$ $(O \in \{G, H\})$, $U_1, \ldots, U_d$ are the unitary operations performed by $\mathcal{A}^O(z)$ between its parallel oracle queries, and $\mathbb{M}_{\mathcal{A}} = \{\mathrm{M}_0^{\mathcal{A}}, \mathrm{M}_1^{\mathcal{A}}\}$ is the final projective measurement performed by $\mathcal{A}^O(z)$. Now, we define the following algorithms $\mathcal{B}_i^{G,H}(z)$ $(1 \le i \le d)$, $\mathcal{B}^{G,H}(z)$, $\mathcal{C}^{G,H,1_S}(z)$ and $\mathcal{D}^{G,H,1_S}(z)$, all with the aim to extract an element from the set S:

- Algorithm $\mathcal{B}_i^{G,H}(z)$ $(1 \le i \le d)$. This algorithm has initial pure state $|\psi_z\rangle|0\rangle$, where $|0\rangle$ is the state of the controlling bit of $O_{G,H}^{\otimes n}$. As mentioned earlier, this controlling bit is used by $O_{G,H}^{\otimes n}$ to determine whether to query $O_G^{\otimes n}$ or $O_H^{\otimes n}$. $\mathcal{B}_i^{G,H}(z)$ first applies $O_{G,H}^{\otimes n}\prod_{j=1}^{i-1}[U_j O_{G,H}^{\otimes n}]$ (here we let $\prod_{j=1}^{0}[U_j O_{G,H}^{\otimes n}] = \mathrm{I}_{\mathcal{A}}$), then measures the controlling bit of $O_{G,H}^{\otimes n}$ in the computational basis:
  1. If the measurement result is 1, $\mathcal{B}_i^{G,H}(z)$ measures the oracle's input register IN in the computational basis, and outputs the result $T_{\mathcal{B}_i}$.
  2. If the measurement result is 0, $\mathcal{B}_i^{G,H}(z)$ outputs $\perp$.

- Algorithm $\mathcal{B}^{G,H}(z)$. This algorithm first uniformly chooses i from $\{1, \ldots, d\}$, and then runs $\mathcal{B}_i^{G,H}(z)$ directly. $\mathcal{B}^{G,H}(z)$ finally outputs $\mathcal{B}_i^{G,H}(z)$'s output and we denote it as $T_{\mathcal{B}}$ if it is not $\perp$.

- Algorithm $\mathcal{C}^{G,H,1_S}(z)$. This algorithm has initial pure state $|\psi_z\rangle|0\rangle$, where $|0\rangle$ is the state of the controlling bit of $O_{G,H}^{\otimes n}$. $\mathcal{C}^{G,H,1_S}(z)$ first applies the operation $\prod_{i=1}^{d}[U_i O_{G,H}^{\otimes n}]$, then performs the projective measurement $\mathbb{M}_{\mathcal{A}} = \{\mathrm{M}_0^{\mathcal{A}}, \mathrm{M}_1^{\mathcal{A}}\}$ and measures the controlling bit of $O_{G,H}^{\otimes n}$ in the computational basis:
  1. If the measurement result of $\mathbb{M}_{\mathcal{A}}$ is 0 or the measurement result of the controlling bit is 1, $\mathcal{C}^{G,H,1_S}(z)$ outputs $\perp$.
  2. If the measurement result of $\mathbb{M}_{\mathcal{A}}$ is 1 and the measurement result of the controlling bit is 0, $\mathcal{C}^{G,H,1_S}(z)$ performs the following operations:
     (a) Initially, let $i = d$.
     (b) Apply (rewinding operation) $O_H^{\otimes n}(U_i)^{\dagger}$, then perform the projective measurement $\mathbb{M}_{S^{\oplus n}} = \{\chi_0, \chi_1\}$ on the oracle's input register IN by querying 1_S (Lemma 6). If the measurement result of $\mathbb{M}_{S^{\oplus n}}$ is 1, abort, measure IN in the computational basis and output the result $T_{\mathcal{C}}$.
     (c) If the measurement result of $\mathbb{M}_{S^{\oplus n}}$ is 0 and $i > 1$, let $i = i - 1$ and repeat the above step. If the measurement result of $\mathbb{M}_{S^{\oplus n}}$ is 0 and $i = 1$, abort and output $\perp$.

- Algorithm $\mathcal{D}^{G,H,1_S}(z)$. This algorithm runs $\mathcal{B}^{G,H}(z)$ and $\mathcal{C}^{G,H,1_S}(z)$, and it outputs $\perp$ if they both outputs $\perp$. Otherwise, $\mathcal{D}^{G,H,1_S}(z)$ outputs $T_{\mathcal{D}}$, where
  - $T_{\mathcal{D}} = T_{\mathcal{B}}$ if $\mathcal{B}^{G,H}(z)$ outputs $T_{\mathcal{B}}$ and $\mathcal{C}^{G,H,1_S}(z)$ outputs $\perp$.
  - $T_{\mathcal{D}} = T_{\mathcal{C}}$ if $\mathcal{B}^{G,H}(z)$ outputs $\perp$ and $\mathcal{C}^{G,H,1_S}(z)$ outputs $T_{\mathcal{C}}$.
  - $T_{\mathcal{D}} = T_{\mathcal{B}} \cup T_{\mathcal{C}}$ if $\mathcal{B}^{G,H}(z)$ outputs $T_{\mathcal{B}}$ and $\mathcal{C}^{G,H,1_S}(z)$ outputs $T_{\mathcal{C}}$.

As for the running time, one can easily check that $\mathcal{T}_{\mathcal{B}} \lesssim \mathcal{T}_{\mathcal{A}}$ and $\mathcal{B}^{G,H}(z)$ makes parallel queries to G and H both with query depth at most d and query width n. Since $\mathcal{C}^{G,H,1_S}(z)$ performs the rewinding operations, we have $\mathcal{T}_{\mathcal{C}} \lesssim 2 \cdot \mathcal{T}_{\mathcal{A}}$ and $\mathcal{C}^{G,H,1_S}(z)$ makes parallel queries to G, H and 1_S all with query depth at most $2d$ and query width n. By the definition of $\mathcal{D}^{G,H,1_S}(z)$, we can conclude that $\mathcal{T}_{\mathcal{D}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}}$, and $\mathcal{D}^{G,H,1_S}(z)$ makes parallel queries to G, H and 1_S all with query depth at most $3d$ and query width n.

We define

$$\begin{aligned}
\mathrm{Adv}(\mathcal{B}_i) &:= \Pr[T_{\mathcal{B}_i} \cap S \neq \varnothing : T_{\mathcal{B}_i} \leftarrow \mathcal{B}_i^{G,H}(z)] \ i \in \{1, \ldots, d\}, \\
\mathrm{Adv}(\mathcal{B}) &:= \Pr[T_{\mathcal{B}} \cap S \neq \varnothing : T_{\mathcal{B}} \leftarrow \mathcal{B}^{G,H}(z)], \\
\mathrm{Adv}(\mathcal{C}) &:= \Pr[T_{\mathcal{C}} \cap S \neq \varnothing : T_{\mathcal{C}} \leftarrow \mathcal{C}^{G,H,1_S}(z)], \\
\mathrm{Adv}(\mathcal{D}) &:= \Pr[T_{\mathcal{D}} \cap S \neq \varnothing : T_{\mathcal{D}} \leftarrow \mathcal{D}^{G,H,1_S}(z)].
\end{aligned}$$

For the algorithm $\mathcal{B}_i^{G,H}(z)$ $(i \in \{1, \ldots, d\})$, since we let $\prod_{j=1}^{0}[U_j O_{G,H}^{\otimes n}] := \mathrm{I}_{\mathcal{A}}$, by Lemma 5 and the definition of states $|\psi_{H-G}^{i}\rangle$ and $|\psi_{H-G}^{S,i}\rangle$ given in Eq. (15), it is not hard to check that

$$\mathrm{Adv}(\mathcal{B}_i) = \left\| \frac{|\psi_{H-G}^{S,i}\rangle}{\||\psi_{H-G}^{i}\rangle\|} \right\|^2 \cdot \left\| \frac{1}{2} |\psi_{H-G}^{i}\rangle \right\|^2 = \frac{1}{4} \cdot \||\psi_{H-G}^{S,i}\rangle\|^2.$$

Then, by the definition of algorithm $\mathcal{B}^{G,H}(z)$, we have

$$\mathrm{Adv}(\mathcal{B}) = \sum_{i=1}^{d} \frac{1}{d} \mathrm{Adv}(\mathcal{B}_i) = \sum_{i=1}^{d} \frac{1}{4d} \cdot \||\psi_{H-G}^{S,i}\rangle\|^2. \tag{18}$$

For the algorithm $\mathcal{C}^{G,H,1_S}(z)$, by Lemma 5 and the definition of $|\psi_H\rangle$ and $|\psi_G\rangle$ given in Eq. (8), we can write the state of $\mathcal{C}^{G,H,1_S}(z)$ just before performing the $\mathbb{M}_{\mathcal{A}} = \{\mathrm{M}_0^{\mathcal{A}}, \mathrm{M}_1^{\mathcal{A}}\}$ and the measurement of the controlling bit of $O_{G,H}^{\otimes n}$ as

$$\frac{1}{2}\left(|\psi_H\rangle + |\psi_G\rangle\right) \otimes |0\rangle + \frac{1}{2}\left(|\psi_H\rangle - |\psi_G\rangle\right) \otimes |1\rangle.$$

The right half, $|0\rangle$ and $|1\rangle$, is the state of the controlling bit of $O_{G,H}^{\otimes n}$. Since $|\psi_{H+G}\rangle := \mathrm{M}_1^{\mathcal{A}}(|\psi_H\rangle + |\psi_G\rangle)$ (i.e. Eq. (15)), the probability that $\mathbb{M}_{\mathcal{A}}$ has result 1 and the measurement of the controlling bit of $O_{G,H}^{\otimes n}$ has result 0 is $\frac{1}{4}\||\psi_{H+G}\rangle\|^2$. Further, the state of $\mathcal{C}^{G,H,1_S}(z)$ will collapse into $|\psi_{H+G}\rangle / \||\psi_{H+G}\rangle\|$[8].

After $\mathbb{M}_{\mathcal{A}}$ obtains result 1 and the measurement of the controlling bit of $O_{G,H}^{\otimes n}$ obtains result 0, $\mathcal{C}^{G,H,1_S}(z)$ will rewind $U_1, \ldots, U_d$ and perform $\mathbb{M}_{S^{\oplus n}} = \{\chi_0, \chi_1\}$ to extract an element from the set S. We refer to this step of $\mathcal{C}^{G,H,1_S}(z)$ as the "rewind-extract" process, and in fact, we can restate the "rewind-extract" process as:

$$\underline{\mathbb{M}_{S^{\oplus n}}} O_H^{\otimes n}(U_1)^\dagger \underleftarrow{\mathbb{M}_{S^{\oplus n}}} O_H^{\otimes n}(U_2)^\dagger \cdots \underleftarrow{\mathbb{M}_{S^{\oplus n}}} O_H^{\otimes n}(U_{d-1})^\dagger \underleftarrow{\mathbb{M}_{S^{\oplus n}}} O_H^{\otimes n}(U_d)^\dagger \frac{|\psi_{H+G}\rangle}{\||\psi_{H+G}\rangle\|}.$$

[8] In this notation, we omit the state of the controlling bit of $O_{G,H}^{\otimes n}$, since this bit is no longer used by the subsequent operations of $\mathcal{C}^{G,H,1_S}(z)$.

Here $|\psi_{H+G}\rangle/\||\psi_{H+G}\rangle\|$ is the initial pure state just before the "rewind-extract" process, and $\underleftarrow{\mathbb{M}_{S^{\oplus n}}}$ denotes the following conditional operation:

Perform $\mathbb{M}_{S^{\oplus n}}$ on the oracle's input register IN. If the measurement result is 1, measure IN in the computational basis and output the result $T_{\mathcal{C}}$. If the measurement result is 0, proceed with the subsequent operations.

$\underline{\mathbb{M}_{S^{\oplus n}}}$ is the same as $\underleftarrow{\mathbb{M}_{S^{\oplus n}}}$, except that it directly outputs $\perp$ if the measurement result of $\mathbb{M}_{S^{\oplus n}}$ is 0. Obviously, $\mathcal{C}^{G,H,1_S}(z)$ performs $\mathbb{M}_{S^{\oplus n}}$ at most d times.

Recall that $\chi_1 := \mathrm{M}_{S^{\oplus n}}$ (i.e. Eq. (15)). By the definition of the projector $\mathrm{M}_{S^{\oplus n}}$ given in Eq. (10), we can conclude that $T_{\mathcal{C}}$ must satisfy $T_{\mathcal{C}} \cap S \neq \varnothing$ if $T_{\mathcal{C}}$ is obtained by measuring the oracle's input register IN (in the computational basis) under the condition that the measurement result of $\mathbb{M}_{S^{\oplus n}}$ just performed was 1. This means that, as long as one of the $\mathbb{M}_{S^{\oplus n}}$ performed by $\mathcal{C}^{G,H,1_S}(z)$ in the "rewinding-extract" process yields a measurement result of 1, $\mathcal{C}^{G,H,1_S}(z)$ will output a set $T_{\mathcal{C}}$ such that $T_{\mathcal{C}} \cap S \neq \varnothing$.

Now, we define the following mutually exclusive events that may be occurred in the "rewinding-extract" process of $\mathcal{C}^{G,H,1_S}(z)$:

E_i: The measurement result of the first $i-1$ measurements $\mathbb{M}_{S^{\oplus n}}$ are all 0, and the measurement result of the i-th measurement $\mathbb{M}_{S^{\oplus n}}$ is 1. $(1 \leq i \leq d)$

According to the definition of the operation $U^{\text{non-}S}_{d\leftarrow k+1}$ $(1 \leq k \leq d)$ given in Eq. (15), one can check that

$$\Pr[E_i] = \left\| \chi_1 O_H^{\otimes n} (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle \right\|^2 \frac{1}{\||\psi_{H+G}\rangle\|^2} \ (1 \leq i \leq d, i+k = d+1).$$

Then, we can compute

$$\begin{aligned}
\mathrm{Adv}(\mathcal{C}) &= \frac{1}{4} \cdot \||\psi_{H+G}\rangle\|^2 \cdot \sum_{i=1}^{d} \Pr[E_i] \\
&= \frac{1}{4} \cdot \||\psi_{H+G}\rangle\|^2 \cdot \frac{\sum_{k=1}^{d} \left\| \chi_1 O_H^{\otimes n} (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle \right\|^2}{\||\psi_{H+G}\rangle\|^2} \\
&\overset{(a)}{=} \frac{1}{4} \cdot \sum_{k=1}^{d} \left\| O_H^{\otimes n} \chi_1 (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle \right\|^2 \\
&\overset{(b)}{=} \frac{1}{4} \cdot \sum_{k=1}^{d} \left\| \chi_1 (U_k)^\dagger \left(U^{\text{non-}S}_{d\leftarrow k+1}\right)^\dagger |\psi_{H+G}\rangle \right\|^2 .
\end{aligned} \tag{19}$$

Here (a) follows from the fact that $\chi_1 = \mathrm{M}_{S^{\oplus n}}$ and $\mathrm{M}_{S^{\oplus n}}$ commutes with $O_H^{\otimes n}$ (i.e. Eq. (12)), (b) uses the fact that $O_H^{\otimes n}$ is a unitary operation.

Combine Eq. (17), Eq. (18) with Eq. (19), we get

$$\left|P^{GHSz}_{\text{left}} - P^{GHSz}_{\text{right}}\right| \leq \sqrt{4d \cdot \mathrm{Adv}(\mathcal{B})} \cdot \sqrt{4 \cdot \mathrm{Adv}(\mathcal{C})}.$$

Since we have $\mathrm{Adv}(\mathcal{D}) \geq \max\{\mathrm{Adv}(\mathcal{B}), \mathrm{Adv}(\mathcal{C})\}$ by the definition of the algorithm $\mathcal{D}^{G,H,1_S}(z)$, we finally obtain $|P^{GHSz}_{\text{left}} - P^{GHSz}_{\text{right}}| \leq 4\sqrt{d} \cdot \mathrm{Adv}(\mathcal{D})$. □

Theorem 4 (Random O2H with MRE). *Let $G, H : X \to Y$ be random functions, $S \subseteq X$ be a random set and $z \in Z$ be a random bitstring. The tuple (G, H, S, z) may have arbitrary joint distribution D and satisfies that $\forall x \notin S$, $G(x) = H(x)$. Let $\mathcal{A}^O$ $(O \in \{G, H\})$ be a quantum oracle algorithm that makes parallel queries with query depth d and query width n. Define*

$$P_{\text{left}} := \Pr_{(G,H,S,z) \leftarrow D}[b = 1 : b \leftarrow \mathcal{A}^H(z)], P_{\text{right}} := \Pr_{(G,H,S,z) \leftarrow D}[b = 1 : b \leftarrow \mathcal{A}^G(z)].$$

Then, we can construct an algorithm $\mathcal{D}^{G,H,1_S}(z)$ which has the following two properties:

- *Let* $\text{Adv}(\mathcal{D}) := \Pr[T_{\mathcal{D}} \cap S \neq \varnothing : T_{\mathcal{D}} \leftarrow \mathcal{D}^{G,H,1_S}(z), (G, H, S, z) \leftarrow D]$*, then*
$$|P_{\text{left}} - P_{\text{right}}| \leq 4\sqrt{d} \cdot \text{Adv}(\mathcal{D}).$$
- *$\mathcal{D}^{G,H,1_S}(z)$ makes parallel queries to G, H and 1_S all with query depth at most $3d$ and query width n. Its running time can be bounded as $\mathcal{T}_{\mathcal{D}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}}$.*

Proof. Based on Eq. (7) in Theorem 3, we can directly prove this theorem by averaging over $(G, H, S, z) \leftarrow D$. □

Remark 5. In the proof of Theorem 3, we assume that $\mathcal{A}$ is a unitary quantum oracle algorithm by the well-known fact Fact 1 in Supplementary Material A.1, which shows that any quantum oracle algorithm can be efficiently transformed into a unitary one with the same query times and query depth. However, as mentioned in [22,35], that transformation has a linear space[9] expansion with the running time of the quantum oracle algorithm, since we need to use unitary operations to simulate the non-unitary computations. Indeed, both the MRM-O2H theorem [25] and our MRE-O2H theorem (Theorem 4) involve this linear space expansion. In our paper, we stress that we do not view the space expansion as a dominant factor since it is only linear and not exponential, and we only view the advantage loss and the running time as crucial factors.

4 Tighter IND-CCA Proofs of FO-like Transformations

In this section, we consider the IND-CCA security of FO-like transformations $\mathsf{FO}^{\not\perp}$, $\mathsf{FO}^{\not\perp}_m$, $\mathsf{FO}^{\perp}$ and $\mathsf{FO}^{\perp}_m$ in the QROM. Note that [4, Theorem 5] has shown that $\mathsf{FO}^{\not\perp}$ (resp. $\mathsf{FO}^{\perp}$) is as secure as $\mathsf{FO}^{\not\perp}_m$ (resp. $\mathsf{FO}^{\perp}_m$) and vice versa. Hence, we only prove the IND-CCA security of $\mathsf{FO}^{\not\perp}$ and $\mathsf{FO}^{\perp}_m$ in the QROM. Our proof idea consists of the following two steps:

1. We first prove that the IND-CCA security of $\mathsf{FO}^{\not\perp}$ and $\mathsf{FO}^{\perp}_m$ can be reduced to its IND-CPA security.

[9] Here the "space" refers to the number of quantum bits used by an algorithm.

