Non-Adaptive One-Way to Hiding not only Implies Adaptive Quantum Reprogramming, but also Does Better *

Heming Liao[1,2], Jiangxia Ge[3(✉)], Rui Xue[1,2], and Xiaogang Zhou[3]

[1] State Key Laboratory of Cyberspace Security Defense, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100085, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
[3] China Telecom Quantum Information Technology Group Co., Ltd
{liaoheming,xuerui}@iie.ac.cn,{gejiangxia,zhouxiaogang}@chinatelecom.cn

Abstract. As three frequently used techniques for adaptive reprogramming in the QROM, the adaptive One-Way to Hiding (O2H) proposed by Unruh (CRYPTO 2014), the GHHM adaptive reprogramming proposed by Grilo et al. (ASIACRYPT 2021), and the Pan-Zeng adaptive reprogramming proposed by Pan and Zeng (PKC 2024), address different reprogramming scenarios, and do not appear to imply one another. Nevertheless, the recent result of Jaeger (ASIACRYPT 2025) reveals a surprising connection: all of these three adaptive techniques can be implied by a non-adaptive technique called Fixed-Permutation O2H (FP-O2H), which also yields tighter bounds for Unruh's adaptive O2H and Pan-Zeng adaptive reprogramming.

In this paper, we reconsider the implication between FP-O2H and GHHM adaptive reprogramming. We first introduce a variant of FP-O2H, called the Double-Oracle-Fixed-Permutation O2H (DOFP-O2H). Then, by applying this variant, we derive a tighter bound for the GHHM adaptive reprogramming. Thereby, our result complements Jaeger's findings by addressing the final piece, showing that the non-adaptive O2H not only implies adaptive reprogramming in the QROM, but also yields tighter bounds. In addition, a direct application of our tighter GHHM adaptive reprogramming yields a tighter EUF-CMA security proof of Fiat-Shamir transform in the QROM: the security loss with respect to the number of signing queries q_s decreases from $O(q_s)$ to $O(\sqrt{q_s})$.

Furthermore, we reconsider the implication between FP-O2H and the ABKM permutation resampling proposed by Alagic et al. (EUROCRYPT 2022). By applying our DOFP-O2H, we reprove the ABKM permutation resampling, and derive the same bound as that of Alagic et al. This improves Jaeger's result, and suggests that the FP-O2H not only can be applied to analyze the reprogramming in the QROM, but also has potential for analyzing reprogramming in the random permutation setting.

Keywords: quantum-accessible random oracle model · non-adaptive reprogramming · adaptive reprogramming · one-way to hiding

* The formal version of this paper appears at PKC 2026, this is the full version.

1 Introduction

Hash functions are widely used in modern cryptography owing to their efficiency and low computational overhead. For schemes that use hash functions, although security proofs in the standard model are generally preferred, they can be complex or inefficient. Therefore, the random oracle model (ROM) [4] remains a popular choice. It models the hash function as a public random oracle H, allowing for tighter and conceptually simpler security proofs. This model allows the use of two powerful techniques. First, Lazy-Sampling technique is used to simulate the oracle H and record adversarial queries. Second, the reprogramming technique can be applied to enable us to modify H's output at a specific point to a fresh random value, which facilitates the security proof.

In the quantum setting, since the adversary can locally compute the hash function in superposition, the ROM should be lifted to the quantum-accessible random oracle model (QROM) [6], where all parties have quantum access to the random oracle. However, completing security proofs in the QROM is usually more challenging than in the ROM, since the Lazy-Sampling and reprogramming techniques are no longer directly applicable in the quantum setting. To address this obstacle, a line of research [27,24,7,21,23,22,10,3,5,16,13,20,12] has proposed quantum analogues of these techniques. We describe the widely used versions as follows.

- **Quantum Lazy-Sampling.** Zhandry [27] introduced the compressed oracle technique, which enables perfect simulation of a quantum-accessible random oracle and allows recording of adversarial queries with only negligible error.
- **Non-adaptive reprogramming.** The One-Way to Hiding (O2H) theorem given by Unruh [23,3] provides a solution for non-adaptive reprogramming in the QROM. Roughly speaking, when the quantum-accessible random oracle is reprogrammed from G to H, this theorem shows that the distinguishing advantage is upper bounded by the probability that a one-way adversary outputs an element in the set $\{x : \mathsf{G}(x) \neq \mathsf{H}(x)\}$. Later, a series of "Double-Oracle" variants were proposed, including the DS-O2H theorem [5], the MRM-O2H theorem [16], and the MRE-O2H theorem [12]. By allowing the one-way adversary to query both G and H, these variants reduce the loss associated with the number of random oracle queries.
- **Unruh's adaptive O2H.** To address adaptive reprogramming in the QROM, Unruh [21,22] proposed two variants of the original O2H theorem. Both variants allow the adversary to adaptively choose the reprogramming point, but restrict only one single reprogramming at one point with high entropy.
- **GHHM adaptive reprogramming.** Grilo et al. [13] proposed the GHHM adaptive reprogramming framework, which allows an arbitrary number of reprogrammings, but the reprogramming points adaptively chosen by the adversary are still required to have high entropy.
- **Pan-Zeng adaptive reprogramming.** The Pan-Zeng adaptive reprogramming framework proposed by Pan and Zeng [20] also allows an arbitrary number of reprogrammings, and it removes the high-entropy requirement for the

reprogramming points. The key idea is to link the adaptive chosen of these points to computationally hard problems, thereby bounding the distinguishing advantage by the hardness of those problems.

It can be observed that the techniques for adaptive reprogramming in the QROM are diverse, with each suited for different scenarios. Even though the Pan-Zeng framework seems the most general one, Jaeger [14] pointed out that it does not seem to directly imply or be implied by the other three techniques. Hence, when dealing with adaptive reprogramming problems in the QROM, it seems inevitable to apply these techniques case-by-case.

However, Jaeger's recent work [14] contradicts the above intuition. He introduced the Fixed-Permutation O2H (FP-O2H) theorem, a new variant of the original O2H [23,3], and proved that it implies Unruh's adaptive O2H, the GHHM adaptive reprogramming, and the Pan-Zeng adaptive reprogramming. This is counterintuitive, since FP-O2H theorem itself is only non-adaptive: the permutations P_0 and P_1 it focuses on are fixed, and thus the set of reprogramming points $\{x : \mathsf{P}_0(x) \neq \mathsf{P}_1(x)\}$ is also fixed at the beginning. Next, we summarize the implications proved by Jaeger by using his FP-O2H theorem.

- **FP-O2H implies Unruh's adaptive O2H.** For Unruh's first adaptive O2H theorem [21], Jaeger provided a new proof which tightened the bound from "$q_0 \cdot 2^{-l/2}$" to "$\sqrt{q_0} \cdot 2^{-l/2}$". For Unruh's second adaptive O2H theorem [22], Jaeger reproved it and tightened the bound by replacing collision entropy with min-entropy.
- **FP-O2H implies GHHM adaptive reprogramming.** Jaeger reproved the GHHM adaptive reprogramming theorem and obtained a new bound $\sqrt{r \cdot \sum_{i=1}^{r} \hat{q}_i \cdot p^i_{\max}}$, where r denotes the total number of reprogrammings, $\hat{q}_i$ is the total oracle queries before the i-th reprogramming, and $p^i_{\max}$ is a non-negative parameter related to the distribution of the i-th reprogramming point. Compared to the original bound $\sum_{i=1}^{r} \sqrt{\hat{q}_i \cdot p^i_{\max}}$ in [13], it is clear from the Cauchy-Schwarz inequality that the new bound given by Jaeger is looser.
- **FP-O2H implies Pan-Zeng adaptive reprogramming.** Jaeger reproved the Pan-Zeng adaptive reprogramming theorem, and achieved a new bound which is hard to compare to the original one given in [20]. Nevertheless, as computed by Jaeger, under the practical scenario that considered the security proof of [20] in the QROM, the new bound is tighter.

However, one limitation of Jaeger's result is that it does not provide a full justification for the superiority of non-adaptive O2H. While applying FP-O2H theorem derives tighter bounds for Unruh's first and second adaptive O2H theorems, as well as the Pan-Zeng adaptive reprogramming theorem (at least in specific practical scenarios), it does not improve the bound for the GHHM adaptive reprogramming theorem. This naturally leads to the following question:

Can we prove a tighter bound for the GHHM adaptive reprogramming theorem by applying the FP-O2H theorem?

1.1 Our Contributions

In this paper, we give an affirmative answer to the above question. By applying the FP-O2H theorem, we present a proof of the GHHM adaptive reprogramming theorem and derive a new bound $\sqrt{\sum_{i=1}^{r} \hat{q}_i \cdot p^i_{\max}}$, which is tighter than the original bound $\sum_{i=1}^{r} \sqrt{\hat{q}_i \cdot p^i_{\max}}$ given in [13][4]. Hence, it can be concluded that non-adaptive O2H not only implies adaptive quantum reprogramming but also does better. Note that the GHHM adaptive reprogramming theorem is a widely-used tool for QROM security proofs involving adaptive reprogramming [13,1,9,8,15,18,25,26,11]. Consequently, our new bound offers the potential for tighter security analyses for such proofs. For instance, applying our new bound of the GHHM adaptive reprogramming theorem in the proof of [13, Theorem 3] leads to a tighter EUF-CMA security proof for the Fiat-Shamir transform in the QROM, in which the security loss related to query times q_s to the signing oracle is reduced from $O(q_s)$ to $O(\sqrt{q_s})$.

In addition, besides its application to the reprogramming in the QROM, the FP-O2H theorem can also be used to analyze random permutation reprogramming. Following the proof idea of [14, Weakened Theorem 10], we prove that the ABKM permutation resampling theorem proposed by Alagic et al. [1] is implied by the FP-O2H theorem. Our resulting bound is tighter than that in [14, Weakened Theorem 10] and matches the tightness of the original ABKM bound.

1.2 Technical Overview

Our core tool is a Double-Oracle variant of the FP-O2H theorem as shown in the following Theorem 1, which we call it Double-Oracle-Fixed-Permutation O2H (DOFP-O2H) theorem. In contrast to the original FP-O2H theorem (Theorem 3) where the distinguishing advantage is bounded by the success probability of one one-way adversary, our DOFP-O2H theorem introduces two one-way adversaries, $\mathcal{B}^{\mathsf{P}_0}_{\mathsf{ow}}$ and $\mathcal{B}^{\mathsf{P}_1}_{\mathsf{ow}}$, and thereby reduce the loss from $O(q)$ to $O(\sqrt{q})$.

Theorem 1 (DOFP-O2H, informal). *Let $\mathsf{P}_0, \mathsf{P}_1$ be two fixed permutations. $\mathcal{A}$ is a quantum oracle algorithm making q queries. Then we have*

$$\left|\Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_0}] - \Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_1}]\right| \leq \sqrt{q \cdot \Pr[\mathsf{P}_0(x) \neq \mathsf{P}_1(x) : x \leftarrow \mathcal{B}^{\mathsf{P}_0}_{\mathsf{ow}}]} + \sqrt{q \cdot \Pr[\mathsf{P}_0(x) \neq \mathsf{P}_1(x) : x \leftarrow \mathcal{B}^{\mathsf{P}_1}_{\mathsf{ow}}]}.$$

Here $\mathcal{B}^{\mathcal{O}}_{\mathsf{ow}}$ ($\mathcal{O} \in \{\mathsf{P}_0, \mathsf{P}_1\}$) chooses $i \leftarrow_{\$} [q]$, runs $\mathcal{A}^{\mathcal{O}}$ until (just before) its i-th query, measure the query register and return the measurement outcome x.

[4] One may be curious about this result because in [13] it is claimed that the GHHM reprogramming theorem is tight, and a matching attack is given. However, our tighter bound does not conflict with that matching attack, and a detailed disscusion is provided in Supplementary Material C.

Intuitively, since the bound of $\mathcal{A}$'s distinguishing advantage depends on both P_0 and P_1, our DOFP-O2H theorem only completes "half-step-reprogramming". This is different from the original FP-O2H theorem, where the bound only depends on either P_0 or P_1 and thus the "whole-step-reprogramming" is completed. Hence, when the reprogramming points are computationally hidden (e.g., in the Pan-Zeng framework), our DOFP-O2H theorem seems not applicable, because in general it is hard to perfectly convert both one-way adversaries, $\mathcal{B}_{\mathsf{ow}}^{\mathsf{P}_0}$ and $\mathcal{B}_{\mathsf{ow}}^{\mathsf{P}_1}$, into an algorithm that solves the underlying hard problem. On the other hand, our DOFP-O2H theorem is well-suited for scenarios involving statistically hidden reprogramming points. Notice that statistically hidden reprogramming points correspond to statistically hard problems, where the adversary can be unbounded and can receive the truth table of P_0 and P_1. This allows us to bound the success probabilities of $\mathcal{B}_{\mathsf{ow}}^{\mathsf{P}_0}$ and $\mathcal{B}_{\mathsf{ow}}^{\mathsf{P}_1}$ using specific parameters, and hence by applying our DOFP-O2H theorem, we can bound the distinguishing advantage of $\mathcal{A}$ with a loss of only $O(\sqrt{q})$. Notably, both our tighter bound for the GHHM adaptive reprogramming theorem and our new proof for the ABKM permutation resampling theorem rely on the condition that the reprogramming points are statistically hidden, combined with the proof strategy of [14, Theorem 6].

In the following, we first briefly explain why we consider the Double-Oracle variant of the original FP-O2H theorem, and then outline how we prove it.

Why Double-Oracle. As we mentioned earlier, there are three Double-Oracle variants of the original O2H theorem [23,3], including the DS-O2H theorem [5], the MRM-O2H theorem [16], and the MRE-O2H theorem [12]. These variants all have tighter bounds by allowing the one-way adversary to query both functions G and H. This naturally leads to the idea that if we allow the one-way adversary to query both permutations P_0 and P_1, we may be able to prove a tighter Double-Oracle variant of the FP-O2H theorem. This, in turn, could be used to achieve tighter bounds for reprogramming theorems with statistically hidden points, such as the GHHM adaptive reprogramming theorem.

How Double-Oracle. Unfortunately, the proof strategies of [5,16,12] cannot be directly applied to the FP-O2H theorem. This is because their proofs all rely on the following property:

Denote X *(resp.* Y*) as the register defined over* $\{0,1\}^m$ *(resp.* $\{0,1\}^n$*). Then for any function* $\mathsf{H}:\{0,1\}^m\to\{0,1\}^n$ *and any projector* J *on* X*,* $\mathrm{O}_{\mathsf{H}}\circ\mathrm{J}=\mathrm{J}\circ\mathrm{O}_{\mathsf{H}}$*, where* $\mathrm{O}_{\mathsf{H}}:|x,y\rangle_{\mathsf{XY}}\mapsto|x,y\oplus\mathsf{H}(x)\rangle_{\mathsf{XY}}$.