2. Then, by using our MRE-O2H theorem (Theorem 4), we prove that the $\mathsf{IND\text{-}CPA}$ security of $\mathsf{FO}^{\not\perp}$ and $\mathsf{FO}_m^{\perp}$ can be reduced to the $\mathsf{IND\text{-}CPA}/\mathsf{OW\text{-}CPA}$ security of the underlying PKE scheme.

The advantage of our proof idea is that, for the $\mathsf{FO}^{\not\perp}$, the additional injectivity assumption assumed in the proof of [25, Theorem 4.6] can be removed. All the relevant security notions can be found in Supplementary Material A.2.

Before proving our results, we first review the transformation T designed in [15] and introduce three lemmas about T that will be used later.

Transformation T: Let $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a randomized PKE (rPKE) scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$. Let $H : \mathcal{M} \to \mathcal{R}$ be a hash function. We associate deterministic PKE (dPKE) scheme $\mathsf{T}[\mathsf{P}, H] := (\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}')$. The constituting algorithms of $\mathsf{T}[\mathsf{P}, H]$ are shown in Fig. 2.

```
Gen
    $(pk, sk) \leftarrow \mathsf{Gen}$
    return $(pk, sk)$
```

```
$\mathsf{Enc}'_{pk}(m \in \mathcal{M})$
    $c := \mathsf{Enc}_{pk}(m; H(m))$
    return $c$
```

```
$\mathsf{Dec}'_{sk}(c)$
    $m' := \mathsf{Dec}_{sk}(c)$
    if $m' = \perp \vee c \neq \mathsf{Enc}_{pk}(m'; H(m'))$
        return $\perp$
    else return $m'$
```

Fig. 2. Deterministic Public Key Encryption $\mathsf{T}[\mathsf{P}, H]$.

Lemma 7 (Security of T from $\mathsf{IND\text{-}CPA}$ [4, Theorem 1]). *Let P be an rPKE scheme with message space $\mathcal{M}$. Let $\mathcal{A}$ be a $\mathsf{OW\text{-}CPA}$ adversary against $\mathsf{T}[\mathsf{P}, H]$, making parallel quantum queries to the random oracle H with query depth d_H and query width n. Let $q_H := d_H \cdot n$.*

Then, we can construct an $\mathsf{IND\text{-}CPA}$ adversary $\mathcal{B}$ against P such that

$$\mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{T}[\mathsf{P},H]}(\mathcal{A}) \leq (d_H + 2) \cdot \left(\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + \frac{8(q_H + 1)}{|\mathcal{M}|} \right)$$

and $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}} + O(q_H^2)$.

Remark 6. The [4, Theorem 1] actually claims $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}}$. In the above lemma, we give a more detailed running time of $\mathcal{B}$ as $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}} + O(q_H^2)$. The additional $O(q_H^2)$ is because $\mathcal{B}$ needs to use a $2q_H$-wise independent function to simulate a quantum accessible random oracle with query bound q_H.

The following lemma also focuses on the $\mathsf{OW\text{-}CPA}$ security of T in the QROM, but the difference is that the following lemma does not require the underlying rPKE scheme to be $\mathsf{IND\text{-}CPA}$-secure.

Lemma 8 (Security of T from $\mathsf{OW\text{-}CPA}$). *Let $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a δ-correct rPKE scheme, and assume P is unique randomness recoverable with the recover algorithm Rec. Let $\mathcal{A}$ be a $\mathsf{OW\text{-}CPA}$ adversary against $\mathsf{T}[\mathsf{P}, H]$, making parallel quantum queries to the random oracle H with query depth d_H and query width n. Let $q_H := d_H \cdot n$.*

Then, we can construct a $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}$ against P such that

$$\mathsf{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{T}[\mathsf{P},H]}(\mathcal{A}) \leq 10 \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + 16 \cdot \delta$$

and $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}} + O(q_H^2) + O(q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$.

Proof. See Supplementary Material A.5. □

Lemma 9 ([26, Lemma 4]). *Let $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an rPKE scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$. Define a set w.r.t fixed (pk, sk) and function $H : \mathcal{M} \to \mathcal{R}$ as*

$$S^{collision}_{pk,sk,H} := \{m \in \mathcal{M} | \exists m' \neq m \ s.t. \ \mathsf{Enc}_{pk}(m'; H(m')) = \mathsf{Enc}_{pk}(m; H(m))\}.$$

Let Ω_H be the set of all functions $H : \mathcal{M} \to \mathcal{R}$. Then, if P is δ-correct, we have

$$\Pr\left[m^* \in S^{collision}_{pk,sk,H} : (pk, sk) \leftarrow \mathsf{Gen}, H \xleftarrow{\$} \Omega_H, m^* \xleftarrow{\$} \mathcal{M}\right] \leq 2 \cdot \delta.$$

4.1 FO-like transformation $\mathsf{FO}^{\not\perp}$

FO-like transformation $\mathsf{FO}^{\not\perp}$. Let $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an rPKE scheme with message space $\mathcal{M}$, randomness space $\mathcal{R}$ and ciphertext space $\mathcal{C}$. For a given set $\mathcal{K}$, let $H : \mathcal{M} \to \mathcal{R}$, $G : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$ be hash functions, let $\mathsf{F} : \mathcal{K}^{prf} \times \mathcal{C} \to \mathcal{K}$ be a pseudorandom function (PRF) with key space $\mathcal{K}^{prf}$. We associate KEM scheme

$$\mathsf{KEM}^{\not\perp} := \mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}] = (\mathsf{Gen}^{\not\perp}, \mathsf{Enca}, \mathsf{Deca}^{\not\perp})$$

that has key space $\mathcal{K}$. The constituting algorithms of $\mathsf{KEM}^{\not\perp}$ are given in Fig. 3.

```
$\mathsf{Gen}^{\not\perp}$
$(pk, sk) \leftarrow \mathsf{Gen}$
$s \xleftarrow{\$} \mathcal{K}^{prf}$
$sk' := (sk, s)$
return $(pk, sk')$
```

```
$\mathsf{Enca}(pk)$
$m \xleftarrow{\$} \mathcal{M}$
$c := \mathsf{Enc}_{pk}(m; H(m))$
$K := G(m, c)$
return $(K, c)$
```

```
$\mathsf{Deca}^{\not\perp}(sk' = (sk, s), c)$
$m' := \mathsf{Dec}_{sk}(c)$
if $m' = \perp \vee c \neq \mathsf{Enc}_{pk}(m'; H(m'))$
    return $K := \mathsf{F}(s, c)$
else return $K := G(m', c)$
```

Fig. 3. Key Encapsulation Mechanism $\mathsf{KEM}^{\not\perp}$.

We first prove the following theorem. It shows that in the QROM, the $\mathsf{IND\text{-}CPA}$ security of $\mathsf{KEM}^{\not\perp}$ implies its $\mathsf{IND\text{-}CCA}$ security.

Theorem 5 **($\mathsf{IND\text{-}CPA}$ of $\mathsf{KEM}^{\not\perp}$ $\overset{\text{QROM}}{\Rightarrow}$ $\mathsf{IND\text{-}CCA}$ of $\mathsf{KEM}^{\not\perp}$).** *Let rPKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be δ-correct. Let $\mathcal{A}$ be an* $\mathsf{IND\text{-}CCA}$ *adversary against* $\mathsf{KEM}^{\not\perp}$ $=\mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}]$*, making q_D classical queries to the decapsulation oracle, making parallel quantum queries to the random oracle H (resp. G) with query depth d_H (resp. d_G) and query width n. Let $q_H := d_H \cdot n$ and $q_G := d_G \cdot n$.*

Then, we can construct the following two adversaries:

- *A* PRF*-adversary $\mathcal{B}_1$ against* F *making at most q_D classical queries. The running time of $\mathcal{B}_1$ is $\mathcal{T}_{\mathcal{B}_1} \approx \mathcal{T}_{\mathcal{A}} + q_D \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Dec}}) + O(q_H^2 + q_G^2)$.*
- *An* $\mathsf{IND\text{-}CPA}$ *adversary $\mathcal{B}_2$ against* $\mathsf{KEM}^{\not\perp}$ *in the QROM. $\mathcal{B}_2$ makes parallel quantum queries to the random oracle H (resp. G) with query depth at most $d_H + d_G$ (resp. d_G) and query width n. The running time of $\mathcal{B}_2$ is $\mathcal{T}_{\mathcal{B}_2} \approx \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}} + O(q_G^2 + q_D^2)$.*

Adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ satisfy the following:

$$\mathrm{Adv}_{\mathsf{KEM}^{\not\perp}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A}) \leq \mathrm{Adv}_{\mathsf{F}}^{\mathsf{PRF}}(\mathcal{B}_1) + \mathrm{Adv}_{\mathsf{KEM}^{\not\perp}}^{\mathsf{IND\text{-}CPA}}(\mathcal{B}_2) + 16(2q_H + 2q_G + 1)^2 \cdot \delta.$$

Proof. The proof of this theorem is similar with the proof of [19, Theorem 1], and we present it in Supplementary Material A.6. □

Next, we focus on the $\mathsf{IND\text{-}CPA}$ security of $\mathsf{KEM}^{\not\perp}$ in the QROM. As introduced in [15], the $\mathsf{KEM}^{\not\perp}$ satisfies that

$$\mathsf{KEM}^{\not\perp} = \mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}] = \mathsf{U}^{\not\perp}[\mathsf{T}[\mathsf{P}, H], G, \mathsf{F}]. \tag{20}$$

Here transformation $\mathsf{U}^{\not\perp}$ transforms a dPKE scheme into a KEM scheme. For the $\mathsf{U}^{\not\perp}$, we can prove the following theorem, which shows that in the QROM, the $\mathsf{IND\text{-}CPA}$ security of $\mathsf{U}^{\not\perp}$ can be reduced to the $\mathsf{OW\text{-}CPA}$ security of the underlying dPKE scheme without the square-root advantage loss.

Theorem 6 **($\mathsf{OW\text{-}CPA}$ of dPKE $\overset{\text{QROM}}{\Rightarrow}$ $\mathsf{IND\text{-}CPA}$ of $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$).** *For a dPKE scheme* $\mathsf{dPKE} = (\mathsf{Gen}, \mathsf{dEnc}, \mathsf{dDec})$ *with message space $\mathcal{M}$, let $\mathcal{A}$ be an* $\mathsf{IND\text{-}CPA}$ *adversary against* $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$*, making parallel quantum queries to the random oracle G with query depth d_G and query width n. Let $q_G := d_G \cdot n$.*

Then, we can construct a $\mathsf{OW\text{-}CPA}$ *adversary $\mathcal{A}_1$ against* dPKE *such that*

$$\mathrm{Adv}_{\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) \leq 2\sqrt{d_G} \cdot \mathrm{Adv}_{\mathsf{dPKE}}^{\mathsf{OW\text{-}CPA}}(\mathcal{A}_1) + 2\sqrt{d_G} \cdot \Pr[E_{\mathsf{dPKE}}]$$

and $\mathcal{T}_{\mathcal{A}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{dEnc}} + O(q_G^2)$. Here E_{dPKE} is the following event:

$$(pk, sk) \leftarrow \mathsf{Gen},\ m^* \xleftarrow{\$} \mathcal{M},\ \exists m \neq m^* \text{ such that } \mathsf{dEnc}_{pk}(m) = \mathsf{dEnc}_{pk}(m^*).$$

Proof. We prove this theorem by directly applying our MRE-O2H theorem (Theorem 4), and we present the detailed proof in Supplementary Material A.8. □

Combining Theorem 6 with Lemma 7 and Lemma 8, we can prove the following result for the $\mathsf{IND\text{-}CPA}$ security of $\mathsf{KEM}^{\not\perp}$ in the QROM.

Theorem 7 ($\mathsf{IND\text{-}CPA}/\mathsf{OW\text{-}CPA}$ **of** $\mathsf{P} \overset{\text{QROM}}{\Rightarrow}$ $\mathsf{IND\text{-}CPA}$ **of** $\mathsf{KEM}^{\not\perp}$)**.** *Let* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a δ-correct rPKE scheme with message space $\mathcal{M}$. Let $\mathcal{A}$ be an* $\mathsf{IND\text{-}CPA}$ *adversary against* $\mathsf{KEM}^{\not\perp} = \mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}]$*, making parallel quantum queries to the random oracle H (resp. G) with query depth d_H (resp. d_G) and query width n. Let $q_H := d_H \cdot n$ and $q_G := d_G \cdot n$.*

Then, we can construct an $\mathsf{IND\text{-}CPA}$ *adversary $\mathcal{B}$ against* P *such that*

$$\begin{aligned}\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq & 2\sqrt{d_G}(6d_G + d_H + 3) \cdot \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + 4\sqrt{d_G} \cdot \delta \\ & + 16\sqrt{d_G}(6d_G + d_H + 3)\frac{(6q_G + 2q_H + 1)}{|\mathcal{M}|}.\end{aligned}$$

and $\mathcal{T}_{\mathcal{B}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}}$*. If* P *is also unique randomness recoverable with the recover algorithm* Rec*, we can also construct a* $\mathsf{OW\text{-}CPA}$ *adversary $\mathcal{B}_1$ against* P *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq 20\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}_1) + 36\sqrt{d_G} \cdot \delta$$

and $\mathcal{T}_{\mathcal{B}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G + q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$*.*

Proof. This theorem can be easily proved by utilizing Eq. (20) and Lemma 9, and we present the detailed proof in Supplementary Material A.9. □

Combine Theorem 5 with Theorem 7, we finally obtain the following corollary. It shows that, in the QROM, the $\mathsf{IND\text{-}CCA}$ security of $\mathsf{KEM}^{\not\perp}$ can be reduced to the $\mathsf{IND\text{-}CPA}/\mathsf{OW\text{-}CPA}$ security of the underlying rPKE scheme without the square-root advantage loss.

Corollary 1 ($\mathsf{IND\text{-}CPA}/\mathsf{OW\text{-}CPA}$ **of** $\mathsf{P} \overset{\text{QROM}}{\Rightarrow}$ $\mathsf{IND\text{-}CCA}$ **of** $\mathsf{KEM}^{\not\perp}$)**.** *Let* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a δ-correct rPKE scheme with message space $\mathcal{M}$. Let $\mathcal{A}$ be an* $\mathsf{IND\text{-}CCA}$ *adversary against* $\mathsf{KEM}^{\not\perp} = \mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}]$*, making q_D classical queries to the decapsulation oracle, making parallel quantum queries to the random oracle H (resp. G) with query depth d_H (resp. d_G) and query width n. Let $q_H := d_H \cdot n$ and $q_G := d_G \cdot n$.*

Then, we can construct the following two adversaries:

- *A* PRF*-adversary $\mathcal{B}_1$ against* F *making at most q_D classical queries. The running time of $\mathcal{B}_1$ is* $\mathcal{T}_{\mathcal{B}_1} \lesssim \mathcal{T}_{\mathcal{A}} + q_D \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Dec}}) + O(q_H^2 + q_G^2)$*.*
- *An* $\mathsf{IND\text{-}CPA}$ *adversary $\mathcal{B}_2$ against* P*. The running time of $\mathcal{B}_2$ is* $\mathcal{T}_{\mathcal{B}_2} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}} + O(q_G^2 + q_H^2 + q_D^2)$*.*

Adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ satisfy the following:

$$\begin{aligned}\mathrm{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq \mathrm{Adv}^{\mathsf{PRF}}_{\mathsf{F}}(\mathcal{B}_1) & + 2\sqrt{d_G}(7d_G + d_H + 3) \cdot \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}_2) \\ & + 16(2q_H + 2q_G + 1)^2 \cdot \delta + 4\sqrt{d_G} \cdot \delta \\ & + 16\sqrt{d_G}(7d_G + d_H + 3)\frac{(8q_G + 2q_H + 1)}{|\mathcal{M}|}.\end{aligned}$$

If P *is also unique randomness recoverable with the recover algorithm* Rec*, we can also construct following adversary:*

– *A* $\mathsf{OW\text{-}CPA}$ *adversary* $\mathcal{B}_3$ *against* P*. The running time of* $\mathcal{B}_3$ *is* $\mathcal{T}_{\mathcal{B}_3} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2 + q_D^2) + O(q_G + q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$.

Adversaries $\mathcal{B}_1$ *and* $\mathcal{B}_3$ *satisfy the following:*

$$\begin{aligned}\mathrm{Adv}_{\mathsf{KEM}^{\not\perp}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A}) \leq \mathrm{Adv}_{\mathsf{F}}^{\mathsf{PRF}}(\mathcal{B}_1) + 20\sqrt{d_G} \cdot \mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{B}_3) + 36\sqrt{d_G} \cdot \delta \\ + 16(2q_H + 2q_G + 1)^2 \cdot \delta.\end{aligned}$$

4.2 FO-like transformation $\mathsf{FO}_m^{\perp}$

Similar to Section 4.1, we use the following two steps to prove the IND-CCA security of $\mathsf{FO}_m^{\perp}$ in the QROM:

1. First, we introduce [14, Theorem 2], which shows that the IND-CCA security of $\mathsf{FO}_m^{\perp}$ can be reduced to its IND-CPA security.
2. Then, by using our MRE-O2H theorem (Theorem 4), we prove that the IND-CPA security of $\mathsf{FO}_m^{\perp}$ can be reduced to the IND-CPA/OW-CPA security of the underlying PKE scheme.

We present the detailed proofs in Supplementary Material A.10.

Acknowledgments. We thank Shujiao Cao, Tianshu Shan, Kexin Gao and Jiawei Bao for their helpful suggestions, and thank anonymous reviewers for their insightful comments. This work is supported by the National Natural Science Foundation of China (Grants No. 62172405) and the Key Research Program of the Chinese Academy of Sciences (Grant NO. ZDRW-XX-2022-1).