However, this commutativity no longer holds when we consider the unitary operation that computes an arbitrary permutation in the in-place manner. Specifically, the unitary $\mathrm{O}_{\mathsf{P}}:|x\rangle_{\mathsf{X}}\mapsto|\mathsf{P}(x)\rangle_{\mathsf{X}}$ and a projector J on X do not always commute, meaning $\mathrm{O}_{\mathsf{P}}\circ\mathrm{J}\neq\mathrm{J}\circ\mathrm{O}_{\mathsf{P}}$ for an arbitrary permutation P on X. Consequently, to design the Double-Oracle variant of the FP-O2H theorem, we need a new proof strategy. In what follows, we first introduce our new proof strategy following the notational convention of [5,16,12], and then explain how to extend it to the permutation setting.

We first consider a simple case where the adversary $\mathcal{A}$ queries G or H two times, and G and H differ only at point m^*. Let G (resp. H) be simulated by $\mathrm{O}_{\mathsf{G}} : |x,y\rangle_{\mathsf{XY}} \mapsto |x, y \oplus \mathsf{G}(x)\rangle_{\mathsf{XY}}$ (resp. $\mathrm{O}_{\mathsf{H}} : |x,y\rangle_{\mathsf{XY}} \mapsto |x, y \oplus \mathsf{H}(x)\rangle_{\mathsf{XY}}$). Denote I as the identity operation, $|\psi\rangle$ as the initial state of $\mathcal{A}$, and U_1 and U_2 as the unitary operations performed by $\mathcal{A}$ between its oracle queries. Define $\mathrm{J}_{m^*} := |m^*\rangle\langle m^*|$, which is a projector on the oracle's input register X. Then, define states

$$\begin{aligned}
|\Psi_{\mathsf{G}}\rangle :=& \mathrm{U}_2 \circ \mathrm{O}_{\mathsf{G}} \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} |\psi\rangle, \\
|\Psi_{\mathsf{H}}\rangle :=& \mathrm{U}_2 \circ \mathrm{O}_{\mathsf{H}} \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{H}} |\psi\rangle, \\
|\Psi_0\rangle :=& \mathrm{U}_2 \circ \mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) |\psi\rangle, \\
\overset{(a)}{=}& \mathrm{U}_2 \circ \mathrm{O}_{\mathsf{H}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{H}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) |\psi\rangle.
\end{aligned}$$

Here (a) follows from the property that $\mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) = \mathrm{O}_{\mathsf{H}} \circ (\mathrm{I} - \mathrm{J}_{m^*})$. At this point, the distinguishing advantage of $\mathcal{A}$ can be computed as follows:

$$\begin{aligned}
\mathrm{Adv}(\mathcal{A}) &= \left|\Pr[1 \leftarrow \mathcal{A}^{\mathsf{G}}] - \Pr[1 \leftarrow \mathcal{A}^{\mathsf{H}}]\right| \\
&\leq \||\Psi_{\mathsf{G}}\rangle - |\Psi_{\mathsf{H}}\rangle\| && \text{(By Lemma 4)} \qquad (1) \\
&\leq \||\Psi_{\mathsf{G}}\rangle - |\Psi_0\rangle\| + \||\Psi_{\mathsf{H}}\rangle - |\Psi_0\rangle\|. && \text{(By triangle inequality)}
\end{aligned}$$

For the first norm $\||\Psi_{\mathsf{G}}\rangle - |\Psi_0\rangle\|$, by substituting $|\Psi_0\rangle$ into $\mathrm{U}_2 \circ \mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) |\psi\rangle$, we can compute

$$\begin{aligned}
&\||\Psi_{\mathsf{G}}\rangle - |\Psi_0\rangle\|^2 \\
=& \|\mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} |\psi\rangle - (\mathrm{I} - \mathrm{J}_{m^*}) \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) |\psi\rangle\|^2 \\
=& \|(\mathrm{J}_{m^*} + \mathrm{I} - \mathrm{J}_{m^*}) \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} |\psi\rangle - (\mathrm{I} - \mathrm{J}_{m^*}) \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) |\psi\rangle\|^2 \\
\overset{(a)}{=}& \|\mathrm{J}_{m^*} \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} |\psi\rangle\|^2 + \|(\mathrm{I} - \mathrm{J}_{m^*}) \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} \circ \mathrm{J}_{m^*} |\psi\rangle\|^2 \\
\leq& \|\mathrm{J}_{m^*} \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{G}} |\psi\rangle\|^2 + \|\mathrm{J}_{m^*} |\psi\rangle\|^2 \\
\overset{(b)}{=}& 2 \cdot \Pr[x = m^* : x \leftarrow \mathcal{B}^{\mathsf{G}}_{\mathsf{ow}}].
\end{aligned}$$

Here (a) uses the property that $\||\phi_1\rangle + |\phi_2\rangle\|^2 = \||\phi_1\rangle\|^2 + \||\phi_2\rangle\|^2$ for any orthogonal states $|\phi_1\rangle$ and $|\phi_2\rangle$. In (b), $\mathcal{B}^{\mathsf{G}}_{\mathsf{ow}}$ is the one-way adversary that first chooses $i \leftarrow_{\$} [2]$, runs $\mathcal{A}^{\mathsf{G}}$ until (just before) its i-th query, measure the input register and return the measurement outcome x. Therefore (b) holds by expressing the success probability of $\mathcal{B}^{\mathsf{G}}_{\mathsf{ow}}$. Hence, we have $\||\Psi_{\mathsf{G}}\rangle - |\Psi_0\rangle\| \leq \sqrt{2 \cdot \Pr[x = m^* : x \leftarrow \mathcal{B}^{\mathsf{G}}_{\mathsf{ow}}]}$. Next, similar to the above computation, by substituting $|\Psi_0\rangle$ into $\mathrm{U}_2 \circ \mathrm{O}_{\mathsf{H}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{H}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) |\psi\rangle$, we have $\||\Psi_{\mathsf{H}}\rangle - |\Psi_0\rangle\| \leq \sqrt{2 \cdot \Pr[x = m^* : x \leftarrow \mathcal{B}^{\mathsf{H}}_{\mathsf{ow}}]}$. Combining these two inequalities with Eq. (1), we can obtain

$$\begin{aligned}
\left|\Pr[1 \leftarrow \mathcal{A}^{\mathsf{G}}] - \Pr[1 \leftarrow \mathcal{A}^{\mathsf{H}}]\right| \leq& \sqrt{2 \cdot \Pr[x = m^* : x \leftarrow \mathcal{B}^{\mathsf{G}}_{\mathsf{ow}}]} \\
&+ \sqrt{2 \cdot \Pr[x = m^* : x \leftarrow \mathcal{B}^{\mathsf{H}}_{\mathsf{ow}}]}.
\end{aligned} \qquad (2)$$

The above proof strategy does not utilize the commutativity of J_{m^*} with O_{G} or O_{H}. Instead, it only relies on the property that $\mathrm{O}_{\mathsf{G}} \circ (\mathrm{I} - \mathrm{J}_{m^*}) = \mathrm{O}_{\mathsf{H}} \circ (\mathrm{I} - \mathrm{J}_{m^*})$.

This property extends naturally to the permutation setting. Specifically, for any permutations P_0 and P_1 on $\{0,1\}^m$ which differ only at a set $S \subseteq \{0,1\}^m$, we have $\mathrm{O}_{\mathsf{P}_0} \circ (\mathrm{I} - \mathrm{J}_S) = \mathrm{O}_{\mathsf{P}_1} \circ (\mathrm{I} - \mathrm{J}_S)$ where $\mathrm{O}_{\mathsf{P}_0} : |x\rangle_{\mathsf{X}} \mapsto |\mathsf{P}_0(x)\rangle_{\mathsf{X}}$, $\mathrm{O}_{\mathsf{P}_1} : |x\rangle_{\mathsf{X}} \mapsto |\mathsf{P}_1(x)\rangle_{\mathsf{X}}$, and $\mathrm{J}_S := \sum_{x\in S} |x\rangle\langle x|$ is a projector on the register X. Hence, the above proof strategy extends directly to the permutation setting to obtain

$$\begin{aligned}\left|\Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_0}] - \Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_1}]\right| \leq & \sqrt{2 \cdot \Pr[\mathsf{P}_0(x) \neq \mathsf{P}_1(x) : x \leftarrow \mathcal{B}_{\mathsf{ow}}^{\mathsf{P}_0}]} \\ & + \sqrt{2 \cdot \Pr[\mathsf{P}_0(x) \neq \mathsf{P}_1(x) : x \leftarrow \mathcal{B}_{\mathsf{ow}}^{\mathsf{P}_1}]},\end{aligned}$$

where P_0 is simulated by $\mathrm{O}_{\mathsf{P}_0}$ and P_1 is simulated by $\mathrm{O}_{\mathsf{P}_1}$. This completes the proof of Theorem 1 for $q = 2$. The general case $q > 2$ follows by a straightforward recursive extension.

Remark 1. In our formal DOFP-O2H theorem. i.e., Theorem 5, there is an extra factor $\sqrt{2}$ in the upper bound. The reason is that, similar to the FP-O2H theorem, we also compute the upper bound for $|\sqrt{\Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_0}]} - \sqrt{\Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_1}]}|$ in Theorem 5, and the factor $\sqrt{2}$ arises exactly here. To ensure consistency, we also add the factor $\sqrt{2}$ into the upper bound for $|\Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_0}] - \Pr[1 \leftarrow \mathcal{A}^{\mathsf{P}_1}]|$. Moreover, although we can directly prove our Theorem 5 by applying the above analysis into the permutation setting, to maintain consistency with Jaeger, in this paper we choose to first prove Theorem 4 which generalizes Eq. (2) to an arbitrary q, and subsequently prove that Theorem 4 and Theorem 5 implies each other (with a constant factor).

1.3 Related Work

Besides bridging non-adaptive O2H and adaptive reprogramming, FP-O2H theorem also provides a new perspective for analyzing quantum oracles. Traditional O2H theorems [23,3,5,16,12] focus only on the function-like oracle $\mathcal{O} : |x, y\rangle \mapsto |x, y \oplus \mathsf{H}(x)\rangle$ associated with a function H, whereas the FP-O2H theorem focuses on a more general permutation-like oracle $\mathcal{O} : |x\rangle \mapsto |\mathsf{P}(x)\rangle$ associated with a permutation P[5]. This generalization may enable us to handle certain scenarios that are challenging by using the traditional O2H theorems. A typical example is the analyzing of the game hop from $\mathbf{G_{14}}$ to $\mathbf{G_{15}}$ in [17], which first formulates the quantum decryption oracle qDec as a permutation-like oracle and then applies the FP-O2H theorem to complete the proof. Additionally, we find that we cannot apply our DOFP-O2H theorem in that game hop to get a tighter bound, as the reprogramming points are only computationally hidden.

Furthermore, although we have proven that the DOFP-O2H theorem implies the ABKM permutation resampling theorem, the application of DOFP-O2H (also FP-O2H) theorem in the random permutation reprogramming remains limited, since we do not know whether the quantum-accessible random permutation

[5] Here we claim that the permutation-like oracle is more general, since the generic function-like oracle on input/output register X/Y is a specific instance of the in-place permutation-like oracle acting on the joint register XY.

can be sparsely represented. As a result, similar to the derivation given in [14, Section 6], the stronger version of the ABKM permutation resampling theorem [2] is hard to reprove by using our DOFP-O2H theorem. However, the surprising work of Carolan [7] very recently provided a sparse representation of the quantum-accessible random permutation. Hence, it is very interesting whether combining DOFP-O2H theorem with Carolan's sparse representation technique could imply the stronger version of the ABKM permutation resampling theorem.

In addition, as mentioned in Section 1.1, our tighter GHHM adaptive reprogramming theorem directly yields a tighter EUF-CMA security proof in the QROM for the Fiat-Shamir transform. Notably, as presented in [11], the EUF-CMA security proof of the Fiat-Shamir-with-Aborts (FSwA) signature paradigm in the QROM also relies on the GHHM reprogramming theorem. Consequently, there remains an interesting question whether a tighter security bound for FSwA can be obtained by combining the proof in [11] with our tighter GHHM adaptive reprogramming theorem. We leave it as future work.

2 Preliminaries

2.1 Notation

Let n be a positive integer and denote $[n]$ as the set $\{1, 2, \ldots, n\}$. Let $[\![P]\!]$ denote a bit which is 1 if the predicate P is true and otherwise 0. $x \leftarrow_{\$} S$ denotes uniformly sampling an element x from a finite set S. $x \leftarrow \mathfrak{D}$ means that x is chosen according to distribution $\mathfrak{D}$. $\mathrm{Fcs}[m, n]$ refers to the set containing all functions from $\{0,1\}^m$ to $\{0,1\}^n$. Let $\mathsf{H}[x \mapsto y]$ represent the function identical to H except that $\mathsf{H}[x \mapsto y](x) = y$. $\Pr[A : B]$ is the probability that predicate A keeps true where all variables in A are conditioned according to predicate B. We use $\Pr[\mathbf{G}]$ to denote the probability that game $\mathbf{G}$ finally returns 1.

2.2 Compressed Oracle Technique

We present some fundamental quantum notions and principles utilized in this paper in Supplementary Material A. For a more thorough introduction to the foundational concepts of quantum computation, we refer to a textbook [19].

In what follows, we briefly introduce the compressed oracle technique proposed by Zhandry [27], which can be used to perfectly simulate a quantum-accessible random oracle $\mathcal{O}_{\mathsf{H}} : |x, y\rangle \mapsto |x, y \oplus \mathsf{H}(x)\rangle$ with $\mathsf{H} \leftarrow_{\$} \mathrm{Fcs}[m, n]$. In this paper, since we only consider hard problems in the information-theoretic setting, it is unnecessary to utilize the efficient version of the compressed oracle. Hence, we only introduce the inefficient versions here, including StO', StO, PhO, and $\mathsf{FourierPhsO}$.

Impurified Standard Oracle StO'. Introduce registers $\mathsf{H}_{0^m} \ldots \mathsf{H}_{1^m}$ and set their initial state as $|\mathsf{H}(0^m)\rangle_{\mathsf{H}_{0^m}} \ldots |\mathsf{H}(1^m)\rangle_{\mathsf{H}_{1^m}}$ to store the truth table of H,

where $\mathsf{H} \leftarrow_{\$} \mathrm{Fcs}[m,n]$ is a random function. Then, a quantum query is implemented as

$$\mathsf{StO}' : |x,y\rangle|\mathsf{H}(0^m)\rangle_{\mathsf{H}_{0^m}} \dots |\mathsf{H}(1^m)\rangle_{\mathsf{H}_{1^m}} \mapsto |x, y \oplus \mathsf{H}(x)\rangle|\mathsf{H}(0^m)\rangle_{\mathsf{H}_{0^m}} \dots |\mathsf{H}(1^m)\rangle_{\mathsf{H}_{1^m}}.$$

Here, $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ are viewed as the internal registers of StO'. StO' is perfectly indistinguishable from $\mathcal{O}_{\mathsf{H}}$.