References

1. Ambainis, A., Hamburg, M., Unruh, D.: Quantum security proofs using semi-classical oracles. In: Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II. pp. 269–295. Springer (2019). https://doi.org/10.1007/978-3-030-26951-7_10
2. Bao, J., Ge, J., Xue, R.: Double-sided: Tight proofs for guessing games in the quantum random oracle model. unpublished manuscript (2024)
3. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: CCS '93, Proceedings of the 1st ACM Conference on Computer and Communications Security, Fairfax, Virginia, USA, November 3-5, 1993. pp. 62–73. ACM (1993). https://doi.org/10.1145/168588.168596
4. Bindel, N., Hamburg, M., Hövelmanns, K., Hülsing, A., Persichetti, E.: Tighter proofs of CCA security in the quantum random oracle model. In: Theory of Cryptography Conference. pp. 61–90. Springer (2019). https://doi.org/10.1007/978-3-030-36033-7_3
5. Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: Advances in Cryptology - ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings. pp. 41–69. Springer (2011). https://doi.org/10.1007/978-3-642-25385-0_3
6. Chen, Z., Lu, X., Jia, D., Li, B.: Implicit rejection in fujisaki-okamoto: Framework and a novel realization. In: Information Security - 25th International Conference, ISC 2022, Bali, Indonesia, December 18-22, 2022, Proceedings. pp. 110–130. Springer (2022). https://doi.org/10.1007/978-3-031-22390-7_8
7. Cramer, R., Shoup, V.: Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM J. Comput. **33**(1), 167–226 (2003). https://doi.org/10.1137/S0097539702403773
8. Czajkowski, J., Majenz, C., Schaffner, C., Zur, S.: Quantum lazy sampling and game-playing proofs for quantum indifferentiability. IACR Cryptol. ePrint Arch. p. 428 (2019), https://eprint.iacr.org/2019/428
9. Dent, A.W.: A designer's guide to kems. In: IMA International Conference on Cryptography and Coding. pp. 133–151. Springer (2003). https://doi.org/10.1007/978-3-540-40974-8_12
10. Don, J., Fehr, S., Majenz, C., Schaffner, C.: Online-extractability in the quantum random-oracle model. In: Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part III. pp. 677–706. Springer (2022). https://doi.org/10.1007/978-3-031-07082-2_24
11. Duman, J., Hartmann, D., Kiltz, E., Kunzweiler, S., Lehmann, J., Riepel, D.: Group action key encapsulation and non-interactive key exchange in the QROM. In: Advances in Cryptology - ASIACRYPT 2022 - 28th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5-9, 2022, Proceedings, Part II. pp. 36–66. Springer (2022). https://doi.org/10.1007/978-3-031-22966-4_2
12. Duman, J., Hövelmanns, K., Kiltz, E., Lyubashevsky, V., Seiler, G., Unruh, D.: A thorough treatment of highly-efficient NTRU instantiations. In: Public-Key Cryptography - PKC 2023 - 26th IACR International Conference on Practice and Theory of Public-Key Cryptography, Atlanta, GA, USA, May 7-10,

2023, Proceedings, Part I. pp. 65–94. Springer (2023). https://doi.org/10.1007/978-3-031-31368-4_3
13. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. J. Cryptol. **26**(1), 80–101 (2013). https://doi.org/10.1007/s00145-011-9114-1
14. Ge, J., Shan, T., Xue, R.: Tighter qcca-secure key encapsulation mechanism with explicit rejection in the quantum random oracle model. In: Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part V. pp. 292–324. Springer (2023). https://doi.org/10.1007/978-3-031-38554-4_10
15. Hofheinz, D., Hövelmanns, K., Kiltz, E.: A modular analysis of the fujisaki-okamoto transformation. In: Theory of Cryptography Conference. pp. 341–371. Springer (2017). https://doi.org/10.1007/978-3-319-70500-2_12
16. Hövelmanns, K., Hülsing, A., Majenz, C.: Failing gracefully: Decryption failures and the fujisaki-okamoto transform. In: Advances in Cryptology - ASIACRYPT 2022 - 28th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5-9, 2022, Proceedings, Part IV. pp. 414–443. Springer (2022). https://doi.org/10.1007/978-3-031-22972-5_15
17. Hövelmanns, K., Kiltz, E., Schäge, S., Unruh, D.: Generic authenticated key exchange in the quantum random oracle model. In: Public-Key Cryptography - PKC 2020 - 23rd IACR International Conference on Practice and Theory of Public-Key Cryptography, Edinburgh, UK, May 4-7, 2020, Proceedings, Part II. pp. 389–422. Springer (2020). https://doi.org/10.1007/978-3-030-45388-6_14
18. Hövelmanns, K., Majenz, C.: A note on failing gracefully: Completing the picture for explicitly rejecting fujisaki-okamoto transforms using worst-case correctness. IACR Cryptol. ePrint Arch. p. 1811 (2023), https://eprint.iacr.org/2023/1811
19. Jiang, H., Zhang, Z., Chen, L., Wang, H., Ma, Z.: Ind-cca-secure key encapsulation mechanism in the quantum random oracle model, revisited. In: Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part III. pp. 96–125. Springer (2018). https://doi.org/10.1007/978-3-319-96878-0_4
20. Jiang, H., Zhang, Z., Ma, Z.: Key encapsulation mechanism with explicit rejection in the quantum random oracle model. In: Public-Key Cryptography - PKC 2019 - 22nd IACR International Conference on Practice and Theory of Public-Key Cryptography, Beijing, China, April 14-17, 2019, Proceedings, Part II. pp. 618–645. Springer (2019). https://doi.org/10.1007/978-3-030-17259-6_21
21. Jiang, H., Zhang, Z., Ma, Z.: Tighter security proofs for generic key encapsulation mechanism in the quantum random oracle model. In: Post-Quantum Cryptography - 10th International Conference, PQCrypto 2019, Chongqing, China, May 8-10, 2019 Revised Selected Papers. pp. 227–248. Springer (2019). https://doi.org/10.1007/978-3-030-25510-7_13
22. Jiang, H., Zhang, Z., Ma, Z.: On the non-tightness of measurement-based reductions for key encapsulation mechanism in the quantum random oracle model. In: Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I. pp. 487–517. Springer (2021). https://doi.org/10.1007/978-3-030-92062-3_17
23. Katsumata, S., Kwiatkowski, K., Pintore, F., Prest, T.: Scalable ciphertext compression techniques for post-quantum kems and their applications. In: Advances

in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I. pp. 289–320. Springer (2020). https://doi.org/10.1007/978-3-030-64837-4_10

24. Kitagawa, F., Nishimaki, R.: KDM security for the fujisaki-okamoto transformations in the QROM. In: Public-Key Cryptography - PKC 2022 - 25th IACR International Conference on Practice and Theory of Public-Key Cryptography, Virtual Event, March 8-11, 2022, Proceedings, Part II. pp. 286–315. Springer (2022). https://doi.org/10.1007/978-3-030-97131-1_10
25. Kuchta, V., Sakzad, A., Stehlé, D., Steinfeld, R., Sun, S.: Measure-rewind-measure: Tighter quantum random oracle model proofs for one-way to hiding and CCA security. In: Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III. pp. 703–728. Springer (2020). https://doi.org/10.1007/978-3-030-45727-3_24
26. Liu, X., Wang, M.: Qcca-secure generic key encapsulation mechanism with tighter security in the quantum random oracle model. In: Public-Key Cryptography - PKC 2021 - 24th IACR International Conference on Practice and Theory of Public Key Cryptography, Virtual Event, May 10-13, 2021, Proceedings, Part I. pp. 3–26. Springer (2021). https://doi.org/10.1007/978-3-030-75245-3_1
27. Lyubashevsky, V., Seiler, G.: NTTRU: truly fast NTRU using NTT. IACR Trans. Cryptogr. Hardw. Embed. Syst. **2019**(3), 180–201 (2019). https://doi.org/10.13154/TCHES.V2019.I3.180-201
28. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information (10th Anniversary edition). Cambridge University Press (2016)
29. NIST: National institute for standards and technology. post quantum crypto project. https://csrc.nist.gov/projects/post-quantum-cryptography (2017)
30. Saito, T., Xagawa, K., Yamakawa, T.: Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In: Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part III. pp. 520–551. Springer (2018). https://doi.org/10.1007/978-3-319-78372-7_17
31. Unruh, D.: Revocable quantum timed-release encryption. J. ACM **62**(6), 49:1–49:76 (2015). https://doi.org/10.1145/2817206
32. Xagawa, K., Yamakawa, T.: (tightly) qcca-secure key-encapsulation mechanism in the quantum random oracle model. In: Post-Quantum Cryptography - 10th International Conference, PQCrypto 2019, Chongqing, China, May 8-10, 2019 Revised Selected Papers. pp. 249–268. Springer (2019). https://doi.org/10.1007/978-3-030-25510-7_14
33. Zhandry, M.: Secure identity-based encryption in the quantum random oracle model. In: Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings. pp. 758–775. Springer (2012). https://doi.org/10.1007/978-3-642-32009-5_44
34. Zhandry, M.: How to record quantum queries, and applications to quantum indifferentiability. In: Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II. pp. 239–268. Springer (2019). https://doi.org/10.1007/978-3-030-26951-7_9

35. Zhandry, M.: The space-time cost of purifying quantum computations. In: 15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA. pp. 102:1–102:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2024). https://doi.org/10.4230/LIPICS.ITCS.2024.102

A Supplementary Material

A.1 Quantum Background

A quantum system (register) Q is a complex Hilbert space $\mathcal{H}_Q$ with an inner product $\langle\cdot|\cdot\rangle$, and notation like '$|\cdot\rangle$' or '$\langle\cdot|$' is called the Dirac notation. We denote $\mathcal{H}_Q = \mathbb{C}[X]$ if Q is defined over a finite set X. The orthonormal basis of $\mathbb{C}[X]$ is $\{|x\rangle\}_{x\in X}$, where the basis state $|x\rangle$ is labeled by the element x of X. We also refer to $\{|x\rangle\}_{x\in X}$ as the computational basis. The norm of a state $|\psi\rangle$ is defined as $\||\psi\rangle\| := \sqrt{\langle\psi|\psi\rangle}$, and we say $|\psi\rangle$ is a unit vector if $\||\psi\rangle\| = 1$.

Composite quantum system. Given quantum systems Q_1 and Q_2, we call tensor product $Q_1 \otimes Q_2$ is the composite quantum system and the product state is $|\psi_1\rangle \otimes |\psi_2\rangle \in Q_1 \otimes Q_2$, where $|\psi_1\rangle \in Q_1$, $|\psi_2\rangle \in Q_2$. We sometimes also abbreviate $|\psi_1\rangle \otimes |\psi_2\rangle$ into $|\psi_1\rangle|\psi_2\rangle$ for simplicity. If the composite quantum system $Q_1 \otimes Q_2$ is in the state $|\psi\rangle$, we say that the systems Q_0 and Q_1 are entangled if $|\psi\rangle$ can not be written as a product state $|\psi_1\rangle\otimes|\psi_2\rangle$ (here $|\psi_1\rangle \in Q_1$, $|\psi_2\rangle \in Q_2$). Otherwise, we say that the systems Q_0 and Q_1 are un-entangled.

A single qubit is a quantum system defined over $\{0, 1\}$. A single qubit in superposition is a linear combination vector $|b\rangle = \alpha|0\rangle + \beta|1\rangle$ of two computational basis states $|0\rangle$ and $|1\rangle$ with $\alpha, \beta \in \mathbb{C}^2$ and $|\alpha|^2 + |\beta|^2 = 1$, α, β are the probability amplitudes of $|b\rangle$. An n-qubit system is $Q^{\otimes n}$ where Q is single qubit system.

Evolution of quantum systems. A pure state $|\psi\rangle$ can be manipulated by performing a unitary operation U, and the resulting state can be denoted as $|\psi'\rangle = U|\psi\rangle$. For any unitary operation U, we have $UU^\dagger = \mathrm{I}$, where $U^\dagger$ is the Hermitian transpose of U and I is the identity operator.

Projector on the quantum system. We introduce a special operation called projector. For the state $|\psi\rangle$ of a quantum system defined over finite set X, a projector M_y ($y \in X$) applies the operation $|y\rangle\langle y|$ to $|\psi\rangle$ to get the new state $|y\rangle\langle y|\psi\rangle$. M_y can also be generalized to a new projector M_S ($S \in X$) which applies the operation $\sum_{y\in S} |y\rangle\langle y|$. We stress that any projector M is Hermitian (i.e., we have $\mathrm{M}^\dagger = \mathrm{M}$) and idempotent (i.e., we have $\mathrm{M}^2 = \mathrm{M}$).

Basic measurement. The state $|\psi\rangle$ in a quantum system can be measured in the computational basis. For example, suppose $|\psi\rangle = \sum_{x\in X} \alpha_x|x\rangle$ ($\sum_{x\in X} |\alpha_x|^2 = 1$) with computational basis $\{|x\rangle\}_{x\in X}$, if we measure $|\psi\rangle$ in basis $\{|x\rangle\}_{x\in X}$, we obtain the measurement result x with probability $|\langle x|\psi\rangle|^2 = |\alpha_x|^2$. Given that result x obtained, the state $|\psi\rangle$ collapses to the state $|x\rangle$. Moreover, we explain an important special class of measurements known as projective measurement. Indeed, a projective measurement $\mathbb{M}$ is defined by a set of projectors $\mathrm{M}_1, \dots, \mathrm{M}_n$ where M_i are mutually orthogonal and $\sum_{i=1}^{n} \mathrm{M}_i = \mathrm{I}$. Upon measuring the state $|\psi\rangle$, the probability of obtaining measurement result i ($i \in \{1, \dots, n\}$) is $\langle\psi|\mathrm{M}_i|\psi\rangle$ ($= \|\mathrm{M}_i|\psi\rangle\|^2$). Given that result i obtained, the state $|\psi\rangle$ collapses to the state $\mathrm{M}_i|\psi\rangle/\sqrt{\langle\psi|\mathrm{M}_i|\psi\rangle}$.

Quantum oracle algorithm and parallel query. A quantum oracle algorithm $\mathcal{A}$ is an algorithm that has quantum access to oracles, and that can perform a mix of classical and quantum unitary algorithms. In this paper, without loss of generality, we default that $\mathcal{A}$'s final output is a single bit $b \in \{0,1\}$.

Following [1], we allow the quantum oracle algorithm $\mathcal{A}^H(z)$ to make parallel queries to its oracle H, where z is the input of $\mathcal{A}$. We say $\mathcal{A}$ makes parallel queries to H with query depth d and query width n if $\mathcal{A}$ satisfies the following property:

- $\mathcal{A}$ splits its queries into d bunches, each bunch contains n queries and are queried in parallel (counting this parallel queries as one invoking).

That is, $\mathcal{A}$ invokes H d times and queries H with times n in each invoking. It is also obvious to see that $\mathcal{A}$ makes in total $d \cdot n$ queries to oracle H if we count parallel queries as separate queries.

Unitary quantum oracle algorithm. We say a quantum oracle algorithm $\mathcal{A}^H(z)$ is a unitary quantum oracle algorithm if its entire execution can be described as:

$$\mathbb{M}_{\mathcal{A}} U_q O_H U_{q-1} O_H \dots U_2 O_H U_1 O_H |\psi_z\rangle.$$

Here $|\psi_z\rangle$ is an initial pure state that may depend on input z, $U_1, \dots, U_q$ are the fixed unitary operations applied between oracle queries, O_H is the unitary operation used to implement the oracle H. $\mathbb{M}_{\mathcal{A}} := \{\mathrm{M}_0^{\mathcal{A}}, \mathrm{M}_1^{\mathcal{A}}\}$ is the final binary projective measurement performed by $\mathcal{A}$ on its quantum register, and the measurement result of $\mathbb{M}_{\mathcal{A}}$ (0 or 1) is the final output of $\mathcal{A}$.

In addition, We say a unitary quantum oracle algorithm $\mathcal{A}^H(z)$ makes parallel queries to H with query depth d and query width n if its entire execution can be described as:

$$\mathbb{M}_{\mathcal{A}} U_d (O_H)^{\otimes n} U_{d-1} (O_H)^{\otimes n} \dots U_2 (O_H)^{\otimes n} U_1 (O_H)^{\otimes n} |\psi_z\rangle.$$

Compared with the non-parallel case, the only two differences are that the unitary operation $(O_H)^{\otimes n}$ is used to implement the n-width parallel quantum oracle for H, and the fixed unitary operations applied between the oracle queries is replaced into $U_1, \dots, U_d$. Here $(O_H)^{\otimes n}$ maps the oracle's input state of n pairs of input/output registers $|in_1\rangle|out_1\rangle \cdots |in_n\rangle|out_n\rangle$ to the state $|in_1\rangle|out_1 \oplus H(in_1)\rangle \cdots |in_n\rangle|out_n \oplus H(in_n)\rangle$.

Next, we introduce the following well-known fact about the quantum oracle algorithm.

Fact 1. *Any quantum oracle algorithm can be transformed into a unitary quantum oracle algorithm with constant factor computational overhead and the same query times and query depth/width.*

The above fact implies that, without loss of generality, we can directly assume the quantum oracle algorithms to be unitary.

A.2 Cryptographic Primitives and Security Definitions

Definition 1 (Public key encryption). *A public key encryption (PKE) scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *consists of three polynomial time algorithms (in the security parameter* λ*) with a finite message space* $\mathcal{M}$ *such that*

1. Gen*, the key generation algorithm, is a probabilistic algorithm that on input* 1^λ *outputs a public/secret key pair* (pk, sk)*.*
2. Enc*, the encryption algorithm, on input* pk *and a message* $m \in \mathcal{M}$*, it chooses* $r \xleftarrow{\$} \mathcal{R}$ *(*$\mathcal{R}$ *is the randomness space), computes* $\mathsf{Enc}_{pk}(m; r) \in \mathcal{C}$ *(*$\mathcal{C}$ *is the ciphertext space) and outputs ciphertext* $c := \mathsf{Enc}_{pk}(m; r)$*. If* Enc *does not use randomness* r *to compute* c*,* Enc *is a deterministic algorithm and outputs* $c := \mathsf{Enc}_{pk}(m)$*.*
3. Dec*, the decryption algorithm, is a deterministic algorithm that on input ciphertext* $c \in \mathcal{C}$ *outputs a message* $m := \mathsf{Dec}_{sk}(c)$*, or a special symbol* $\perp \notin \mathcal{M}$ *to indicate that* c *is not a valid ciphertext.*

We say P *is a randomized PKE (rPKE) scheme if* Enc *uses randomness* r *to compute ciphertext* c*. Otherwise, we say* P *is a deterministic PKE (dPKE) scheme.*

Definition 2 (Correctness error [15]). *We say a randomized PKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *with message/randomness space* $\mathcal{M}/\mathcal{R}$ *is* δ*-correct if*

$$\mathop{\mathbb{E}}_{(pk,sk)\leftarrow\mathsf{Gen}}\left[\max_{m\in\mathcal{M}} \Pr\left[\mathsf{Dec}_{sk}(c) \neq m : c = \mathsf{Enc}_{pk}(m; r), r \xleftarrow{\$} \mathcal{R}\right]\right] \leq \delta.$$

Definition 3 (η**-injective [25]).** *Let* $\eta \geq 0$*. We say a deterministic PKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{dEnc}, \mathsf{dDec})$ *is* η*-injective if*

$$\Pr[\mathsf{dEnc}_{pk}(\cdot) \textit{ is not injective} : (pk, sk) \leftarrow \mathsf{Gen}, H \leftarrow \Omega_H] \leq \eta.$$

Here the $H \leftarrow \Omega_H$ *represents the random oracle that may be used by* P*.*

Definition 4 (Weakly γ**-spread [10]).** *We say a randomized PKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *with message/ciphertext space* $\mathcal{M}/\mathcal{C}$ *is weakly* γ*-spread if*

$$\mathop{\mathbb{E}}_{(pk,sk)\leftarrow\mathsf{Gen}}\left[\max_{m\in\mathcal{M},c\in\mathcal{C}} \Pr[c = \mathsf{Enc}_{pk}(m)]\right] \leq 2^{-\gamma},$$

where the probability is over the randomness of the encryption.

Definition 5 (Randomness recoverable [12]). *We say a randomized PKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *with message/randomness space* $\mathcal{M}/\mathcal{R}$ *is randomness recoverable if there exists a deterministic (recover) algorithm* Rec *such that for all* $m \in \mathcal{M}$ *and* $r \in \mathcal{R}$*,*

$$\Pr[\forall m' \in \mathsf{Preimg}(pk, c), \mathsf{Enc}_{pk}(m'; \mathsf{Rec}(pk, m', c)) \neq c \mid c := \mathsf{Enc}_{pk}(m; r)] = 0,$$

where $\mathsf{Preimg}(pk, c) := \{m \in \mathcal{M} \mid \exists r \in \mathcal{R} : \mathsf{Enc}_{pk}(m; r) = c\}$ *and the probability above is taken over* $(pk, sk) \leftarrow \mathsf{Gen}$*. Additionally, we require that* $\mathsf{Rec}(pk, m, c)$ *returns the symbol* $\perp \notin \mathcal{R}$ *if* $m \notin \mathsf{Preimg}(pk, c)$*.*

Definition 6 (Unique randomness recoverable). *We say a randomized PKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *with message/randomness space* $\mathcal{M}/\mathcal{R}$ *is unique randomness recoverable if there exists a deterministic (recover) algorithm* Rec *such that for all* $m \in \mathcal{M}$ *and* $r \in \mathcal{R}$,

$$\Pr[\mathsf{Rec}(pk, m, c) \neq r \mid c := \mathsf{Enc}_{pk}(m; r)] = 0,$$

where the probability is taken over $(pk, sk) \leftarrow \mathsf{Gen}$. *For the pair* (m, c), *if* $\nexists r \in \mathcal{R}$ *such that* $c = \mathsf{Enc}_{pk}(m; r)$, $\mathsf{Rec}(pk, m, c)$ *returns the symbol* $\perp \notin \mathcal{R}$.