Purified Standard Oracle StO. Introduce registers $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ and set their initial state as $|\beta_{0^n}\rangle_{\mathsf{H}_{0^m}} \dots |\beta_{0^n}\rangle_{\mathsf{H}_{1^m}}$, where $|\beta_{0^n}\rangle := \sum_{y\in\{0,1\}^n} 1/\sqrt{2^n}|y\rangle$. Then, a quantum query is implemented as

$$\mathsf{StO} : |x,y\rangle|y_{0^m}\rangle_{\mathsf{H}_{0^m}} \dots |y_{1^m}\rangle_{\mathsf{H}_{1^m}} \mapsto |x, y \oplus y_x\rangle|y_{0^m}\rangle_{\mathsf{H}_{0^m}} \dots |y_{1^m}\rangle_{\mathsf{H}_{1^m}}.$$

Here, $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ are viewed as the internal registers of StO.

Lemma 1 ([27, Lemma 2]). *StO' and StO are perfectly indistinguishable. That is, for any quantum adversary $\mathcal{A}$, $\Pr[1 \leftarrow \mathcal{A}^{\mathsf{StO}'}] = \Pr[1 \leftarrow \mathcal{A}^{\mathsf{StO}}]$.*

Phase Oracle PhO. Introduce registers $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ and set their initial state as $|\beta_{0^n}\rangle_{\mathsf{H}_{0^m}} \dots |\beta_{0^n}\rangle_{\mathsf{H}_{1^m}}$, where $|\beta_{0^n}\rangle := \sum_{y\in\{0,1\}^n} 1/\sqrt{2^n}|y\rangle$. Then, a quantum query is implemented as

$$\mathsf{PhO} : |x,y\rangle|y_{0^m}\rangle_{\mathsf{H}_{0^m}} \dots |y_{1^m}\rangle_{\mathsf{H}_{1^m}} \mapsto (-1)^{y\cdot y_x}|x,y\rangle|y_{0^m}\rangle_{\mathsf{H}_{0^m}} \dots |y_{1^m}\rangle_{\mathsf{H}_{1^m}}.$$

Here, $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ are viewed as the internal registers of PhO.

Lemma 2 ([27, Lemma 3]). *StO and $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{PhO} \circ \mathsf{H}^{\mathsf{Y}}$ are perfectly indistinguishable. That is, for any quantum adversary $\mathcal{A}$, we have $\Pr[1 \leftarrow \mathcal{A}^{\mathsf{StO}}] = \Pr[1 \leftarrow \mathcal{A}^{\mathsf{H}^{\mathsf{Y}} \circ \mathsf{PhO} \circ \mathsf{H}^{\mathsf{Y}}}]$. Here, $\mathsf{H}^{\mathsf{Y}} : |x\rangle \mapsto \sum_{y\in\{0,1\}^n} (-1)^{y\cdot x}/\sqrt{2^n}|y\rangle$ denotes the Hadamard operation applied to the output register Y of PhO.*

Fourier Phase Oracle $\mathsf{FourierPhsO}$. Introduce registers $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ and set their initial state as $|0^n\rangle_{\mathsf{H}_{0^m}} \dots |0^n\rangle_{\mathsf{H}_{1^m}}$. Then, a quantum query is executed as

$$\begin{aligned}\mathsf{FourierPhsO} : |x,y\rangle|y_{0^m}\rangle_{\mathsf{H}_{0^m}} \dots |y_{1^m}\rangle_{\mathsf{H}_{1^m}} \\ \mapsto |x,y\rangle|y_{0^m}\rangle_{\mathsf{H}_{0^m}} \dots |y_x \oplus y\rangle_{\mathsf{H}_x} \dots |y_{1^m}\rangle_{\mathsf{H}_{1^m}}.\end{aligned}$$

Here, $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ are viewed as the internal registers of $\mathsf{FourierPhsO}$.

Lemma 3 ([27, Lemma 23]). *PhO and $\mathsf{FourierPhsO}$ are perfectly indistinguishable, i.e., for any adversary $\mathcal{A}$, $\Pr[1 \leftarrow \mathcal{A}^{\mathsf{PhO}}] = \Pr[1 \leftarrow \mathcal{A}^{\mathsf{FourierPhsO}}]$.*

One feature of $\mathsf{FourierPhsO}$ is that the basis state of the registers $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ has a sparse representation, i.e., if $\mathsf{FourierPhsO}$ is queried at most q times, then at most q of these 2^m registers are in the basis state $|y\rangle$ where $y \neq 0^n$. Hence, $\mathsf{FourierPhsO}$ is also called the sparse representation of the quantum-accessible random oracle. Intuitively speaking, the existence of a state $|y\rangle_{\mathsf{H}_x}$ where $y \neq 0^n$ indicates that the adversary must have queried $\mathsf{FourierPhsO}$ on input x. Now, by combining Lemma 1, 2, and 3, we obtain the following corollary.

Corollary 1. StO' *and* $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$ *are perfectly indistinguishable. That is, for any adversary* $\mathcal{A}$, $\Pr[1 \leftarrow \mathcal{A}^{\mathsf{StO}'}] = \Pr[1 \leftarrow \mathcal{A}^{\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}}]$. *Here,* $\mathsf{H}^{\mathsf{Y}} : |y\rangle \mapsto \sum_{z \in \{0,1\}^n} (-1)^{z \cdot y} / \sqrt{2^n} |z\rangle$ *denotes the Hadamard operation on the output register* Y *of* $\mathsf{FourierPhsO}$.

We now present the following corollary, which will be applied in our subsequent analysis. It demonstrates that if the quantum-accessible random oracle is simulated by $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$, the reprogramming procedure can be achieved by directly modifying the state of registers $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$, without firstly choosing the reprogramming value y.

$\mathsf{Reprogram}_0(x)$	$\mathsf{Reprogram}_1(x)$
1. $y \leftarrow_{\$} \{0,1\}^n$	1. Transform $\|*\rangle_{\mathsf{H}_x}$ into $\|0^n\rangle_{\mathsf{H}_x}$
2. Transform $\|*\rangle_{\mathsf{H}_x}$ into $\|y\rangle_{\mathsf{H}_x}$	

Fig. 1. Two reprogramming oracles in Corollary 2. Both oracles employ the registers $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ as work registers. The "Transform" step in both oracles is straightforward to implement using basic quantum operations.

Corollary 2. *Let* $\mathsf{Reprogram}_b, b \in \{0,1\}$ *be two reprogramming oracles defined in Fig. 1. Then, for any adversary* $\mathcal{A}$, *it follows that:*

$$\Pr\left[1 \leftarrow \mathcal{A}^{\mathsf{StO}', \mathsf{Reprogram}_0}\right] = \Pr\left[1 \leftarrow \mathcal{A}^{\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}, \mathsf{Reprogram}_1}\right].$$

Proof. By Corollary 1, StO' and $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$ are perfectly indistinguishable, thus, we focus on the difference between $\mathsf{Reprogram}_0$ and $\mathsf{Reprogram}_1$. Note that $\mathsf{Reprogram}_0$ reprograms $\mathsf{H}(x)$ to y, where y is a fresh random value uniformly chosen from $\{0,1\}^n$. For any subsequent query $|x, y'\rangle$ to StO', $\mathcal{A}$ will receive $|x, y' \oplus y\rangle$. Indeed, $\mathsf{Reprogram}_1$ combined with the $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$ oracle produces an identical effect, as demonstrated below.

Suppose that $\mathcal{A}$ has queried $\mathsf{Reprogram}_1$ using x. At this point, the state of H_x is $|0^n\rangle$, and the simulation for the subsequent query $|x, y'\rangle$ to $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$ will be simulated as:

$$\begin{aligned}
|x, y'\rangle|0^n\rangle_{\mathsf{H}_x} &\xrightarrow{\mathsf{H}^{\mathsf{Y}}} \frac{1}{\sqrt{2^n}} \sum_z (-1)^{y' \cdot z} |x, z\rangle|0^n\rangle_{\mathsf{H}_x} \\
&\xrightarrow{\mathsf{FourierPhsO}} \frac{1}{\sqrt{2^n}} \sum_z (-1)^{y' \cdot z} |x, z\rangle|z\rangle_{\mathsf{H}_x} \\
&\xrightarrow{\mathsf{H}^{\mathsf{Y}}} \frac{1}{2^n} \sum_z \sum_k (-1)^{y' \cdot z} (-1)^{k \cdot z} |x, k\rangle|z\rangle_{\mathsf{H}_x} \\
&= \frac{1}{\sqrt{2^n}} \sum_k |x, k\rangle \mathsf{H}^{\mathsf{H}_x} |y' \oplus k\rangle_{\mathsf{H}_x} = \frac{1}{\sqrt{2^n}} \sum_k |x, y' \oplus k\rangle \mathsf{H}^{\mathsf{H}_x} |k\rangle_{\mathsf{H}_x}.
\end{aligned}$$

Here, for simplicity, we only write out the register H_x and use $\mathsf{H}^{\mathsf{H}_\times}$ to denote the Hadamard operation on H_x. Clearly, the final state of the above equation is a purification of the state $|x, y' \oplus y\rangle$ with a uniformly random y, which is obtained by querying $\mathsf{Reprogram}_0$ and StO'. This implies that querying $\mathsf{Reprogram}_0$ and StO' is equivalent to querying $\mathsf{Reprogram}_1$ and $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$. This completes the proof of Corollary 2. □

2.3 One-Way to Hiding

To formalize our discussion of the One-Way to Hiding (O2H) theorem, we first introduce the necessary notations. Here, similar to [14], we only consider one type of oracle $\mathcal{O}$ which quantumly computes a specific permutation P, i.e., $\mathcal{O}$ maps $|w\rangle$ to $|\mathsf{P}(w)\rangle$. For the sake of simplicity, in this paper we sometimes directly write $\mathcal{O}$ as P. Meanwhile, the traditional function-like oracle $\mathcal{O}_{\mathsf{H}} : |x, y\rangle \mapsto |x, y \oplus \mathsf{H}(x)\rangle$ can be viewed as computing the permutation $\mathsf{H}[\oplus]$ quantumly, where $\mathsf{H}[\oplus](x, y) = (x, y \oplus \mathsf{H}(x))$.

Game $\mathbf{G}^{\mathsf{hide}}_{\mathcal{O}}(\mathcal{A})$ $\mathcal{O} \in \{\mathsf{P}_0, \mathsf{P}_1\}$	**Game** $\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathcal{O}})$ $\mathcal{O} \in \{\mathsf{P}_0, \mathsf{P}_1\}$
1. $b \leftarrow \mathcal{A}^{\mathcal{O}}$	1. $i \leftarrow_{\$} [q]$
2. **return** $[\![b = 1]\!]$	2. run $\mathcal{A}^{\mathcal{O}}$ until (just before) its i-th query
	3. measure the query register to get the outcome w
	4. **return** $[\![\mathsf{P}_0(w) \neq \mathsf{P}_1(w)]\!]$

Fig. 2. Hiding game $\mathbf{G}^{\mathsf{hide}}_{\mathcal{O}}(\mathcal{A})$ and one-way game $\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathcal{O}})$ for $\mathcal{O} \in \{\mathsf{P}_0, \mathsf{P}_1\}$.

At this point, with respect to two fixed permutations P_0 and P_1, in Fig. 2 we define the corresponding hiding game $\mathbf{G}^{\mathsf{hide}}_{\mathcal{O}}(\mathcal{A})$ and one-way game $\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathcal{O}})$ for $\mathcal{O} \in \{\mathsf{P}_0, \mathsf{P}_1\}$. In these games, the distinguisher $\mathcal{A}$ is a quantum oracle algorithm with query time q, and for simplicity we omit its classical input. Following the notation aforementioned, now we recall two previously proven O2H theorems.

Theorem 2 (One-Way to Hiding [3, Theorem 3]). *Let* $\mathsf{G}, \mathsf{H} : \{0,1\}^m \to \{0,1\}^n$ *be two random functions. Let* $\mathcal{A}$ *be a quantum oracle algorithm that takes a random string* z *as input and makes* q *queries. The tuple* $(\mathsf{G}, \mathsf{H}, z)$ *may have an arbitrary joint distribution* $\mathfrak{D}$*. For* $e \in \{1, 1/2\}$*, define*

$$\mathsf{Adv}^{\mathsf{hide}}_e(\mathcal{A}) := \left| \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{G}[\oplus]}(\mathcal{A}) \right] \right)^e - \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{H}[\oplus]}(\mathcal{A}) \right] \right)^e \right| \text{ and}$$

$$\mathsf{Adv}^{\mathsf{ow}}(\mathcal{A}) := \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{G}[\oplus]}) \right].$$

Here the above probabilities are taken over $(\mathsf{G}, \mathsf{H}, z) \leftarrow \mathfrak{D}$*. Then we have*

$$\mathsf{Adv}^{\mathsf{hide}}_e(\mathcal{A}) \leq 2q\sqrt{\mathsf{Adv}^{\mathsf{ow}}(\mathcal{A})}.$$

The same result holds with $\mathcal{A}^{\mathsf{H}[\oplus]}$ *instead of* $\mathcal{A}^{\mathsf{G}[\oplus]}$ *in the definition of* $\mathsf{Adv}^{\mathsf{ow}}(\mathcal{A})$*.*

Theorem 3 (Fixed-Permutation O2H [14, Theorem 1]). *Let* $\mathsf{P}_0, \mathsf{P}_1$ *be two fixed permutations on* $\{0,1\}^n$*. Let* $\mathcal{A}$ *be a quantum oracle algorithm that takes a random string* z *as input and makes* q *queries.* z *may be sampled from an arbitrary distribution* $\mathfrak{D}_z$*. For* $e \in \{1, 1/2\}$*, define*

$$\mathsf{Adv}^{\mathsf{hide}}_{\mathsf{P}_0,\mathsf{P}_1,e}(\mathcal{A}) := \left| \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{P}_0}(\mathcal{A})\right]\right)^e - \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{P}_1}(\mathcal{A})\right]\right)^e \right| \text{ and}$$
$$\mathsf{Adv}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}) := \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_0})\right].$$

Here the above probabilities are taken over $z \leftarrow \mathfrak{D}_z$*. Then we have*

$$\mathsf{Adv}^{\mathsf{hide}}_{\mathsf{P}_0,\mathsf{P}_1,e}(\mathcal{A}) \leq 2q\sqrt{\mathsf{Adv}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A})}.$$

The same result holds with $\mathcal{A}^{\mathsf{P}_1}$ *instead of* $\mathcal{A}^{\mathsf{P}_0}$ *in the definition of* $\mathsf{Adv}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A})$*.*

3 Double-Oracle-Fixed-Permutation O2H

In this section, we first propose a new Double-Oracle O2H theorem. Next, we present its equivalent version in the expression of random permutation, and we call it Double-Oracle-Fixed-Permutation O2H theorem. In the following, we introduce a proven lemma that will be used in later proofs.