Remark 7. Since Rec is a deterministic algorithm, a unique randomness recoverable PKE scheme must satisfy that, for any $(m, r) \in \mathcal{M} \times \mathcal{R}$, there does not exist $r' \neq r$ such that $\mathsf{Enc}_{pk}(m; r) = \mathsf{Enc}_{pk}(m; r')$. This is why we call this property as unique randomness recoverable.

Definition 7 (Security notions for PKE scheme). *For any polynomial time quantum adversary* $\mathcal{A}$, *we define its* $\mathsf{OW\text{-}CPA}$ *and* $\mathsf{IND\text{-}CPA}$ *advantage against PKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *as follows:*

$$\mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{A}) := \Pr[1 \leftarrow \mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{OW\text{-}CPA}}],$$
$$\mathrm{Adv}_{\mathsf{P}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) := \left|\Pr[1 \leftarrow \mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{IND\text{-}CPA}}] - \frac{1}{2}\right|.$$

Here games $\mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{OW\text{-}CPA}}$ *and* $\mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{IND\text{-}CPA}}$ *are shown in Fig. 4. We say scheme* P *is* $\mathsf{OW\text{-}CPA}$*-secure/*$\mathsf{IND\text{-}CPA}$*-secure, if for any polynomial time quantum adversary* $\mathcal{A}$, *the* $\mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{A})/\mathrm{Adv}_{\mathsf{P}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A})$ *is negligible.*

```
$\mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{OW\text{-}CPA}}$
$(pk, sk) \leftarrow \mathsf{Gen}$
$m^* \xleftarrow{\$} \mathcal{M}$
$c^* := \mathsf{Enc}_{pk}(m^*)$
$m' \leftarrow \mathcal{A}(pk, c^*)$
return $[\![m' = m^*]\!]$
```

```
$\mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{IND\text{-}CPA}}$
$(pk, sk) \leftarrow \mathsf{Gen}$
$b \xleftarrow{\$} \{0, 1\}$
$(m_0^*, m_1^*) \leftarrow \mathcal{A}(pk)$
$c^* := \mathsf{Enc}_{pk}(m_b^*)$
$b' \leftarrow \mathcal{A}(pk, c^*)$
return $[\![b' = b]\!]$
```

Fig. 4. Games $\mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{OW\text{-}CPA}}$ and $\mathbf{G}_{\mathcal{A},\mathsf{P}}^{\mathsf{IND\text{-}CPA}}$.

Definition 8 (Key encapsulation mechanism). *A key encapsulation mechanism* $\mathsf{KEM} = (\mathsf{Gen}, \mathsf{Enca}, \mathsf{Deca})$ *consists of three polynomial time algorithms with the key space* $\mathcal{K}$ *and the ciphertext space* $\mathcal{C}$ *such that*

1. Gen, *the key generation algorithm, is a probabilistic algorithm that on input* 1^λ *outputs a public/secret key pair* (pk, sk).

2. Enca, *the encapsulation algorithm, is a probabilistic algorithm that on input* pk *outputs a key* $k \in \mathcal{K}$ *and a ciphertext* $c \in \mathcal{C}$.
3. Deca, *the decapsulation algorithm, is a deterministic algorithm that on input ciphertext* c *outputs a key* $k := \mathsf{Deca}(sk, c)$ *or a special symbol* $\perp \notin \mathcal{K}$ *to indicate that* c *is not a valid ciphertext.*

Definition 9 (Security notions for KEM scheme). *For any polynomial time quantum adversary* $\mathcal{A}$, *we define its* IND-CPA *and* IND-CCA *advantage against KEM scheme* $\mathsf{KEM} = (\mathsf{Gen}, \mathsf{Enca}, \mathsf{Deca})$ *as follows:*

$$\mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A}) := \left|\Pr[1 \leftarrow \mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CPA}}] - \frac{1}{2}\right|,$$
$$\mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A}) := \left|\Pr[1 \leftarrow \mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}] - \frac{1}{2}\right|.$$

Here games $\mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CPA}}$ *and* $\mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}$ *are shown in Fig. 5. We say scheme* KEM *is* IND-CPA*-secure/*IND-CCA*-secure, if for any polynomial time quantum adversary* $\mathcal{A}$, *the advantage* $\mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CPA}}(\mathcal{A})/\mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A})$ *is negligible.*

```
$\mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CPA}}$
  $(pk, sk) \leftarrow \mathsf{Gen}$
  $b \xleftarrow{\$} \{0,1\}$
  $(k_0, c^*) \leftarrow \mathsf{Enca}(pk)$
  $k_1 \xleftarrow{\$} \mathcal{K}$
  $b' \leftarrow \mathcal{A}(pk, c^*, k_b)$
  return $[\![b' = b]\!]$
```

```
$\mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}$
  $(pk, sk) \leftarrow \mathsf{Gen}$
  $b \xleftarrow{\$} \{0,1\}$
  $(k_0, c^*) \leftarrow \mathsf{Enca}(pk)$
  $k_1 \xleftarrow{\$} \mathcal{K}$
  $b' \leftarrow \mathcal{A}^{\mathsf{oDeca}}(pk, c^*, k_b)$
  return $[\![b' = b]\!]$
```

```
$\mathsf{oDeca}(c)$
  if $c = c^*$
    return $\perp$
  else return $\mathsf{Deca}(sk, c)$
```

Fig. 5. Games $\mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CPA}}$ and $\mathbf{G}_{\mathcal{A},\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}$.

We also define OW-CPA/IND-CPA security of PKE schemes and IND-CPA/IND-CCA security of KEM schemes in the QROM, where the adversary $\mathcal{A}$ can make quantum queries to random oracles.

A.3 Proof of Lemma 3 and Lemma 4

Lemma 3 ([25, Lemma 3.1]). *For any states* $|\phi_1\rangle$ *and* $|\phi_2\rangle$, *we have*

$$\left|\||\phi_1\rangle\|^2 - \||\phi_2\rangle\|^2\right| \leq |(|\phi_1\rangle - |\phi_2\rangle, |\phi_1\rangle + |\phi_2\rangle)|.$$

Proof. We have

$$\begin{aligned}\left|\||\phi_1\rangle\|^2 - \||\phi_2\rangle\|^2\right| &= |(|\phi_1\rangle, |\phi_1\rangle) - (|\phi_2\rangle, |\phi_2\rangle)| \\ &\overset{(a)}{\leq} |(|\phi_1\rangle, |\phi_1\rangle) - (|\phi_2\rangle, |\phi_2\rangle) + (|\phi_1\rangle, |\phi_2\rangle) - (|\phi_2\rangle, |\phi_1\rangle)| \\ &= |(|\phi_1\rangle - |\phi_2\rangle, |\phi_1\rangle + |\phi_2\rangle)|.\end{aligned}$$

Here (a) follows from the fact that $(|\phi_1\rangle, |\phi_1\rangle), (|\phi_2\rangle, |\phi_2\rangle) \geq 0$ and $(|\phi_1\rangle, |\phi_2\rangle) - (|\phi_2\rangle, |\phi_1\rangle)$ is a complex number with real part 0. □

Lemma 4. *For any states $|\varphi_1\rangle, \ldots, |\varphi_n\rangle$ and $|\phi_1\rangle, \ldots, |\phi_n\rangle$, we have*

$$\sum_{i=1}^{n} |(|\varphi_i\rangle, |\phi_i\rangle)| \leq \sqrt{\sum_{i=1}^{n} \||\varphi_i\rangle\|^2} \cdot \sqrt{\sum_{i=1}^{n} \||\phi_i\rangle\|^2}.$$

Proof. We have

$$\sum_{i=1}^{n} |(|\varphi_i\rangle, |\phi_i\rangle)| \overset{(a)}{\leq} \sum_{i=1}^{n} \||\varphi_i\rangle\| \cdot \||\phi_i\rangle\| \overset{(b)}{\leq} \sqrt{\sum_{i=1}^{n} \||\varphi_i\rangle\|^2} \cdot \sqrt{\sum_{i=1}^{n} \||\phi_i\rangle\|^2}.$$

Here (a) and (b) both use the Cauchy-Schwarz inequality.

A.4 Proof of Lemma 6

Lemma 6. *Recall that 1_S is the indicator function of the set S, that is, $1_S(x) = 1$ if $x \in S$ and 0 otherwise. Let $\chi_0 := \mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}}$ and $\chi_1 := \mathrm{M}_{S^{\oplus n}}$. Then, the binary projective measurement $\mathbb{M}_{S^{\oplus n}} := \{\chi_0, \chi_1\}$ on the oracle's input register IN can be implemented by making two parallel queries to 1_S with query width n.*

Proof. We first define an operation $\mathcal{E}^{1_S}$ that acts on the register IN as:

1. Initialize a single qubit register L with $|0\rangle$.
2. Apply unitary operation $U_{S^{\oplus n}}$ on registers IN and L, where

$$U_{S^{\oplus n}} : |in_1\rangle \cdots |in_n\rangle|l\rangle \mapsto (\chi_1|in_1\rangle \cdots |in_n\rangle)|l \oplus 1\rangle + (\chi_0|in_1\rangle \cdots |in_n\rangle)|l\rangle.$$

3. Measure L in the computational basis and output the measurement result.

Note that $\chi_0 = \mathrm{I}_{IN} - \mathrm{M}_{S^{\oplus n}}$ and $\chi_1 = \mathrm{M}_{S^{\oplus n}}$. By the definition of $\mathrm{M}_{S^{\oplus n}}$ given in Eq. (10), it is obvious that $U_{S^{\oplus n}}$ can be efficiently implemented by making two parallel queries to 1_S with input register IN. Thus, $\mathcal{E}^{1_S}$ can be efficiently implemented by making two parallel queries to 1_S with query width n.

Indeed, if the (unit) state of register IN is $|\phi\rangle$, the detailed quantum state transformation process when we apply $\mathcal{E}^{1_S}$ on register IN can be described as:

1. Initialize a single qubit register L with $|0\rangle$. At this point we get state $|\phi\rangle|0\rangle$.
2. Based on the state $|\phi\rangle|0\rangle$, apply unitary operation $U_{S^{\oplus n}}$ on registers IN and L. After this application, we get the state

$$U_{S^{\oplus n}}|\phi\rangle|0\rangle = \chi_1|\phi\rangle|1\rangle + \chi_0|\phi\rangle|0\rangle.$$

3. Measure L in the computational basis and output the measurement result. Obviously, this measurement has result 0 (resp. 1) with probability $\|\chi_0|\phi\rangle\|^2$ (resp. $\|\chi_1|\phi\rangle\|^2$), and next the state of register IN will collapse into $\chi_0|\phi\rangle/\|\chi_0|\phi\rangle\|$ (resp. $\chi_1|\phi\rangle/\|\chi_1|\phi\rangle\|$).

In other words, if the (unit) state of register IN is $|\phi\rangle$, $\mathcal{E}^{1_S}$ outputs the measurement result 0 (resp. 1) with probability $\|\chi_0|\phi\rangle\|^2$ (resp. $\|\chi_1|\phi\rangle\|^2$), and the state of register IN after $\mathcal{E}^{1_S}$ outputs this result will collapse into $\chi_0|\phi\rangle/\|\chi_0|\phi\rangle\|$ (resp. $\chi_1|\phi\rangle/\|\chi_1|\phi\rangle\|$). This means that we can implement the binary projective measurement $\mathbb{M}_{S^{\oplus n}} := \{\chi_0, \chi_1\}$ on the register IN by performing $\mathcal{E}^{1_S}$. □

A.5 Proof of Lemma 8

Before proving Lemma 8, we first introduce the following two lemmas that will be used later.

Lemma A1 (Double-Sided O2H [4, Lemma 5]) *Let $G, H : X \to Y$ be random functions, $S := \{w^*\}$ ($w^* \in X$) be a random set and $z \in Z$ be a random bitstring. The tuple (G, H, S, z) may have arbitrary joint distribution D and satisfies that $\forall x \notin S$, $G(x) = H(x)$. Let $\mathcal{A}^O$ ($O \in \{G, H\}$) be a quantum oracle algorithm that makes parallel queries with query depth d and query width n. Let Ev be an arbitrary classical event. Define*

$$P_{\text{left}} := \Pr_{(G,H,S,z)\leftarrow D}[\mathsf{Ev} : \mathcal{A}^H(z)],\ P_{\text{right}} := \Pr_{(G,H,S,z)\leftarrow D}[\mathsf{Ev} : \mathcal{A}^G(z)].$$

Then, we can construct an algorithm $\mathcal{B}^{G,H}(z)$ which has the following two properties:

- *Let $\mathrm{Adv}(\mathcal{B}) := \Pr[w = w^* : w \leftarrow \mathcal{B}^{G,H}(z), (G, H, S, z) \leftarrow D]$, then*

$$|P_{\text{left}} - P_{\text{right}}| \leq 2\sqrt{\mathrm{Adv}(\mathcal{B})},\ \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \leq 2\sqrt{\mathrm{Adv}(\mathcal{B})}.$$

- *$\mathcal{B}^{G,H}(z)$ makes parallel queries to G and H both with query depth d and query width n. Its running time can be bounded as $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}}$.*

Lemma A2 *Let $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a δ-correct rPKE scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$, and assume P is unique randomness recoverable. Then we have*

$$\Pr\left[\exists (m \neq m^*, r) \in \mathcal{M} \times \mathcal{R},\ \mathsf{Enc}_{pk}(m; r) = \mathsf{Enc}_{pk}(m^*; r^*)\right] \leq 2 \cdot \delta,$$

where the probability is taken over $(pk, sk) \leftarrow \mathsf{Gen}$, $m^* \xleftarrow{\$} \mathcal{M}$ and $r^* \xleftarrow{\$} \mathcal{R}$.

Proof. The proof of this lemma is similar with the proof of [26, Lemma 4]. After proving Lemma 8, we will give a detailed proof of this lemma. □

Now, we prove our Lemma 8.

Lemma 8 (Security of T from $\mathsf{OW\text{-}CPA}$). *Let $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a δ-correct rPKE scheme, and assume P is unique randomness recoverable with the recover algorithm Rec. Let $\mathcal{A}$ be a $\mathsf{OW\text{-}CPA}$ adversary against $\mathsf{T}[\mathsf{P}, H]$, making* *parallel quantum queries to the random oracle H with query depth d_H and query width n. Let $q_H := d_H \cdot n$.*

Then, we can construct a OW-CPA *adversary $\mathcal{B}$ against* P *such that*

$$\mathsf{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{T}[\mathsf{P},H]}(\mathcal{A}) \leq 10 \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + 16 \cdot \delta$$

and $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}} + O(q_H^2) + O(q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$.

Proof. Let Ω_H be the set of all functions $H : \mathcal{M} \to \mathcal{R}$, where $\mathcal{M}/\mathcal{R}$ is the message/randomness space of P. We introduce two games as shown in Fig. 6.

```
GAMES $\mathbf{G_0}$-$\mathbf{G_1}$
  $(pk, sk) \leftarrow \mathsf{Gen}, H_1 \xleftarrow{\$} \Omega_H$   $//\mathbf{G_0}$-$\mathbf{G_1}$
  $m^* \xleftarrow{\$} \mathcal{M}, r \xleftarrow{\$} \mathcal{R}$   $//\mathbf{G_0}$-$\mathbf{G_1}$
  $c^* := \mathsf{Enc}_{pk}(m^*; H(m^*))$   $//\mathbf{G_0}$-$\mathbf{G_1}$
  $m' \leftarrow \mathcal{A}^H(pk, c^*)$   $//\mathbf{G_0}$
  $m' \leftarrow \mathcal{A}^G(pk, c^*)$   $//\mathbf{G_1}$
  return $[\![m' = m^*]\!]$   $//\mathbf{G_0}$-$\mathbf{G_1}$

$H(m)$ $//\mathbf{G_0}$-$\mathbf{G_1}$
  return $H_1(m)$

$G(m)$ $//\mathbf{G_1}$
  if $m = m^*$
    return $r$
  else return $H_1(m)$
```

Fig. 6. Games $\mathbf{G_0}$-$\mathbf{G_1}$ for the proof of Lemma 8. Note that the oracles G, H can both be quantum accessed in those games. In this figure, we just write the classical descriptions of G, H for brevity.

Game $\mathbf{G_0}$: This is the OW-CPA game of $\mathsf{T}[\mathsf{P},H]$ with adversary $\mathcal{A}$ in the QROM.

$$\mathsf{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{T}[\mathsf{P},H]}(\mathcal{A}) = \Pr[1 \leftarrow \mathbf{G_0}]. \tag{21}$$

Game $\mathbf{G_1}$: In this game, the oracle H is changed into G, where G is identical with H except that $G(m^*)$ is a fresh random value uniformly sampled from $\mathcal{R}$.

Let D be a joint distribution of the tuple $(G, H, \{m^*\}, pk, c^*)$, where $H = H_1$ ($H_1 \xleftarrow{\$} \Omega_H$), G is identical with H except that $G(m^*)$ is a fresh random value uniformly sampled from $\mathcal{R}$, $m^* \xleftarrow{\$} \mathcal{M}$, pk is sampled by $(pk, sk) \leftarrow \mathsf{Gen}$ and $c^* := \mathsf{Enc}_{pk}(m^*; H(m^*))$. By using Lemma A1 with the joint distribution D, we can construct an algorithm $\mathcal{A}_1$ such that

$$\begin{aligned} &\left|\sqrt{\Pr[1 \leftarrow \mathbf{G_0}]} - \sqrt{\Pr[1 \leftarrow \mathbf{G_1}]}\right| \\ &\leq 2\sqrt{\Pr_{(G,H,\{m^*\},pk,c^*)\leftarrow D}\left[m' = m^* : m' \leftarrow \mathcal{A}_1^{G,H}(pk, c^*)\right]}. \end{aligned} \tag{22}$$

Here $\mathcal{A}_1$ makes parallel quantum queries to G and H both with query depth d_H and query width n. The running time of $\mathcal{A}_1$ is $\mathcal{T}_{\mathcal{A}_1} \approx \mathcal{T}_{\mathcal{A}}$.

Now, by using the recover algorithm Rec of P, we construct a $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}_1$ against P as:

1. $\mathcal{B}_1$ gets the challenge ciphertext $c^* := \mathsf{Enc}_{pk}(m^*; r^*)$ and public key pk, where m^* is the challenge plaintext uniformly sampled from $\mathcal{M}$ by the challenger, $r^* \overset{\$}{\leftarrow} \mathcal{R}$ and pk is sampled by $(pk, sk) \leftarrow \mathsf{Gen}$.
2. $\mathcal{B}_1$ chooses $r \overset{\$}{\leftarrow} \mathcal{R}$ and a $2q_H$-wise independent function $H_1 : \mathcal{M} \to \mathcal{R}$.
3. $\mathcal{B}_1$ runs the algorithm $\mathcal{A}_1$ and simulates oracles G and H for $\mathcal{A}_1$ as follows[10].
   (a) Simulation of G: If the input m satisfies $\mathsf{Enc}_{pk}(m; \mathsf{Rec}(pk, m, c^*)) = c^*$, the r will be returned. Otherwise, $H_1(m)$ is returned.
   (b) Simulation of H: If the input m satisfies $\mathsf{Enc}_{pk}(m; \mathsf{Rec}(pk, m, c^*)) = c^*$, the $\mathsf{Rec}(pk, m, c^*)$ will be returned. Otherwise, $H_1(m)$ is returned.
4. $\mathcal{B}_1$ finally outputs $\mathcal{A}_1$'s output.