Lemma 4 (Distance measure v.s. probability [3, Lemma 4]). *Let* ρ_1*,* ρ_2 *be two density operators. Let* $\mathbb{M} = \{\mathrm{M}_0, \mathrm{M}_1\}$ *be a binary projective measurement. Let* P_i *be the probability that* $\mathbb{M}$ *returns 1 when measuring* ρ_i $(i \in \{1,2\})$*. Then we have* $|(\mathrm{P}_1)^e - (\mathrm{P}_2)^e| \leq B(\rho_1, \rho_2)$ *for* $e \in \{1, 1/2\}$*, where* $B(\rho_1, \rho_2)$ *is the Bures distance between* ρ_1 *and* ρ_2*.*

Now we present our Double-Oracle O2H theorem. This theorem shows that we can bound the advantage of $\mathcal{A}$ distinguishing G and H by the success probabilities of two one-way adversaries, each of which queries G and H respectively.

Theorem 4 (Double-Oracle O2H). *Let* $\mathsf{G}, \mathsf{H} : \{0,1\}^m \to \{0,1\}^n$ *be two random functions. Let* $\mathcal{A}$ *be a quantum oracle algorithm that takes a random string* z *as input and makes* q *queries. The tuple* $(\mathsf{G}, \mathsf{H}, z)$ *may have arbitrary joint distribution* $\mathfrak{D}$*. For* $e \in \{1, 1/2\}$*, define*

$$\mathsf{Adv}^{\mathsf{hide}}_{e}(\mathcal{A}) := \left| \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{G}[\oplus]}(\mathcal{A})\right]\right)^e - \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{H}[\oplus]}(\mathcal{A})\right]\right)^e \right|.$$

Then we have

$$\mathsf{Adv}^{\mathsf{hide}}_{e}(\mathcal{A}) \leq \sqrt{2q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{G}[\oplus]})\right]} + \sqrt{2q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{H}[\oplus]})\right]}.$$

Here the above probabilities are taken over $(\mathsf{G}, \mathsf{H}, z) \leftarrow \mathfrak{D}$*.*

Proof. In this proof, we first consider a fixed $(\mathsf{G}, \mathsf{H}, z)$ tuple. In addition, based on the Fact 1 in Supplementary Material A,

Denote $|\psi_z\rangle$ as the initial state of $\mathcal{A}$, and let $\mathrm{U}_1, \ldots, \mathrm{U}_q$ be the unitary operations performed by $\mathcal{A}$ between its oracle queries. Then, the joint state of $\mathcal{A}^{\mathsf{G}[\oplus]}(z)$ (resp. $\mathcal{A}^{\mathsf{H}[\oplus]}(z)$) just before performing the final binary projective measurement $\mathbb{M}$ can be written as

$$|\Psi_{\mathsf{G}}\rangle := \prod_{i=1}^{q} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}}] |\psi_z\rangle, \quad \left(resp.\ |\Psi_{\mathsf{H}}\rangle := \prod_{i=1}^{q} [\mathrm{U}_i \mathrm{O}_{\mathsf{H}}] |\psi_z\rangle \right). \tag{3}$$

Here $\mathrm{O}_{\mathsf{G}} : |x, y\rangle \mapsto |x, y \oplus \mathsf{G}(x)\rangle$ and $\mathrm{O}_{\mathsf{H}} : |x, y\rangle \mapsto |x, y \oplus \mathsf{H}(x)\rangle$ are the unitary operations used to simulate (quantum-accessible) permutations $\mathsf{G}[\oplus]$ and $\mathsf{H}[\oplus]$, respectively. These two operations act on the query registers IO, where I is the input register and O is the output register.

Define set $S := \{x : \mathsf{G}(x) \neq \mathsf{H}(x)\}$. Define the following two projectors operating on the input register I:

$$\mathrm{J}_S := \sum\nolimits_{x \in S} |x\rangle\langle x|, \mathrm{J}_S^{\perp} := \sum\nolimits_{x \notin S} |x\rangle\langle x|. \tag{4}$$

Note that $\mathrm{O}_{\mathsf{G}} \mathrm{J}_S^{\perp} = \mathrm{O}_{\mathsf{H}} \mathrm{J}_S^{\perp}$ holds. Given this equality, we define the state

$$|\Psi_S^{\perp}\rangle := \prod_{i=1}^{q} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}} \mathrm{J}_S^{\perp}] |\psi_z\rangle = \prod_{i=1}^{q} [\mathrm{U}_i \mathrm{O}_{\mathsf{H}} \mathrm{J}_S^{\perp}] |\psi_z\rangle. \tag{5}$$

Then, applying the inequality $\||\phi_1\rangle + |\phi_2\rangle\|^2 \leq 2 \cdot \||\phi_1\rangle\|^2 + 2 \cdot \||\phi_2\rangle\|^2$ which holds for any states for any states $|\phi_1\rangle$ and $|\phi_2\rangle$, we have

$$\||\Psi_{\mathsf{G}}\rangle - |\Psi_{\mathsf{H}}\rangle\|^2 \leq 2 \cdot \left\||\Psi_{\mathsf{G}}\rangle - |\Psi_S^{\perp}\rangle\right\|^2 + 2 \cdot \left\||\Psi_{\mathsf{H}}\rangle - |\Psi_S^{\perp}\rangle\right\|^2. \tag{6}$$

In the following, we first compute the squared norm $\||\Psi_{\mathsf{G}}\rangle - |\Psi_S^{\perp}\rangle\|^2$ by substituting $|\Psi_S^{\perp}\rangle$ into $\prod_{i=1}^{q} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}} \mathrm{J}_S^{\perp}] |\psi_z\rangle$, where $\mathrm{J}_S^{\perp}$ is defined in Eq. (4):

$$\begin{aligned}
&\left\||\Psi_{\mathsf{G}}\rangle - |\Psi_S^{\perp}\rangle\right\|^2 \\
&= \left\| \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}}] |\psi_z\rangle - \mathrm{J}_S^{\perp} \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}} \mathrm{J}_S^{\perp}] |\psi_z\rangle \right\|^2 \\
&= \left\| \mathrm{J}_S \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}}] |\psi_z\rangle + \mathrm{J}_S^{\perp} \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}}] |\psi_z\rangle - \mathrm{J}_S^{\perp} \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}} \mathrm{J}_S^{\perp}] |\psi_z\rangle \right\|^2 \\
&\overset{(a)}{\leq} \left\| \mathrm{J}_S \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}}] |\psi_z\rangle \right\|^2 + \left\| \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}}] |\psi_z\rangle - \prod_{i=1}^{q-1} [\mathrm{U}_i \mathrm{O}_{\mathsf{G}} \mathrm{J}_S^{\perp}] |\psi_z\rangle \right\|^2.
\end{aligned}$$