Since $\mathcal{A}_1$'s running time is almost the same as $\mathcal{A}$, one can easily check that the running time of $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}_1$ is $\mathcal{T}_{\mathcal{B}_1} \approx \mathcal{T}_{\mathcal{A}} + O(q_H^2) + O(q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$.

By the definition of unique randomness recoverable given in Definition 6, it is obvious that $\mathsf{Rec}(pk, m^*, c^*) = r^*$, and hence $\mathsf{Enc}_{pk}(m^*; \mathsf{Rec}(pk, m^*, c^*)) = c^*$. For the challenge plaintext m^* of the $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}_1$ defined above, let Bad be the following event:

$$\mathsf{Bad}\text{: } \exists m \in \mathcal{M},\ r \in \mathcal{R} \text{ s.t. } m \neq m^* \text{ and } \mathsf{Enc}_{pk}(m; r) = c^* \ (c^* = \mathsf{Enc}_{pk}(m^*; r^*)).$$

If the event Bad does not occur, we can conclude that $\mathsf{Enc}_{pk}(x; \mathsf{Rec}(pk, x, c^*)) = c^*$ iff $x = m^*$ since $\mathsf{Enc}_{pk}(m^*; \mathsf{Rec}(pk, m^*, c^*)) = c^*$. That is to say, if the event Bad does not occur, the $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}_1$ defined above perfectly[11] simulates G and H for $\mathcal{A}_1$. Thus,

$$\Pr_{(G,H,\{m^*\},pk,c^*)\leftarrow D}\left[m' = m^* : m' \leftarrow \mathcal{A}_1^{G,H}(pk, c^*)\right] \leq \mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{B}_1) + \Pr[\mathsf{Bad}]. \tag{23}$$

Since P is unique randomness recoverable and δ-correct, by using Lemma A2, it is easy to see that

$$\Pr[\mathsf{Bad}] \leq 2 \cdot \delta. \tag{24}$$

Combining Eq. (21) to Eq. (24), we obtain

$$\begin{aligned}\mathsf{Adv}_{\mathsf{T}[\mathsf{P},H]}^{\mathsf{OW\text{-}CPA}}(\mathcal{A}) &\leq \left(\sqrt{\Pr[1 \leftarrow \mathbf{G_1}]} + 2\sqrt{\mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{B}_1) + 2 \cdot \delta}\right)^2 \\ &\leq 2 \cdot \Pr[1 \leftarrow \mathbf{G_1}] + 8 \cdot \mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{B}_1) + 16 \cdot \delta.\end{aligned}$$

[10] Note that $\mathcal{A}_1$ actually makes parallel quantum queries to G and H. Here we just explain their simulations in the classical manner for brevity.

[11] Here the "perfectly" means that, in $\mathcal{A}_1$'s view, the oracle G (resp. H) simulated by $\mathcal{B}_1$ is perfectly indistinguishable with the oracles G (resp. H) obtained by sampling $(G, H, \{m^*\}, pk, c^*)$ according to the distribution D.

For the $\Pr[1 \leftarrow \mathbf{G_1}]$, since the oracle G in game $\mathbf{G_1}$ is independent with the $c^* = \mathsf{Enc}_{pk}(m^*; H(m^*))$, one can easily construct a $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}_2$ against P by running $\mathcal{A}$ such that

$$\Pr[1 \leftarrow \mathbf{G_1}] = \mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{B}_2)$$

and $\mathcal{T}_{\mathcal{B}_2} \approx \mathcal{T}_{\mathcal{A}} + O(q_H^2)$. Combine the above two equations and fold $\mathcal{B}_1$ and $\mathcal{B}_2$ into one single $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}$ against P yields

$$\mathsf{Adv}_{\mathsf{T}[\mathsf{P},H]}^{\mathsf{OW\text{-}CPA}}(\mathcal{A}) \leq 10 \cdot \mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{B}) + 16 \cdot \delta.$$

Here $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}} + O(q_H^2) + O(q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$. □

Proof of Lemma A2

Proof. For fixed (pk, sk), we define the two following subsets of $\mathcal{M} \times \mathcal{R}$:

$$S_{pk,sk}^{coll} := \{(m, r) \mid \exists (m' \neq m, r') \in \mathcal{M} \times \mathcal{R},\ \mathsf{Enc}_{pk}(m'; r') = \mathsf{Enc}_{pk}(m; r)\},$$
$$S_{pk,sk}^{error} := \{(m, r) \mid \mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m; r)) \neq m\}.$$

According to the definition of δ-correct, we have

$$\Pr[(m, r) \in S_{pk,sk}^{error} \mid (pk, sk) \leftarrow \mathsf{Gen}, m \xleftarrow{\$} \mathcal{M}, r \xleftarrow{\$} \mathcal{R}] \leq \delta. \tag{25}$$

Denote the ciphertext space of P as $\mathcal{C}$. Then for $c \in \mathcal{C}$, define the set

$$S_{pk,sk,c}^{coll} := \{(m, r) \mid \exists (m' \neq m, r') \in \mathcal{M} \times \mathcal{R},\ \mathsf{Enc}_{pk}(m'; r') = \mathsf{Enc}_{pk}(m; r) = c\}.$$

By the definition of $S_{pk,sk}^{coll}$ given above, there must exist a subset $C \subseteq \mathcal{C}$ such that $S_{pk,sk}^{coll} = \bigcup_{c \in C} S_{pk,sk,c}^{coll}$. That is, we separate $S_{pk,sk}^{coll}$ into several disjoint sets through the corresponding encryption value of $(m, r) \in S_{pk,sk}^{coll}$.

By the definition of unique randomness recoverable given in Definition 6 and the next Remark 7, one can easily check that for any $m \in \mathcal{M}$, $r \in \mathcal{R}$ and (pk, sk) sampled by Gen, there does not exist $r' \in \mathcal{R}$ such that $r' \neq r$ and $\mathsf{Enc}_{pk}(m; r) = \mathsf{Enc}_{pk}(m; r')$. Based on this property, for the set $S_{pk,sk,c}^{coll}$ with $c \in C$, we can conclude that there exists at most one pair in $S_{pk,sk,c}^{coll}$ which does not belong to $S_{pk,sk}^{error}$ since the decryption algorithm Dec must be deterministic. That is to say, the other pairs in $S_{pk,sk,c}^{coll}$ must belong to $S_{pk,sk,c}^{coll} \cap S_{pk,sk}^{error}$. Note that $|S_{pk,sk,c}^{coll}| \geq 2$ for any $c \in C$, hence we can obtain

$$\left|S_{pk,sk,c}^{coll} \backslash S_{pk,sk}^{error}\right| \leq \left|S_{pk,sk,c}^{coll} \cap S_{pk,sk}^{error}\right|.$$

Since $S_{pk,sk}^{coll} = \bigcup_{c \in C} S_{pk,sk,c}^{coll}$ and $S_{pk,sk,c}^{coll}$ are disjoint for different $c \in C$, we have

$$\left|S_{pk,sk}^{coll} \backslash S_{pk,sk}^{error}\right| \leq \left|S_{pk,sk}^{coll} \cap S_{pk,sk}^{error}\right|. \tag{26}$$

Then, for $(pk, sk) \leftarrow \mathsf{Gen}$, $m^* \xleftarrow{\$} \mathcal{M}$, and $r^* \xleftarrow{\$} \mathcal{R}$, we can compute

$$\begin{aligned}
&\Pr\left[\exists (m \neq m^*, r) \in \mathcal{M} \times \mathcal{R},\ \mathsf{Enc}_{pk}(m; r) = \mathsf{Enc}_{pk}(m; r^*)\right] \\
&= \Pr[(m^*, r^*) \in S^{coll}_{pk,sk}] \\
&= \Pr[(m^*, r^*) \in S^{coll}_{pk,sk} \backslash S^{error}_{pk,sk}] + \Pr[(m^*, r^*) \in S^{coll}_{pk,sk} \cap S^{error}_{pk,sk}] \\
&\overset{(a)}{\leq} 2 \cdot \Pr[(m^*, r^*) \in S^{coll}_{pk,sk} \cap S^{error}_{pk,sk}] \\
&\leq 2 \cdot \Pr[(m^*, r^*) \in S^{error}_{pk,sk}] \overset{(b)}{\leq} 2 \cdot \delta.
\end{aligned}$$

Here (a) and (b) use Eq. (26) and Eq. (25), respectively. □

A.6 Proof of Theorem 5

Theorem 5 **(IND-CPA of $\mathsf{KEM}^{\not\perp} \overset{\text{QROM}}{\Rightarrow}$ IND-CCA of $\mathsf{KEM}^{\not\perp}$).** *Let rPKE scheme* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be δ-correct. Let $\mathcal{A}$ be an* IND-CCA *adversary against* $\mathsf{KEM}^{\not\perp}$ $=\mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}]$*, making q_D classical queries to the decapsulation oracle, making parallel quantum queries to the random oracle H (resp. G) with query depth d_H (resp. d_G) and query width n. Let $q_H := d_H \cdot n$ and $q_G := d_G \cdot n$.*

Then, we can construct the following two adversaries:

- *A* PRF*-adversary $\mathcal{B}_1$ against* F *making at most q_D classical queries. The running time of $\mathcal{B}_1$ is $\mathcal{T}_{\mathcal{B}_1} \lesssim \mathcal{T}_{\mathcal{A}} + q_D \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Dec}}) + O(q_H^2 + q_G^2)$.*
- *An* IND-CPA *adversary $\mathcal{B}_2$ against $\mathsf{KEM}^{\not\perp}$ in the QROM. $\mathcal{B}_2$ makes parallel quantum queries to the random oracle H (resp. G) with query depth at most $d_H + d_G$ (resp. d_G) and query width n. The running time of $\mathcal{B}_2$ is $\mathcal{T}_{\mathcal{B}_2} \lesssim \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}} + O(q_G^2 + q_D^2)$.*

Adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ satisfy the following:

$$\mathrm{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq \mathrm{Adv}^{\mathsf{PRF}}_{\mathsf{F}}(\mathcal{B}_1) + \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{B}_2) + 16(2q_H + 2q_G + 1)^2 \cdot \delta.$$

Proof. For a fixed (pk, sk) pair sampled by Gen and $m \in \mathcal{M}$, we define sets

$$\mathcal{R}_{\text{bad}}(pk, sk, m) := \{r \in \mathcal{R} : \mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m; r)) \neq m\} \qquad (27)$$

and

$$\mathcal{R}_{\text{good}}(pk, sk, m) := \mathcal{R} \backslash \mathcal{R}_{\text{bad}}(pk, sk, m). \qquad (28)$$

Here $\mathcal{M}/\mathcal{R}$ is the message/randomness space of P. Then, we define two values

$$\delta(pk, sk, m) := \frac{|\mathcal{R}_{\text{bad}}(pk, sk, m)|}{|\mathcal{R}|} \text{ and } \delta(pk, sk) := \max_{m \in \mathcal{M}} \delta(pk, sk, m). \qquad (29)$$

By the definition of δ-correct given in Definition 2, we have $\mathbb{E}_{(pk,sk)\leftarrow\mathsf{Gen}}[\delta(pk, sk)] \leq \delta$ since P is δ-correct.

Let Ω_H, Ω_G, Ω_R be the set of all functions $H : \mathcal{M} \to \mathcal{R}$, $G : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$, $R : \mathcal{C} \to \mathcal{K}$, respectively. Here $\mathcal{C}$ is the ciphertext space of P, $\mathcal{K}$ is the key space of $\mathsf{KEM}^{\not\perp}$. Let $H_{\text{good}} : \mathcal{M} \rightarrow \mathcal{R}$ be a random function such that $H_{\text{good}}(m)$ is uniformly sampled from $\mathcal{R}_{\text{good}}(pk, sk, m)$ for any $m \in \mathcal{M}$. Now we introduce a sequence of hybrid games as shown in Fig. 7.

Game $\mathbf{G_0}$: This is the IND-CCA game of $\mathsf{KEM}^{\not\perp}$ with adversary $\mathcal{A}$ in the QROM.

$$\mathrm{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) = \left|\Pr[1 \leftarrow \mathbf{G_0}] - \frac{1}{2}\right|. \tag{30}$$

```
GAMES $\mathbf{G_0}$-$\mathbf{G_8}$
  $(pk, sk) \leftarrow \mathsf{Gen}$                                  $//\mathbf{G_0}$-$\mathbf{G_8}$
  $s \xleftarrow{\$} \mathcal{K}^{prf}$                                    $//\mathbf{G_0}$
  $sk' := (sk, s)$                                  $//\mathbf{G_0}$
  $H \xleftarrow{\$} \Omega_H$                                      $//\mathbf{G_0}$-$\mathbf{G_8}$
  $G_1 \xleftarrow{\$} \Omega_G$                                     $//\mathbf{G_0}$-$\mathbf{G_8}$
  $R \xleftarrow{\$} \Omega_R$                                      $//\mathbf{G_1}$-$\mathbf{G_8}$
  $H := H_{\text{good}}$                                   $//\mathbf{G_2}$-$\mathbf{G_6}$
  $R_1 \xleftarrow{\$} \Omega_R$                                     $//\mathbf{G_3}$-$\mathbf{G_8}$
  $G_2 \xleftarrow{\$} \Omega_G$                                     $//\mathbf{G_5}$-$\mathbf{G_8}$
  $m^* \xleftarrow{\$} \mathcal{M}, b \xleftarrow{\$} \{0,1\}$                       $//\mathbf{G_0}$-$\mathbf{G_8}$
  $c^* := \mathsf{Enc}_{pk}(m^*; H(m^*))$                  $//\mathbf{G_0}$-$\mathbf{G_8}$
  $k_0^* := G(m^*, c^*), k_1^* \xleftarrow{\$} \mathcal{K}$                  $//\mathbf{G_0}$-$\mathbf{G_8}$
  $b' \leftarrow \mathcal{A}^{H,G,\mathsf{Deca}}(pk, c^*, k_b^*)$              $//\mathbf{G_0}$-$\mathbf{G_7}$
  $b' \leftarrow \mathcal{B}^{H,G}(pk, c^*, k_b^*)$                     $//\mathbf{G_8}$
  return $[\![b' = b]\!]$                           $//\mathbf{G_0}$-$\mathbf{G_8}$
```

```
$G(m, c)$            $//\mathbf{G_0}$-$\mathbf{G_2}$,$\mathbf{G_8}$
  return $G_1(m, c)$ $//\mathbf{G_0}$-$\mathbf{G_2}$
  return $G_2(m, c)$     $//\mathbf{G_8}$
```

```
$G(m, c)$            $//\mathbf{G_3}$-$\mathbf{G_4}$
  if $\mathsf{Enc}_{pk}(m; H(m)) = c$
    return $R_1(c)$
  else return $G_1(m, c)$
```

```
$G(m, c)$                                  $//\mathbf{G_5}$-$\mathbf{G_7}$
  if $m = m^* \wedge c = c^*$                           $//\mathbf{G_5}$
  if $\mathsf{Enc}_{pk}(m; H(m)) = c^* \wedge c = c^*$ $//\mathbf{G_6} - \mathbf{G_7}$
    return $G_2(m, c)$
  else if $\mathsf{Enc}_{pk}(m; H(m)) = c$
    return $R_1(m, c)$
  else return $G_1(m, c)$
```

```
$\mathsf{Deca}(c \neq c^*)$                          $//\mathbf{G_0}$-$\mathbf{G_3}$
  parse $sk' = (sk, s)$
  $m' := \mathsf{Dec}_{sk}(c)$
  if $m' \neq \perp \wedge \mathsf{Enc}_{pk}(m'; H(m')) = c$
    return $K := G(m', c)$
  else return $K := \mathsf{F}(s, c)$                  $//\mathbf{G_0}$
  else return $K := R(c)$                  $//\mathbf{G_1}$-$\mathbf{G_3}$
```

```
$\mathsf{Deca}(c \neq c^*)$ $//\mathbf{G_4}$-$\mathbf{G_8}$
  return $K := R_1(c)$
```

Fig. 7. Games $\mathbf{G_0}$-$\mathbf{G_8}$ for the proof of Theorem 5. Note that the oracle G can be parallel quantum accessed in those games. In this figure, we just write the classical description of G for brevity.

Game $\mathbf{G_1}$ (PRF is random): In this game, the PRF F used in the decapsulation oracle Deca is changed into a random function $R \xleftarrow{\$} \Omega_R$.

We construct a PRF adversary $\mathcal{B}_1^{O}$ against F as:

1. By using a $2q_H$-wise (resp. $2q_G$-wise) independent function to simulate H (resp. G), $\mathcal{B}_1$ runs the adversary $\mathcal{A}$ in game $\mathbf{G_0}$. If $\mathcal{A}$ makes a query c to Deca that failed to decapsulate, $\mathcal{B}_1$ returns $O(c)$ to $\mathcal{A}$ by querying O with input c. $\mathcal{B}_1$ finally outputs 1 if game $\mathbf{G_0}$ returns 1 and 0 otherwise.

Here O can be $\mathsf{F}(s, \cdot)$ with $s \xleftarrow{\$} \mathcal{K}_{prf}$ or a uniformly random function $R \xleftarrow{\$} \Omega_R$. Since the PRF F is only used by Deca in game $\mathbf{G_0}$, $\mathcal{B}_1$ makes at most q_D classical queries to its oracle O and its running time can be bounded as $\mathcal{T}_{\mathcal{B}_1} \approx \mathcal{T}_{\mathcal{A}} + q_D \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Dec}}) + O(q_H^2 + q_G^2)$.

According to the above definition of adversary $\mathcal{B}_1$, it is easy to check that $\Pr[1 \leftarrow \mathcal{B}_1^{\mathsf{F}(s,\cdot)}] = \Pr[1 \leftarrow \mathbf{G_0}]$ and $\Pr[1 \leftarrow \mathcal{B}_1^{R(\cdot)}] = \Pr[1 \leftarrow \mathbf{G_1}]$. Thus

$$|\Pr[1 \leftarrow \mathbf{G_0}] - \Pr[1 \leftarrow \mathbf{G_1}]| = \mathrm{Adv}_{\mathsf{F}}^{\mathsf{PRF}}(\mathcal{B}_1). \tag{31}$$

Game $\mathbf{G_2}$: This game is the same as game $\mathbf{G_1}$, except that H is replaced by H_{good}. Recall that $H_{\text{good}} : \mathcal{M} \to \mathcal{R}$ is a random function such that $H_{\text{good}}(m)$ is uniformly sampled from $\mathcal{R}_{\text{good}}(pk, sk, m)$ for any $m \in \mathcal{M}$.

For $H \xleftarrow{\$} \Omega_H$ and a fixed (pk, sk), we construct the following algorithm $\mathcal{B}^H(pk, sk)$ with quantum oracle access to H that plays game $\mathbf{G_1}$:

1. $\mathcal{B}^H(pk, sk)$ simulates G (resp. R) by using a $2q_G$-wise (resp. $2q_D$-wise) independent function, replaces the calls to H in game $\mathbf{G_1}$ by calls to its oracle and simulates the decapsulation oracle Deca by using (pk, sk). $\mathcal{B}^H(pk, sk)$ finally outputs 1 if game $\mathbf{G_1}$ returns 1 and 0 otherwise.

Obviously, $\mathcal{B}^H(pk, sk)$ makes q_H quantum queries to its oracle H and

$$\Pr\left[1 \leftarrow \mathbf{G_1} : (pk, sk)\right] = \Pr\left[1 \leftarrow \mathcal{B}^H(pk, sk) : H \xleftarrow{\$} \Omega_H\right]. \tag{32}$$

Since game $\mathbf{G_2}$ is the same as game $\mathbf{G_1}$ except that H is changed into H_{good}, we have

$$\Pr\left[1 \leftarrow \mathbf{G_2} : (pk, sk)\right] = \Pr\left[1 \leftarrow \mathcal{B}^{H_{\text{good}}}(pk, sk)\right]. \tag{33}$$

Let $N_1 : \mathcal{M} \to \{0, 1\}$ be a random function such that $\Pr[N_1(m) = 1] = \delta(pk, sk, m)$ and $\Pr\left[N_1(m) = 0\right] = 1 - \delta(pk, sk, m)$ for any $m \in \mathcal{M}$. Here the value $\delta(pk, sk, m)$ is defined in Eq. (29). Let $N_2 : \mathcal{M} \to \{0, 1\}$ be a constant function such that $N_2(m) = 0$ for any $m \in \mathcal{M}$.

Then, by running $\mathcal{B}^H(pk, sk)$, we construct a quantum oracle algorithm $\mathcal{C}^N(pk, sk)$ $(N \in \{N_1, N_2\})$ as shown in Fig. 8. Here $\mathcal{C}^N(pk, sk)$ introduces a new quantum oracle K relates to its quantum oracle N and replaces H into K. In Supplementary Material A.7, we show that $\mathcal{C}^N(pk, sk)$ answers one query to K by querying N twice[12]. That is to say, our construction of $\mathcal{C}^N(pk, sk)$ shown in Fig. 8 is well-defined and $\mathcal{C}^N(pk, sk)$ makes $2q_H$ quantum queries to N.

[12] We actually construct a quantum circuit that answers one query to K by querying N twice in Supplementary Material A.7.

```
$\mathcal{C}^N(pk, sk)$
    $H \xleftarrow{\$} \Omega_H$
    $H := K$
    $b \leftarrow \mathcal{B}^H(pk, sk)$
    return $b$
```

```
$K(m)$
    if $N(m) = 0$
        return $r \xleftarrow{\$} \mathcal{R}_{\text{good}}(pk, sk, m)$
    if $N(m) = 1$
        return $r \xleftarrow{\$} \mathcal{R}_{\text{bad}}(pk, sk, m)$
```

Fig. 8. Quantum oracle algorithm $\mathcal{C}^N(pk, sk)$ for the proof of Theorem 5.

Since $H_{\text{good}} : \mathcal{M} \to \mathcal{R}$ is a random function such that $H_{\text{good}}(m)$ is uniformly sampled from $\mathcal{R}_{\text{good}}(pk, sk, m)$ for any $m \in \mathcal{M}$, it is easy to check that $\Pr[1 \leftarrow \mathcal{C}^{N_1}(pk, sk)] = \Pr[1 \leftarrow \mathcal{B}^H(pk, sk) : H \xleftarrow{\$} \Omega_H]$ and $\Pr[1 \leftarrow \mathcal{C}^{N_2}(pk, sk)] = \Pr[1 \leftarrow \mathcal{B}^{H_{\text{good}}}(pk, sk)]$. Then, by Lemma 2 and the fact that $\mathcal{C}^N(pk, sk)$ ($N \in \{N_1, N_2\}$) makes $2q_H$ quantum queries to N, we can obtain

$$\begin{aligned}
&\left|\Pr\left[1 \leftarrow \mathcal{B}^H(pk, sk) : H \xleftarrow{\$} \Omega_H\right] - \Pr\left[1 \leftarrow \mathcal{B}^{H_{\text{good}}}(pk, sk)\right]\right| \\
&= \left|\Pr\left[1 \leftarrow \mathcal{C}^{N_1}(pk, sk)\right] - \Pr\left[1 \leftarrow \mathcal{C}^{N_2}(pk, sk)\right]\right| \\
&\le 8(2q_H + 1)^2 \cdot \max_{m \in \mathcal{M}} \delta(pk, sk, m) \overset{(a)}{=} 8(2q_H + 1)^2 \cdot \delta(pk, sk).
\end{aligned}$$

Here (a) uses the definition of $\delta(pk, sk)$ shown in Eq. (29). Combine the above equation with Eq. (32) and Eq. (33), we get

$$|\Pr[1 \leftarrow \mathbf{G_1} : (pk, sk)] - \Pr[1 \leftarrow \mathbf{G_2} : (pk, sk)]| \le 8(2q_H + 1)^2 \cdot \delta(pk, sk).$$

By averaging over $(pk, sk) \leftarrow \mathsf{Gen}$, we finally obtain

$$|\Pr[1 \leftarrow \mathbf{G_1}] - \Pr[1 \leftarrow \mathbf{G_2}]| \le 8(2q_H + 1)^2 \cdot \delta. \tag{34}$$

Game $\mathbf{G_3}$: Based on game $\mathbf{G_2}$, in this new game, we sample $R_1 \xleftarrow{\$} \Omega_R$ and let $G(m, c) = R_1(c)$ if $\mathsf{Enc}_{pk}(m; H(m)) = c$.

Obviously, games $\mathbf{G_2}$ and $\mathbf{G_3}$ have completely the same distribution if the following event does not occur:

$\exists m_1, m_2 \in \mathcal{M}$ *such that* $m_1 \neq m_2$ *and* $\mathsf{Enc}_{pk}(m_1; H(m_1)) = \mathsf{Enc}_{pk}(m_2; H(m_2))$.

Note that H has replaced by H_{good} in game $\mathbf{G_2}$ and $\mathbf{G_3}$, where $H_{\text{good}}(m)$ is uniformly sampled from $\mathcal{R}_{\text{good}}(pk, sk, m)$ for any $m \in \mathcal{M}$. Hence, by the definition of $\mathcal{R}_{\text{good}}(pk, sk, m)$ given in Eq. (28), we can conclude that the function

$$g(\cdot) := \mathsf{Enc}_{pk}(\cdot; H(\cdot))$$

is an injective function and thus the above event can not occur. Therefore,

$$\Pr[1 \leftarrow \mathbf{G_2}] = \Pr[1 \leftarrow \mathbf{G_3}]. \tag{35}$$

Game $\mathbf{G_4}$: This game is the same as game $\mathbf{G_3}$, except that the decapsulation oracle Deca is changed as it returns $K := R_1(c)$ for $c \neq c^*$. That is, the answer of Deca in this game no longer uses the secret key sk'.

We consider the following cases for the input c of Deca in games $\mathbf{G_3}$ and $\mathbf{G_4}$:

- $\mathsf{Dec}_{sk}(c) = m' \wedge m' \neq \perp \wedge \mathsf{Enc}_{pk}(m'; H(m')) = c$. In this case, Deca in games $\mathbf{G_3}$ and $\mathbf{G_4}$ return the same value $G(m', c) = R_1(c)$.
- $\mathsf{Dec}_{sk}(c) = \perp$ or $\mathsf{Dec}_{sk}(c) = m' \wedge m' \neq \perp \wedge \mathsf{Enc}_{pk}(m'; H(m')) \neq c$. In this case, uniformly random value $R(c)$ is returned in game $\mathbf{G_3}$ and random value $R_1(c)$ is returned in game $\mathbf{G_4}$. Indeed, for the adversary $\mathcal{A}$ of game $\mathbf{G_4}$, besides obtaining $R_1(c)$ through querying Deca, the $R_1(c)$ can only be obtained if it queries G with an input $(m_1 \neq \perp, c)$ satisfying $\mathsf{Enc}_{pk}(m_1; H(m_1)) = c$. However, since H has replaced by H_{good} in game $\mathbf{G_4}$ and $H_{\text{good}}(m) \in \mathcal{R}_{\text{good}}(pk, sk, m)$ for any $m \in \mathcal{M}$, such (m_1, c) pair does not exist. The reason is that, if such (m_1, c) pair exists, we must have $\mathsf{Enc}_{pk}(m_1; H(m_1)) = c$ and $\mathsf{Dec}_{sk}(c) \neq m_1$[13], which means that $H(m_1) = H_{\text{good}}(m_1) \notin \mathcal{R}_{\text{good}}(pk, sk, m_1)$ by the definition of $\mathcal{R}_{\text{good}}(pk, sk, m)$ shown in Eq. (28), and this is contradictory with $H_{\text{good}}(m) \in \mathcal{R}_{\text{good}}(pk, sk, m)$ for any $m \in \mathcal{M}$. Therefore, the random value $R_1(c)$ returned by Deca in game $\mathbf{G_4}$ is also uniform, which means that the output distributions of Deca in game $\mathbf{G_3}$ and $\mathbf{G_4}$ are identical in $\mathcal{A}$'s view.

According to the above analysis, we have

$$\Pr[1 \leftarrow \mathbf{G_3}] = \Pr[1 \leftarrow \mathbf{G_4}]. \tag{36}$$

Game $\mathbf{G_5}$: Based on game $\mathbf{G_4}$, in this new game, we sample $G_2 \xleftarrow{\$} \Omega_G$ and let $G(m, c) = G_2(m, c)$ if $m = m^*$ and $c = c^*$.

According to the analysis to obtain Eq. (35), we know $g(\cdot) := \mathsf{Enc}_{pk}(\cdot; H(\cdot))$ is an injective function. Therefore, the adversary $\mathcal{A}$ in game $\mathbf{G_4}$ can only obtain $R_1(c^*)$ by querying G with (m^*, c^*)[14], which means that the distributions of $G(m^*, c^*)$ in game $\mathbf{G_4}$ and $\mathbf{G_5}$ are both uniformly random in $\mathcal{A}$'s view. Hence

$$\Pr[1 \leftarrow \mathbf{G_4}] = \Pr[1 \leftarrow \mathbf{G_5}]. \tag{37}$$

Game $\mathbf{G_6}$: This game is the same as game $\mathbf{G_5}$, except that we let $G(m, c) = G_2(m, c)$ if $\mathsf{Enc}_{pk}(m; H(m)) = c^*$ and $c = c^*$.

Similarly, since $g(\cdot) := \mathsf{Enc}_{pk}(\cdot; H(\cdot))$ is an injective function and $c^* := \mathsf{Enc}_{pk}(m^*; H(m^*))$, we know $\mathsf{Enc}_{pk}(m; H(m)) = c^*$ is equivalent with $m = m^*$. Hence,

$$\Pr[1 \leftarrow \mathbf{G_5}] = \Pr[1 \leftarrow \mathbf{G_6}]. \tag{38}$$

[13] If $\mathsf{Dec}_{sk}(c) = \perp$, we obviously have $\mathsf{Dec}_{sk}(c) \neq m_1$. If $\mathsf{Dec}_{sk}(c) = m' \wedge m' \neq \perp \wedge \mathsf{Enc}_{pk}(m'; H(m')) \neq c$, we have $m' \neq m_1$ since $\mathsf{Enc}_{pk}(m_1; H(m_1)) = c$.

[14] Although the decapsulation oracle Deca in game $\mathbf{G_4}$ is also answered by querying R_1, $\mathcal{A}$ is not allowed to query Deca by the challenge ciphertext c^*.

Game $\mathbf{G_7}$: In game $\mathbf{G_2}$ to $\mathbf{G_6}$, we note that H has replaced by H_{good}, in this new game, we remove this replacement. That is, the H in this new game is uniformly sampled from Ω_H.

With the similar analysis in the game hop between games $\mathbf{G_1}$ and $\mathbf{G_2}$, we can first convert the distinguishing problem between games $\mathbf{G_6}$ and $\mathbf{G_7}$ to the distinguishing problem between H_{good} and H, and then use the Lemma 2 to upper bound the corresponding distinguishing advantage. But one thing to note is that the answer of $G(m, c)$ in game $\mathbf{G_6}$ and $\mathbf{G_7}$ also needs to query H, which results in the total query times of H in game $\mathbf{G_6}$ (resp. $\mathbf{G_7}$) being $q_H + q_G$ instead of the q_H in game $\mathbf{G_1}$ (resp. $\mathbf{G_2}$). Overall, we have

$$|\Pr[1 \leftarrow \mathbf{G_6}] - \Pr[1 \leftarrow \mathbf{G_7}]| \leq 8(2(q_H + q_G) + 1)^2 \cdot \delta. \tag{39}$$

Game $\mathbf{G_8}$: This game is the same as game $\mathbf{G_7}$, except that we let $G = G_2$ and replace $\mathcal{A}^{H,G,\mathsf{Deca}}(pk, c^*, k_b^*)$ by a new adversary $\mathcal{B}_2^{H,G}(pk, c^*, k_b^*)$ defined as:

1. Choose a $2q_G$-wise (resp. $2(q_G + q_D)$-wise) independent function to simulate the random oracle G_1 (resp. R_1).
2. Run the adversary $\mathcal{A}^{H,G,\mathsf{Deca}}(pk, c^*, k_b^*)$ in game $\mathbf{G_7}$ and finally output $\mathcal{A}$'s single bit output b'.
   (a) When $\mathcal{A}$ queries its oracle H, $\mathcal{B}_2$ answers it by querying its oracle H.
   (b) When $\mathcal{A}$ queries its oracle G with (m, c), $\mathcal{B}_2$ computes $\mathsf{Enc}_{pk}(m, H(m))$ by querying its oracle H and does:
      i. if $\mathsf{Enc}_{pk}(m, H(m)) = c^* \wedge c = c^*$, $\mathcal{B}_2$ answers $G(m, c)$ by querying its oracle G.
      ii. else if $\mathsf{Enc}_{pk}(m, H(m)) = c$, $\mathcal{B}_2$ answers $R_1(c)$.
      iii. else if $\mathsf{Enc}_{pk}(m, H(m)) \neq c$, $\mathcal{B}_2$ answers $G_1(m, c)$.
   (c) When $\mathcal{A}$ queries its oracle Deca with input c $(\neq c^*)$, $\mathcal{B}_2$ answers $R_1(c)$.

Obviously, $\mathcal{B}_2$ makes parallel quantum queries to its oracle H (resp. G) with query depth at most $d_H + d_G$ (resp. d_G) and query width n. The running time of $\mathcal{B}_2$ can also be bounded as $\mathcal{T}_{\mathcal{B}_2} \approx \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}} + O(q_G^2 + q_D^2)$.

It is easy to check that the change in game $\mathbf{G_8}$ compared with game $\mathbf{G_7}$ is only conceptual, and $\mathcal{B}_2$ actually an $\mathsf{IND\text{-}CPA}$ adversary against $\mathsf{KEM}^{\not\perp}$ in the QROM. Hence, we have

$$\Pr[1 \leftarrow \mathbf{G_7}] = \Pr[1 \leftarrow \mathbf{G_8}] \text{ and } \Pr[1 \leftarrow \mathbf{G_8}] = \mathrm{Adv}_{\mathsf{KEM}^{\not\perp}}^{\mathsf{IND\text{-}CPA}}(\mathcal{B}_2). \tag{40}$$

Combine Eq. (30), Eq. (31) with Eq. (34) to Eq. (40), we finally obtain

$$\mathrm{Adv}_{\mathsf{KEM}^{\not\perp}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A}) \leq \mathrm{Adv}_{\mathsf{F}}^{\mathsf{PRF}}(\mathcal{B}_1) + \mathrm{Adv}_{\mathsf{KEM}^{\not\perp}}^{\mathsf{IND\text{-}CPA}}(\mathcal{B}_2) + 16(2q_H + 2q_G + 1)^2 \cdot \delta.$$

□

A.7 Quantum Circuit to Answer the Quantum Oracle K

In this section, based on a quantum oracle $N : \mathcal{M} \to \{0,1\}$, we construct a quantum circuit to answer the quantum oracle $K : \mathcal{M} \to \mathcal{R}$ defined in Fig. 9. Here the definition of the set $\mathcal{R}_{\text{bad}}(pk, sk, m)$ and $\mathcal{R}_{\text{good}}(pk, sk, m)$ is shown in Eq. (27) and Eq. (28), respectively.

```
$K(m \in \mathcal{M})$
if $N(m) = 0$
    return $y \xleftarrow{\$} \mathcal{R}_{\text{good}}(pk, sk, m)$
if $N(m) = 1$
    return $y \xleftarrow{\$} \mathcal{R}_{\text{bad}}(pk, sk, m)$
```

Fig. 9. Quantum oracle K. Here we just write the classical description of K for brevity.

Let $H_0 : \mathcal{M} \to \mathcal{R}$ be a random function such that $H_0(m)$ is uniformly sampled from $\mathcal{R}_{\text{good}}(pk, sk, m)$ for any $m \in \mathcal{M}$. Let $H_1 : \mathcal{M} \to \mathcal{R}$ be a random function such that $H_1(m)$ is uniformly sampled from $\mathcal{R}_{\text{bad}}(pk, sk, m)$ for each $m \in \mathcal{M}$. Then, we define unitary operation

$$U_{H_0} : |m, r\rangle \mapsto |m, r \oplus H_0(m)\rangle \text{ and } U_{H_1} : |m, r\rangle \mapsto |m, r \oplus H_1(m)\rangle,$$

where $m \in \mathcal{M}$ and $r \in \mathcal{R}$. Let M/R be the input/output register of the quantum oracle K. Now, we construct the following quantum circuit to answer the quantum oracle K:

- Initialize a single qubit register L with $|0\rangle$.
- Query N by registers M and L, where L is the output register.
- Apply the following conditional operation:
  - The controlling register is L, and apply U_{H_1} on registers M and R (R is the output register) if the state on L is $|1\rangle$.
- Apply unitary operation X on register L, where $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$.
- Apply the following conditional operation:
  - The controlling register is L, and apply U_{H_0} on registers M and R (R is the output register) if the state on L is $|1\rangle$.
- Apply unitary operation X on register L again.
- Query N by registers M and L again, where L is the output register. At this point the register L is guaranteed to be in state $|0\rangle$, so it can be discarded.

By utilizing the above circuit, $\mathcal{C}^N(pk, sk)$ defined in Fig. 8 can answer once query to K by querying N twice. Note that the above circuit needs to apply unitary operation U_{H_0} and U_{H_1}. Indeed, by sampling H_0 and H_1 directly, $\mathcal{C}^N(pk, sk)$ can perform these two unitary operations. We stress that there is no problem asking $\mathcal{C}^N(pk, sk)$ to sample H_0 and H_1 directly, because $\mathcal{C}$ has input (pk, sk) and we do not limit $\mathcal{C}$ to be polynomial time, it can be unbounded.

A.8 Proof of Theorem 6

Transformation $\mathsf{U}^{\not\perp}$: Let $\mathsf{dPKE} = (\mathsf{Gen}, \mathsf{dEnc}, \mathsf{dDec})$ be a dPKE scheme with message space $\mathcal{M}$ and ciphertext space $\mathcal{C}$. For a given set $\mathcal{K}$, let $G : \mathcal{M} \to \mathcal{K}$ be a hash function, and let $\mathsf{F} : \mathcal{K}^{prf} \times \mathcal{C} \to \mathcal{K}$ be a pseudorandom function (PRF) with key space $\mathcal{K}^{prf}$. We associate KEM scheme

$$\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}] := (\mathsf{Gen}', \mathsf{Enca}', \mathsf{Deca}').$$

The constituting algorithms of $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$ are given in Fig. 10. Here the key space of $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$ is $\mathcal{K}$.

```
$\mathsf{Gen}'$
    $(pk, sk) \leftarrow \mathsf{Gen}$
    $s \xleftarrow{\$} \mathcal{K}^{prf}$
    $sk' := (sk, s)$
    return $(pk, sk')$
```

```
$\mathsf{Enca}'(pk)$
    $m \xleftarrow{\$} \mathcal{M}$
    $c := \mathsf{dEnc}_{pk}(m)$
    $K := G(m, c)$
    return $(K, c)$
```

```
$\mathsf{Deca}'(sk' = (sk, s), c)$
    $m' := \mathsf{dDec}_{sk}(c)$
    if $m' = \perp$
        return $K := \mathsf{F}(s, c)$
    else return $K := G(m', c)$
```

Fig. 10. Key Encapsulation Mechanism $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$.

Theorem 6 (OW-CPA of dPKE $\overset{\text{QROM}}{\Rightarrow}$ IND-CPA of $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$). *For a dPKE scheme $\mathsf{dPKE} = (\mathsf{Gen}, \mathsf{dEnc}, \mathsf{dDec})$ with message space $\mathcal{M}$, let $\mathcal{A}$ be an* IND-CPA *adversary against $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$, making parallel quantum queries to the random oracle G with query depth d_G and query width n. Let $q_G := d_G \cdot n$.*

Then, we can construct a OW-CPA *adversary $\mathcal{A}_1$ against* dPKE *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp}[\mathsf{dPKE},G,\mathsf{F}]}(\mathcal{A}) \leq 2\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{dPKE}}(\mathcal{A}_1) + 2\sqrt{d_G} \cdot \Pr[E_{\mathsf{dPKE}}]$$

and $\mathcal{T}_{\mathcal{A}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{dEnc}} + O(q_G^2)$. Here E_{dPKE} is the following event:

$$(pk, sk) \leftarrow \mathsf{Gen},\ m^* \xleftarrow{\$} \mathcal{M},\ \exists m \neq m^* \text{ such that } \mathsf{dEnc}_{pk}(m) = \mathsf{dEnc}_{pk}(m^*).$$

Proof. Let Ω_G be the set of all functions $G : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$, where $\mathcal{C}$ is the ciphertext space of dPKE and $\mathcal{K}$ is the key space of $\mathsf{U}^{\not\perp}[\mathsf{dPKE}, G, \mathsf{F}]$. Let D be a joint distribution of the tuple $(G, H, \{(m^*, c^*)\}, pk, c^*)$, where $G \xleftarrow{\$} \Omega_G$, $m^* \xleftarrow{\$} \mathcal{M}$, $c^* := \mathsf{dEnc}_{pk}(m^*)$ and pk is sampled by $(pk, sk) \leftarrow \mathsf{Gen}$; H is identical with G except that $H(m^*, c^*)$ is a fresh value uniformly sampled from $\mathcal{K}$.

Then, we introduce two games $\mathbf{G}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp},b=0}$ and $\mathbf{G}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp},b=1}$ as shown in Fig. 11, and it is easy to check that

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp}[\mathsf{dPKE},G,\mathsf{F}]}(\mathcal{A}) = \frac{1}{2}\left|\Pr\left[1 \leftarrow \mathbf{G}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp},b=0}\right] - \Pr\left[1 \leftarrow \mathbf{G}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp},b=1}\right]\right|. \quad (41)$$

```
$\mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=0}$
    $(G, H, \{(m^*, c^*)\}, pk, c^*) \leftarrow D$
    $b := 0$
    $k_0^* := G(m^*, c^*), k_1^* \xleftarrow{\$} \mathcal{K}$
    $b' \leftarrow \mathcal{A}^G(pk, c^*, k_b^*)$
    return $b'$
```

```
$\mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=1}$
    $(G, H, \{(m^*, c^*)\}, pk, c^*) \leftarrow D$
    $b := 1$
    $k_0^* := G(m^*, c^*), k_1^* \xleftarrow{\$} \mathcal{K}$
    $b' \leftarrow \mathcal{A}^G(pk, c^*, k_b^*)$
    return $b'$
```

```
$\mathbf{G_0}$
    $(G, H, \{(m^*, c^*)\}, pk, c^*, k) \leftarrow D_1$
    $b' \leftarrow \mathcal{A}^G(pk, c^*, k)$
    return $b'$
```

```
$\mathbf{G_1}$
    $(G, H, \{(m^*, c^*)\}, pk, c^*, k) \leftarrow D_1$
    $b' \leftarrow \mathcal{A}^H(pk, c^*, k)$
    return $b'$
```

Fig. 11. Games for the proof of Theorem 6.

Next, we rewrite game $\mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=0}$ (resp. $\mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=1}$) to a new game $\mathbf{G_0}$ (resp. $\mathbf{G_1}$) as shown in Fig. 11. The distribution D_1 involved in games $\mathbf{G_0}$ and $\mathbf{G_1}$ is a joint distribution identical with D, except that an additional value $k := G(m^*, c^*)$ is sampled. Obviously,

$$\Pr\left[1 \leftarrow \mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=0}\right] = \Pr[1 \leftarrow \mathbf{G_0}] = \Pr_{(G,H,\{(m^*,c^*)\},pk,c^*,k)\leftarrow D_1}[1 \leftarrow \mathcal{A}^G(pk, c^*, k)],$$

$$\Pr\left[1 \leftarrow \mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=1}\right] = \Pr[1 \leftarrow \mathbf{G_1}] = \Pr_{(G,H,\{(m^*,c^*)\},pk,c^*,k)\leftarrow D_1}[1 \leftarrow \mathcal{A}^H(pk, c^*, k)].$$

Let $S := \{(m^*, c^*)\}$ and $z := (pk, c^*, k)$, the above equation actually implies that

$$\begin{aligned} &\left|\Pr\left[1 \leftarrow \mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=0}\right] - \Pr\left[1 \leftarrow \mathbf{G}^{\textsf{IND-CPA}}_{\mathsf{U}^{\not\perp},b=1}\right]\right| \\ &= \left|\Pr_{(G,H,S,z)\leftarrow D_1}[1 \leftarrow \mathcal{A}^G(z)] - \Pr_{(G,H,S,z)\leftarrow D_1}[1 \leftarrow \mathcal{A}^H(z)]\right|. \end{aligned} \tag{42}$$

Then, by applying Theorem 4 with the distribution D_1, we can construct an adversary $\mathcal{D}^{G,H,1_S}(z)$ such that

$$\begin{aligned} &\left|\Pr_{(G,H,S,z)\leftarrow D_1}[1 \leftarrow \mathcal{A}^G(z)] - \Pr_{(G,H,S,z)\leftarrow D_1}[1 \leftarrow \mathcal{A}^H(z)]\right| \\ &\leq 4\sqrt{d_G} \cdot \Pr_{(G,H,S,z)\leftarrow D_1}[T \cap S \neq \emptyset : T \leftarrow \mathcal{D}^{G,H,1_S}(z)]. \end{aligned} \tag{43}$$

Here 1_S is the indicator function of S, that is, $1_S(x) = 1$ if $x = (m^*, c^*)$ and 0 otherwise. $\mathcal{D}$ makes parallel quantum queries to G, H and 1_S all with query depth at most $3d_G$ and query width n. The running time of $\mathcal{D}$ is $\mathcal{T}_\mathcal{D} \lesssim 3 \cdot \mathcal{T}_\mathcal{A}$.

Now, we construct a $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{A}_1$ against dPKE as:

1. $\mathcal{A}_1$ gets the challenge ciphertext $c^* := \mathsf{dPKE}_{pk}(m^*)$ and public key pk, where m^* is the challenge plaintext uniformly sampled from $\mathcal{M}$ by the challenger and pk is sampled by $(pk, sk) \leftarrow \mathsf{Gen}$.
2. $\mathcal{A}_1$ samples k uniformly from $\mathcal{K}$ and chooses a $12q_G$-wise function $f : \mathcal{M} \times \mathcal{C} \to \mathcal{K}$.
3. $\mathcal{A}_1$ uses $z = (pk, c^*, k)$ as input to run the adversary $\mathcal{D}^{G,H,1_S}(z)$ and simulates the oracles G, H and 1_S for $\mathcal{D}$[15]:
   (a) When $\mathcal{D}$ queries G with input (m, c):
      i. If $\mathsf{dPKE}_{pk}(m) = c^*$ and $c = c^*$, $\mathcal{A}_1$ answers k.
      ii. If $\mathsf{dPKE}_{pk}(m) \neq c^*$ or $c \neq c^*$, $\mathcal{A}_1$ answers $f(m, c)$.
   (b) When $\mathcal{D}$ queries H with input (m, c), $\mathcal{A}_1$ answers $f(m, c)$.
   (c) When $\mathcal{D}$ queries 1_S with input (m, c):
      i. If $\mathsf{dPKE}_{pk}(m) = c^*$ and $c = c^*$, $\mathcal{A}_1$ answers 1.
      ii. If $\mathsf{dPKE}_{pk}(m) \neq c^*$ or $c \neq c^*$, $\mathcal{A}_1$ answers 0.
4. After $\mathcal{D}$ returns set T, $\mathcal{A}_1$ searches in T for the m satisfying $\exists c \in \mathcal{C}$ s.t. $(m, c) \in T$ and $\mathsf{dPKE}_{pk}(m) = c^*$. $\mathcal{A}_1$ finally outputs the minimum m satisfies that property. If such m does not exist, $\mathcal{A}_1$ outputs $\perp$.

One can check that the running time of $\mathcal{A}_1$ is $\mathcal{T}_{\mathcal{A}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{dEnc}} + O(q_G^2)$.

In the above construction, since $\mathcal{A}_1$ can not get the challenge plaintext m^* directly, it checks if m equals m^* by testing if $\mathsf{dPKE}_{pk}(m)$ equals c^* during the simulation of oracles G and 1_S for $\mathcal{D}$. Indeed, since m^* is uniformly sampled from $\mathcal{M}$ by the challenger, $\mathcal{A}_1$ simulates the oracles G, H and 1_S for $\mathcal{D}$ perfectly[16] if the following event does not occur:

$$(pk, sk) \leftarrow \mathsf{Gen},\ m^* \xleftarrow{\$} \mathcal{M},\ \exists m \neq m^* \textit{ such that } \mathsf{dEnc}_{pk}(m) = \mathsf{dEnc}_{pk}(m^*).$$

The reason is that, if the above event does not occur, $\mathsf{dEnc}_{pk}(m) = \mathsf{dEnc}_{pk}(m^*)$ is equivalent with $m = m^*$. Note that the above event is exactly the E_{dPKE}, thus we have

$$\Pr_{(G,H,S,z)\leftarrow D_1}[T \cap S \neq \emptyset : T \leftarrow \mathcal{D}^{G,H,1_S}(z)] \leq \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{dPKE}}(\mathcal{A}_1) + \Pr[E_{\mathsf{dPKE}}]. \quad (44)$$

Finally, combine Eq. (41) to Eq. (44), we obtain

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp}[\mathsf{dPKE},G,\mathsf{F}]}(\mathcal{A}) \leq 2\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{dPKE}}(\mathcal{A}_1) + 2\sqrt{d_G} \cdot \Pr[E_{\mathsf{dPKE}}].$$

□

[15] Note that $\mathcal{D}$ actually makes parallel quantum queries to G, H and 1_S. Here we just explain their simulations in the classical manner for brevity.

[16] Here the "perfectly" means that, in $\mathcal{D}$'s view, the oracles G, H and 1_S simulated by $\mathcal{A}_1$ is perfectly indistinguishable with the oracles G, H and 1_S obtained by sampling $(G, H, \{(m^*, c^*)\}, pk, c^*, k)$ according to the distribution D_1.

A.9 Proof of Theorem 7

Before proving, we first count that how many times the $\mathcal{A}_1$ constructed in the proof of Theorem 6 has computed the encryption algorithm dEnc of the dPKE scheme dPKE. This value will be used in the proof of Theorem 7.

Remark 8. Indeed, the computation of dEnc performed by $\mathcal{A}_1$ contains the following three parts:

1. The first part derived from the simulation of oracle G for $\mathcal{D}^{G,H,1_S}(z)$. Since $\mathcal{D}^{G,H,1_S}(z)$ makes parallel quantum queries to G with query depth at most $3d_G$ and query width n, $\mathcal{A}_1$ needs to compute dEnc also in parallel with depth $3d_G$ and width n to simulate oracle G for $\mathcal{D}^{G,H,1_S}(z)$.
2. The second part derived from the simulation of oracle 1_S for $\mathcal{D}^{G,H,1_S}(z)$. Since $\mathcal{D}^{G,H,1_S}(z)$ makes parallel quantum queries to 1_S with query depth at most $3d_G$ and query width n, $\mathcal{A}_1$ needs to compute dEnc also in parallel with depth $3d_G$ and width n to simulate oracle 1_S for $\mathcal{D}^{G,H,1_S}(z)$.
3. The third part derived from the the final check of $\mathcal{A}_1$ after $\mathcal{D}^{G,H,1_S}(z)$ returns the set T. By the detailed construction of $\mathcal{D}^{G,H,1_S}(z)$ given in the proof of Theorem 4, we know that T is a set with $|T| = n$, where n is the query width of oracles G and H. Hence, without loss of generality, we can say that $\mathcal{A}_1$ compute dEnc in parallel with depth 1 and width n in this final check.

In summary, $\mathcal{A}_1$ needs to compute dEnc in parallel with depth in total $6d_G + 1$ and width n.

Theorem 7 (IND-CPA/OW-CPA **of** $\mathsf{P} \overset{\text{QROM}}{\Rightarrow}$ IND-CPA **of** $\mathsf{KEM}^{\not\perp}$)**.** *Let* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a* δ*-correct rPKE scheme with message space* $\mathcal{M}$. *Let* $\mathcal{A}$ *be an* IND-CPA *adversary against* $\mathsf{KEM}^{\not\perp} = \mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}]$, *making parallel quantum queries to the random oracle* H *(resp.* G*) with query depth* d_H *(resp.* d_G*) and query width* n. *Let* $q_H := d_H \cdot n$ *and* $q_G := d_G \cdot n$.

Then, we can construct an IND-CPA *adversary* $\mathcal{B}$ *against* P *such that*

$$\begin{aligned}\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq & 2\sqrt{d_G}(6d_G + d_H + 3) \cdot \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + 4\sqrt{d_G} \cdot \delta \\ & + 16\sqrt{d_G}(6d_G + d_H + 3)\frac{(6q_G + 2q_H + 1)}{|\mathcal{M}|}.\end{aligned}$$

and $\mathcal{T}_{\mathcal{B}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}}$. *If* P *is also unique randomness recoverable with the recover algorithm* Rec, *we can also construct a* OW-CPA *adversary* $\mathcal{B}_1$ *against* P *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq 20\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}_1) + 36\sqrt{d_G} \cdot \delta$$

and $\mathcal{T}_{\mathcal{B}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G + q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$.

Proof. Since $\mathsf{FO}^{\not\perp}[\mathsf{P}, H, G, \mathsf{F}] = \mathsf{U}^{\not\perp}[\mathsf{T}[\mathsf{P}, H], G, \mathsf{F}]$ (i,e, Eq. (20)), we can compute

$$\begin{aligned}\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) &= \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}^{\not\perp}[\mathsf{T}[\mathsf{P},H],G,\mathsf{F}]}(\mathcal{A}) \\ &\overset{(a)}{\leq} 2\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{T}[\mathsf{P},H]}(\mathcal{A}_1) + 2\sqrt{d_G} \cdot \Pr\left[E_{\mathsf{T}[\mathsf{P},H]}\right].\end{aligned} \tag{45}$$

Here (a) follows form Theorem 6. Note that in Fig. 2, the encryption algorithm Enc' of dPKE scheme $\mathsf{T}[\mathsf{P}, H]$ satisfies that $\mathcal{T}_{\mathsf{Enc}'} \approx \mathcal{T}_{\mathsf{Enc}}$. Hence, the running time of adversary $\mathcal{A}_1$ is $\mathcal{T}_{\mathcal{A}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}} + O(q_G^2)$.

Now, let's count that how many times the adversary $\mathcal{A}_1$ obtained in Eq. (45) queried the random oracle H. Firstly, the inequality (a) of Eq. (45) uses the Theorem 6, which, in essence, only considers the random oracle G queries of the adversary $\mathcal{A}$. Consequently, $\mathcal{A}_1$ inherits the H queries of $\mathcal{A}$, meaning that $\mathcal{A}_1$ needs to make parallel quantum queries to H with query depth d_H and query width n. Additionally, based on the analysis of Remark 8, $\mathcal{A}_1$ needs to compute Enc'[17] in parallel with depth in total $6d_G + 1$ and width n. As shown in Fig. 2, the computation of Enc' needs to query H once, which means that $\mathcal{A}_1$ also needs to make parallel quantum queries to H with query depth $6d_G + 1$ and query width n. In summary, $\mathcal{A}_1$ needs to make parallel quantum queries to H with query depth $6d_G + d_H + 1$ and query width n.

Then, by using Lemma 7, we can construct an $\mathsf{IND\text{-}CPA}$ adversary $\mathcal{B}$ against P such that

$$\mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{T}[\mathsf{P},H]}(\mathcal{A}_1) \leq (6d_G + d_H + 3) \cdot \left(\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + \frac{8(6q_G + q_H + n + 1)}{|\mathcal{M}|} \right) \tag{46}$$

and $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}_1} + O(q_G^2 + q_H^2)$. As for the probability $\Pr[E_{\mathsf{T}[\mathsf{P},H]}]$, by the definition of the event E_{dPKE}, $E_{\mathsf{T}[\mathsf{P},H]}$ actually denotes the following event:

$$(pk, sk) \leftarrow \mathsf{Gen},\ H \xleftarrow{\$} \Omega_H,\ m^* \xleftarrow{\$} \mathcal{M},\ \exists m \neq m^*\ \textit{such that}$$
$$\mathsf{Enc}_{pk}(m; H(m)) = \mathsf{Enc}_{pk}(m^*; H(m^*)).$$

Here Ω_H is the set of all functions $H : \mathcal{M} \to \mathcal{R}$, and the $H \xleftarrow{\$} \Omega_H$ stems from the fact that we consider H as a random oracle. Obviously, by using Lemma 9,

$$\Pr\left[E_{\mathsf{T}[\mathsf{P},H]}\right] \leq 2 \cdot \delta. \tag{47}$$

Combine Eq. (45) to Eq. (47) and use the fact that $n \leq q_H$, we finally obtain

$$\begin{aligned}\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq & 2\sqrt{d_G}(6d_G + d_H + 3) \cdot \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + 4\sqrt{d_G} \cdot \delta \\ & + 16\sqrt{d_G}(6d_G + d_H + 3)\frac{(6q_G + 2q_H + 1)}{|\mathcal{M}|}.\end{aligned}$$

The running time of $\mathcal{B}$ is $\mathcal{T}_{\mathcal{B}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}}$.

If P is also unique randomness recoverable with the recover algorithm Rec, by using Lemma 8, we can also construct a $\mathsf{OW\text{-}CPA}$ adversary $\mathcal{B}_1$ against P such that

$$\mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{T}[\mathsf{P},H]}(\mathcal{A}_1) \leq 10 \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}_1) + 16 \cdot \delta \tag{48}$$

and $\mathcal{T}_{\mathcal{B}_1} \approx \mathcal{T}_{\mathcal{A}_1} + O(q_G^2 + q_H^2) + O(q_G + q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$. Combine Eq. (45), Eq. (47) with Eq. (48), we finally obtain

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}^{\not\perp}}(\mathcal{A}) \leq 20\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}_1) + 36\sqrt{d_G} \cdot \delta$$

and $\mathcal{T}_{\mathcal{B}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G + q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$. □

[17] Since we consider $\mathsf{T}[\mathsf{P}, H]$, the dEnc involved in Remark 8 actually the Enc'.

A.10 Tighter IND-CCA Security Proof of $\mathsf{FO}_m^{\perp}$ in the QROM

FO-like transformation $\mathsf{FO}_m^{\perp}$: Let $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an rPKE scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$. For a given set $\mathcal{K}$, let $H : \mathcal{M} \to \mathcal{R}$, $G : \mathcal{M} \to \mathcal{K}$ be hash functions. We associate KEM scheme

$$\mathsf{KEM}_m^{\perp} := \mathsf{FO}_m^{\perp}[\mathsf{P}, H, G] = (\mathsf{Gen}, \mathsf{Enca}, \mathsf{Deca}_m^{\perp})$$

that has key space $\mathcal{K}$. The constituting algorithms of $\mathsf{KEM}_m^{\perp}$ are given in Fig. 12.

```
Gen
    $(pk, sk) \leftarrow \mathsf{Gen}$
    return $(pk, sk)$
```

```
Enca($pk$)
    $m \xleftarrow{\$} \mathcal{M}$
    $c := \mathsf{Enc}_{pk}(m; H(m))$
    $K := G(m)$
    return $(K, c)$
```

```
Deca$_m^{\perp}$($sk, c$)
    $m' := \mathsf{Dec}_{sk}(c)$
    if $m' = \perp \vee c \neq \mathsf{Enc}_{pk}(m'; H(m'))$
        return $\perp$
    else return $K := G(m')$
```

Fig. 12. Key Encapsulation Mechanism $\mathsf{KEM}_m^{\perp}$.

We first introduce the following theorem. It shows that in the QROM, the IND-CPA security of $\mathsf{KEM}_m^{\perp}$ implies its IND-CCA security.

Theorem 8 (**IND-CPA of $\mathsf{KEM}_m^{\perp} \overset{\text{QROM}}{\Rightarrow}$ IND-CCA of $\mathsf{KEM}_m^{\perp}$ [14, Theorem 2]).** *Let rPKE scheme $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be δ-correct and weakly γ-spread. Let $\mathcal{A}$ be an IND-CCA adversary against $\mathsf{KEM}_m^{\perp} = \mathsf{FO}_m^{\perp}[\mathsf{P}, H, G]$, making q_D classical queries to the decapsulation oracle, making parallel quantum queries to the random oracle H (resp. G) with query depth d_H (resp. d_G) and query width n. Let $q_H := d_H \cdot n$ and $q_G := d_G \cdot n$.*

Then, we can construct an IND-CPA adversary $\mathcal{B}$ (in the QROM) against $\mathsf{KEM}_m^{\perp}$ such that

$$\mathrm{Adv}_{\mathsf{KEM}_m^{\perp}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A}) \leq \mathrm{Adv}_{\mathsf{KEM}_m^{\perp}}^{\mathsf{IND\text{-}CPA}}(\mathcal{B}) + 8\sqrt{q_H(q_H+1)\cdot\delta} + (64q_H+2)\cdot\delta + 40q_D\cdot 2^{-\gamma/2}.$$

The adversary $\mathcal{B}$ makes parallel quantum queries to the random oracle H (resp. G) with query depth $2d_H$ (resp. $d_G + q_D$) and query width n. The running time of adversary $\mathcal{B}$ is $\mathcal{T}_{\mathcal{B}} \approx \mathcal{T}_{\mathcal{A}} + O(q_H q_D + q_H^2)$.

Next, we focus on the IND-CPA security of $\mathsf{KEM}_m^{\perp}$ in the QROM. As introduced in [15], the $\mathsf{KEM}_m^{\perp}$ satisfies that

$$\mathsf{KEM}_m^{\perp} = \mathsf{FO}_m^{\perp}[\mathsf{P}, H, G] = \mathsf{U}_m^{\perp}[\mathsf{T}[\mathsf{P}, H], G]. \tag{49}$$

Here transformation $\mathsf{U}_m^{\perp}$ transforms a dPKE scheme into a KEM scheme. For the $\mathsf{U}_m^{\perp}$, we can prove the following theorem, which shows that in the QROM, the IND-CPA security of $\mathsf{U}_m^{\perp}$ can be reduced to the OW-CPA security of the underlying dPKE scheme without the square-root advantage loss.

Theorem 9 **($\mathsf{OW\text{-}CPA}$ of dPKE $\overset{\text{QROM}}{\Rightarrow}$ $\mathsf{IND\text{-}CPA}$ of $\mathsf{U}_m^{\perp}[\mathsf{dPKE}, G]$).** *For a dPKE scheme* $\mathsf{dPKE} = (\mathsf{Gen}, \mathsf{dEnc}, \mathsf{dDec})$ *with message space* $\mathcal{M}$*, let* $\mathcal{A}$ *be an* $\mathsf{IND\text{-}CPA}$ *adversary against* $\mathsf{U}_m^{\perp}[\mathsf{dPKE}, G]$*, making parallel quantum queries to the random oracle* G *with query depth* d_G *and query width* n*. Let* $q_G := d_G \cdot n$*.*

Then, we can construct a $\mathsf{OW\text{-}CPA}$ *adversary* $\mathcal{A}_1$ *against* dPKE *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{U}_m^{\perp}[\mathsf{dPKE}, G, \mathsf{F}]}(\mathcal{A}) \leq 2\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{dPKE}}(\mathcal{A}_1) + 2\sqrt{d_G} \cdot \Pr[E_{\mathsf{dPKE}}]$$

and $\mathcal{T}_{\mathcal{A}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{dEnc}} + O(q_G^2)$*. Here* E_{dPKE} *is the following event:*

$$(pk, sk) \leftarrow \mathsf{Gen},\ m^* \xleftarrow{\$} \mathcal{M},\ \exists m \neq m^* \text{ such that } \mathsf{dEnc}_{pk}(m) = \mathsf{dEnc}_{pk}(m^*).$$

Proof. Since we only consider the $\mathsf{IND\text{-}CPA}$ security, where the adversary can not query the decapsulation oracle, the proof of this theorem is almost the same as Theorem 6 and we omit it. □

Combining Theorem 9 with Lemma 7 and Lemma 8, we can prove the following result for the $\mathsf{IND\text{-}CPA}$ security of $\mathsf{KEM}_m^{\perp}$ in the QROM.

Theorem 10 **($\mathsf{IND\text{-}CPA}/\mathsf{OW\text{-}CPA}$ of P $\overset{\text{QROM}}{\Rightarrow}$ $\mathsf{IND\text{-}CPA}$ of $\mathsf{KEM}_m^{\perp}$).** *Let* $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a* δ*-correct rPKE scheme with message space* $\mathcal{M}$*. Let* $\mathcal{A}$ *be an* $\mathsf{IND\text{-}CPA}$ *adversary against* $\mathsf{KEM}_m^{\perp} = \mathsf{FO}_m^{\perp}[\mathsf{P}, H, G]$*, making parallel quantum queries to the random oracle* H *(resp.* G*) with query depth* d_H *(resp.* d_G*) and query width* n*. Let* $q_H := d_H \cdot n$ *and* $q_G := d_G \cdot n$*.*

Then, we can construct an $\mathsf{IND\text{-}CPA}$ *adversary* $\mathcal{B}$ *against* P *such that*

$$\begin{aligned}\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}_m^{\perp}}(\mathcal{A}) \leq & 2\sqrt{d_G}(6d_G + d_H + 3) \cdot \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}) + 4\sqrt{d_G} \cdot \delta \\ & + 16\sqrt{d_G}(6d_G + d_H + 3)\frac{(6q_G + 2q_H + 1)}{|\mathcal{M}|}.\end{aligned}$$

and $\mathcal{T}_{\mathcal{B}} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}}$*. If* P *is also unique randomness recoverable with the recover algorithm* Rec*, we can also construct a* $\mathsf{OW\text{-}CPA}$ *adversary* $\mathcal{B}_1$ *against* P *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{KEM}_m^{\perp}}(\mathcal{A}) \leq 20\sqrt{d_G} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{P}}(\mathcal{B}_1) + 36\sqrt{d_G} \cdot \delta$$

and $\mathcal{T}_{\mathcal{B}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2) + O(q_G + q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$*.*

Proof. Similar to Theorem 7, this theorem can be easily proved by utilizing Eq. (49) and Lemma 9. As for the detailed proof, we omit it since it is almost the same as the proof of Theorem 7. □

Combine Theorem 8 with Theorem 10, we finally obtain the following corollary. It shows that, in the QROM, the $\mathsf{IND\text{-}CCA}$ security of $\mathsf{KEM}_m^{\perp}$ can be reduced to the $\mathsf{IND\text{-}CPA}/\mathsf{OW\text{-}CPA}$ security of the underlying rPKE scheme without the square-root advantage loss.

Corollary 2 (IND-CPA/OW-CPA of $\mathsf{P} \overset{\mathrm{QROM}}{\Rightarrow}$ IND-CCA of $\mathsf{KEM}_m^{\perp}$). *Let rPKE scheme $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\mathcal{M}$ be δ-correct and weakly γ-spread. Let $\mathcal{A}$ be an* IND-CCA *adversary against $\mathsf{KEM}_m^{\perp} = \mathsf{FO}_m^{\perp}[\mathsf{P}, H, G]$, making q_D classical queries to the decapsulation oracle, making parallel quantum queries to the random oracle H (resp. G) with query depth d_H (resp. d_G) and query width n. Let $q_H := d_H \cdot n$ and $q_G := d_G \cdot n$.*

Then, we can construct an IND-CPA *adversary $\mathcal{B}_1$ against* P *such that*

$$\begin{aligned}\mathrm{Adv}_{\mathsf{KEM}_m^{\perp}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A}) \leq\ & 2\sqrt{d_G + q_D}(6d_G + 2d_H + 6q_D + 3) \cdot \mathrm{Adv}_{\mathsf{P}}^{\mathsf{IND\text{-}CPA}}(\mathcal{B}_1) \\ & + 4\sqrt{d_G + q_D} \cdot \delta + 8\sqrt{q_H(q_H + 1) \cdot \delta} \\ & + (64q_H + 2) \cdot \delta + 40q_D \cdot 2^{-\gamma/2} \\ & + 16\sqrt{d_G + q_D}(6d_G + 2d_H + 6q_D + 3)\frac{(8q_G + 8q_D + 4q_H + 1)}{|\mathcal{M}|}\end{aligned}$$

and $\mathcal{T}_{\mathcal{B}_1} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G) \cdot \mathcal{T}_{\mathsf{Enc}} + O(q_G^2 + q_H^2 + q_H q_D)$. If P *is also unique randomness recoverable with the recover algorithm* Rec*, we can also construct an* OW-CPA *adversary $\mathcal{B}_2$ against* P *such that*[18]

$$\begin{aligned}\mathrm{Adv}_{\mathsf{KEM}_m^{\perp}}^{\mathsf{IND\text{-}CCA}}(\mathcal{A}) \leq\ & 20\sqrt{d_G + q_D} \cdot \mathrm{Adv}_{\mathsf{P}}^{\mathsf{OW\text{-}CPA}}(\mathcal{B}_2) + 36\sqrt{d_G + q_D} \cdot \delta \\ & + 8\sqrt{q_H(q_H + 1) \cdot \delta} + (64q_H + 2) \cdot \delta + 40q_D \cdot |\mathcal{R}|^{-1/2}\end{aligned}$$

and $\mathcal{T}_{\mathcal{B}_2} \lesssim 3 \cdot \mathcal{T}_{\mathcal{A}} + O(q_G^2 + q_H^2 + q_H q_D) + O(q_G + q_H) \cdot (\mathcal{T}_{\mathsf{Enc}} + \mathcal{T}_{\mathsf{Rec}})$.

[18] Here, we implicitly use a property that P must be weakly $\log_2 |\mathcal{R}|$-spread if P is unique randomness recoverable.

B Using Our MRE-O2H Theorem Instead

We recall that G, H are oracles with domain X and codomain Y, S is a subset of X and G, H, S satisfy $G(x) = H(x)$ for all $x \notin S$.

Table 3. Comparison of MRM-O2H and MRE-O2H. Here $\mathcal{A}$ makes parallel queries to its oracle with query depth d. The $|S|$ denotes the number of elements in set S. The 1_S denotes the indicator function of set S, i.e., $1_S(x) = 1$ if $x \in S$ and 0 otherwise.

O2H theorem	Proved by	$\|S\|$	$\mathrm{Adv}(\mathcal{A}) \leq$	$\mathcal{B}_{\mathrm{ow}}$'s oracle access
MRM-O2H [25]	Measure-Rewind-Measure (MRM) technique [25]	Arbitrary	$4d \cdot \mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})$	H and G
MRE-O2H Th. 4	Measure-Rewind-Extract (MRE) technique	Arbitrary	$4\sqrt{d} \cdot \mathrm{Adv}(\mathcal{B}_{\mathrm{ow}})$	H, G and 1_S

The comparison of MRM-O2H theorem and our MRE-O2H theorem is shown in Table 3. It seems that our MRE-O2H theorem is more restrictive since it additionally requires the oracle access to 1_S. However, for the existing series of works used the MRM-O2H theorem [25,26,24,6,11,14], we observe that their proofs can also use our MRE-O2H theorem. In the following, we give a high-level explanation of this observation for each work in [25,26,24,6,11,14].

- Work [25]. The proof of Theorem 4.6 in this work uses the MRM-O2H theorem. In that proof, the set S is defined as $\{(m, c^*) : \mathsf{Encr}(\mathsf{pk}, m) = c^*\}$, where c^* is the challenge ciphertext and Encr is the encryption algorithm of the underlying deterministic PKE scheme. Obviously, the function 1_S can be computed by computing Encr, thus one can simulate the oracle 1_S by computing Encr. Therefore, the proof of Theorem 4.6 in [25] can also use our MRE-O2H theorem.

- Work [26]. The proof of Theorem 1 in this work uses the MRM-O2H theorem. In that proof, the set S is defined as $\{m : \mathsf{Enc}(pk, m) = c^*\}$, where c^* is the challenge ciphertext and Enc is the encryption algorithm of the underlying deterministic PKE scheme. Similar with the proof of Theorem 4.6 in [25], one can simulate the oracle 1_S by computing Enc, which means that the proof of Theorem 1 in [26] can also use our MRE-O2H theorem.

- Work [24]. The proof of Theorem 4.2 in this work uses the MRM-O2H theorem. In that proof, the set S is defined as $\{x : \exists j \in [i^*], \mathsf{dEnc}(\mathsf{pk}^{k_j}, x) = ct_j\}$, where ct_j is the challenge ciphertext and dEnc is the encryption algorithm of the underlying deterministic PKE scheme. Similar to the proof of Theorem 4.6 in [25], one can simulate the oracle 1_S by computing dEnc, which means that the proof of Theorem 4.2 in [24] can also use our MRE-O2H theorem.

- Work [6]. The proof of Theorem 2 in this work uses the MRM-O2H theorem. In that proof, the set S is defined as $\{(m, c^*) : \mathsf{Pco}(m, c^*) = 1\}$, where c^* is the challenge ciphertext and Pco is the plaintext checking oracle of the underlying PKE scheme. Since the underlying security considered in that proof is the $\mathsf{OW\text{-}qPCA}$ security and Pco can be quantum accessed in the $\mathsf{OW\text{-}qPCA}$ game, one can simulate the (quantum accessible) oracle 1_S by querying Pco. Therefore, the proof of Theorem 2 in [6] can also use our MRE-O2H theorem.

- Work [11]. The proof of Theorem 3 in this work uses the MRM-O2H theorem. In that proof, the set S is defined as $\{(c^*, r \star \mathsf{pk})\}$, where c^* is the challenge ciphertext, r is the randomness used to generate c^*, "$\star$" is the group action of the underlying group. In fact, in order to transform $\mathcal{B}_{\text{ow}}$, which is obtained by using the MRM-O2H theorem, into an algorithm that attacks the underlying hard problem, the proof of Theorem 3 introduces a (publicly quantum accessible) oracle $\mathsf{GA\text{-}DDH}_g$. We observe that this oracle can also be used to check whether an element (c_1, c_2) belongs to the set S, that is, if $c_1 = c^*$ and $\mathsf{GA\text{-}DDH}_g(c_1, c_2) = 1$, we have $(c_1, c_2) \in S$ and $(c_1, c_2) \notin S$ otherwise. This means that one can simulate the oracle 1_S by querying $\mathsf{GA\text{-}DDH}_g$. Hence, the proof of Theorem 3 in [11] can also use our MRE-O2H theorem.

- Work [14]. The proof of Theorem 3 in this work uses the MRM-O2H theorem. In that proof, the set S is defined as $\{m^*\}$, where m^* is the challenge plaintext uniformly sampled by the challenger. We note that the following classical event is introduced in that proof:

  $$E_{\mathsf{dPKE}} : m^* \overset{\$}{\leftarrow} \mathcal{M},\, \exists m \neq m^*,\, \mathsf{dEnc}_{pk}(m) = \mathsf{dEnc}_{pk}(m^*).$$

  Here dEnc is the encryption algorithm of the underlying deterministic PKE scheme, $\mathsf{dEnc}_{pk}(m^*)$ actually the challenge ciphertext c^*. Obviously, if the event E_{dPKE} does not occur, we can conclude that $\mathsf{dEnc}_{pk}(x) = c^*$ iff $x \in S$. Since [14] has proved that the probability $\Pr[E_{\mathsf{dPKE}}]$ is negligible, one can simulate the oracle 1_S by computing dEnc with an negligible error probability. Hence, the proof of Theorem 3 in [14] can also use our MRE-O2H theorem.

In summary, the proofs of [25,26,24,6,11,14] that used the MRM-O2H theorem can also use our MRE-O2H theorem.

C Why NTRU-based Encryption Is Usually Unique Randomness Recoverable

Here we roughly explain why NTRU-based PKE schemes generally satisfy the assumption of unique randomness recoverable.

Consider a typical NTRU-based PKE scheme $\mathsf{P} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as:

1. Gen, the key generation algorithm. Choose $\mathbf{f}', \mathbf{g} \leftarrow \mathbb{Z}_q[X]/\langle f(X)\rangle$ and compute $\mathbf{f} := p\mathbf{f}' + 1$, where p is a small integer relatively-prime to q. If $\mathbf{f}$ is not invertible in $\mathbb{Z}_q[X]/\langle f(X)\rangle$, restart. Otherwise, return $(pk, sk) := (p\mathbf{g}\mathbf{f}^{-1}, \mathbf{f})$.
2. Enc, the encryption algorithm. For a plaintext m belong to the message space $\mathcal{M}$, choose a randomness $\mathbf{r}$ from the randomness space $\mathcal{R}(\subseteq \mathbb{Z}_q[X]/\langle f(X)\rangle)$. Then, compute and return the ciphertext $c := \mathsf{Enc}_{pk}(m; \mathbf{r})$, where $\mathsf{Enc}_{pk}(m; \mathbf{r})$ $= p\mathbf{g}\mathbf{f}^{-1}\mathbf{r} + \mathsf{Encode}(m)$.

Since the assumption of unique randomness recoverable does not involve the decryption algorithm Dec, in the above, we omit the description of Dec.

As stated in [27], with a overwhelming probability, the element $\mathbf{g}$ that used to generate the public key pk is invertible in $\mathbb{Z}_q[X]/\langle f(X)\rangle$. Note that p is relatively-prime to q. Hence, without loss of generality, we can consider the public key $pk = p\mathbf{g}\mathbf{f}^{-1}$ of P to be invertible in $\mathbb{Z}_q[X]/\langle f(X)\rangle$ as well.

Now, for all plaintext $m \in \mathcal{M}$ and all randomness $\mathbf{r} \in \mathcal{R}$, we can construct the following recover algorithm Rec to recover $\mathbf{r}$ from the ciphertext c:

Rec: Given input $pk(= p\mathbf{g}\mathbf{f}^{-1})$, m and c, it works as:

1. Compute the value $\mathbf{r}' := (p\mathbf{g}\mathbf{f}^{-1})^{-1}(c - \mathsf{Encode}(m))$. If $\mathbf{r}' \in \mathcal{R}$, return $\mathbf{r}'$. Otherwise, return $\perp$. Here $\mathcal{R}$ is the randomness space of P.

Since $\mathsf{Enc}_{pk}(m; \mathbf{r}) = p\mathbf{g}\mathbf{f}^{-1}\mathbf{r} + \mathsf{Encode}(m)$, one can easily check that $\mathsf{Rec}(pk, m, c)$ must have output $\mathbf{r}$ if $c = \mathsf{Enc}_{pk}(m; \mathbf{r})$. According to the definition of unique randomness recoverable given in Definition 6, this means that P is unique randomness recoverable with the recover algorithm Rec.