Here (a) follows from the equality that $\||\phi_1\rangle + |\phi_2\rangle\|^2 = \||\phi_1\rangle\|^2 + \||\phi_2\rangle\|^2$ holds for any orthogonal states $|\phi_1\rangle$ and $|\phi_2\rangle$, together with the property $\|\mathrm{J}_S^{\perp} |\phi_3\rangle\| \leq \||\phi_3\rangle\|$ for any state $|\phi_3\rangle$. Notably, by comparing the second squared norm above

in (a) with $\||\Psi_{\mathsf{G}}\rangle - |\Psi_S^{\perp}\rangle\|^2$ (where $|\Psi_{\mathsf{G}}\rangle$ and $|\Psi_S^{\perp}\rangle$ are defined in Eq. (3) and Eq. (5)), one can observe that the former reduces the number of queries to O_{G} from q to $q-1$. By defining $\prod_{i=1}^{0}[\mathrm{U}_i\mathrm{O}_{\mathsf{G}}]|\psi_z\rangle = |\psi_z\rangle$, we can derive the final formulation through a recursive argument:

$$\left\||\Psi_{\mathsf{G}}\rangle - |\Psi_S^{\perp}\rangle\right\|^2 \leq \sum_{j=0}^{q-1} \left\| \mathrm{J}_S \prod_{i=1}^{j} [\mathrm{U}_i\mathrm{O}_{\mathsf{G}}]|\psi_z\rangle \right\|^2 .$$

Meanwhile, it is straightforward to verify that

$$\Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{G}[\oplus]}) : \mathsf{G},\mathsf{H},z\right] = \sum_{j=0}^{q-1} \frac{1}{q} \cdot \left\| \mathrm{J}_S \prod_{i=1}^{j} [\mathrm{U}_i\mathrm{O}_{\mathsf{G}}]|\psi_z\rangle \right\|^2 ,$$

where we include G,H,z in the probability expression on the left to emphasize that the tuple $(\mathsf{G},\mathsf{H},z)$ is fixed. Therefore, by combining the above two formulations, we can obtain

$$\left\||\Psi_{\mathsf{G}}\rangle - |\Psi_S^{\perp}\rangle\right\|^2 \leq q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{G}[\oplus]}) : \mathsf{G},\mathsf{H},z\right]. \tag{7}$$

As for the $\||\Psi_{\mathsf{H}}\rangle - |\Psi_S^{\perp}\rangle\|^2$, by substituting $|\Psi_S^{\perp}\rangle$ into $\prod_{i=1}^{q}[\mathrm{U}_i\mathrm{O}_{\mathsf{H}}\mathrm{J}_S^{\perp}]|\psi_z\rangle$ and by using a computation similar to that for $\||\Psi_{\mathsf{G}}\rangle - |\Psi_S^{\perp}\rangle\|^2$, we can obtain

$$\left\||\Psi_{\mathsf{H}}\rangle - |\Psi_S^{\perp}\rangle\right\|^2 \leq q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{H}[\oplus]}) : \mathsf{G},\mathsf{H},z\right]. \tag{8}$$

Next, we consider random $(\mathsf{G},\mathsf{H},z)$ tuple which is sampled according to the distribution $\mathfrak{D}$. At this point, denote the mixed state just before the final binary projective measurement $\mathbb{M}$ of $\mathcal{A}^{\mathsf{G}[\oplus]}(z)$ (resp. $\mathcal{A}^{\mathsf{H}[\oplus]}(z)$) as ρ_{G} (resp. ρ_{H}). Then we have $\rho_{\mathsf{G}} = \mathbb{E}|\Psi_{\mathsf{G}}\rangle\langle\Psi_{\mathsf{G}}|$ and $\rho_{\mathsf{H}} = \mathbb{E}|\Psi_{\mathsf{H}}\rangle\langle\Psi_{\mathsf{H}}|$, where the expectations are taken over $(\mathsf{G},\mathsf{H},z) \leftarrow \mathfrak{D}$. Hence, for the fidelity $F(\rho_{\mathsf{G}},\rho_{\mathsf{H}})$ [3] between ρ_{G} and ρ_{H}, we can compute

$$\begin{aligned} F(\rho_{\mathsf{G}},\rho_{\mathsf{H}}) &= F\left(\mathbb{E}|\Psi_{\mathsf{G}}\rangle\langle\Psi_{\mathsf{G}}|, \mathbb{E}|\Psi_{\mathsf{H}}\rangle\langle\Psi_{\mathsf{H}}|\right) \\ &\overset{(a)}{\geq} \mathbb{E}\left[F\left(|\Psi_{\mathsf{G}}\rangle\langle\Psi_{\mathsf{G}}|, |\Psi_{\mathsf{H}}\rangle\langle\Psi_{\mathsf{H}}|\right)\right] \\ &\overset{(b)}{\geq} 1 - \frac{1}{2}\cdot\mathbb{E}\left[\||\Psi_{\mathsf{G}}\rangle - |\Psi_{\mathsf{H}}\rangle\|^2\right] \\ &\overset{(c)}{\geq} 1 - q\cdot\left(\Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{G}[\oplus]})\right] + \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{H}[\oplus]})\right]\right). \end{aligned} \tag{9}$$

Here (a) follows from the joint concavity of the fidelity [19, (9.95)], (b) is derived from the property that $F(|\Psi\rangle\langle\Psi|, |\Phi\rangle\langle\Phi|) \geq 1 - \||\Psi\rangle - |\Phi\rangle\|^2/2$ for any pure states $|\Psi\rangle$ and $|\Phi\rangle$, and (c) is obtained from Eq. (6) to Eq. (8).

Let $\mathrm{P}_{\mathbb{M}}(\rho_{\mathsf{G}})$ (resp. $\mathrm{P}_{\mathbb{M}}(\rho_{\mathsf{H}})$) denote the probability that $\mathbb{M}$ returns 1 when measuring ρ_{G} (resp. ρ_{H}). Thus, for $e \in \{1, 1/2\}$, we can compute

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{hide}}_{e}(\mathcal{A}) &= \left| \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{G}[\oplus]}(\mathcal{A})\right] \right)^{e} - \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{H}[\oplus]}(\mathcal{A})\right] \right)^{e} \right| \\
&= \left| (\mathrm{P}_{\mathbb{M}}(\rho_{\mathsf{G}}))^{e} - (\mathrm{P}_{\mathbb{M}}(\rho_{\mathsf{H}}))^{e} \right| \\
&\overset{(a)}{\leq} B(\rho_{\mathsf{G}}, \rho_{\mathsf{H}}) \\
&\overset{(b)}{=} \sqrt{2(1 - F(\rho_{\mathsf{G}}, \rho_{\mathsf{H}}))} \\
&\overset{(c)}{\leq} \sqrt{2q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{G}[\oplus]})\right]} + \sqrt{2q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{H}[\oplus]})\right]}.
\end{aligned}$$

Here (a) uses Lemma 4, (b) follows from the definition of the Bures distance, and (c) uses Eq. (9). This completes the proof of Theorem 4. □

We now introduce Theorem 5, the main result of this section, which can be viewed as a double-oracle variant of Theorem 3. To prove Theorem 5, we adapt the proof routine from [14, Theorem 6] and show that it is equivalent to Theorem 4 up to a constant factor.

Theorem 5 (Double-Oracle-Fixed-Permutation O2H, DOFP-O2H). *Let $\mathsf{P}_0, \mathsf{P}_1$ be two fixed permutations. Let $\mathcal{A}$ be a quantum oracle algorithm that takes a random string z as input and makes q queries. z may be sampled from an arbitrary distribution $\mathfrak{D}_z$. For $e \in \{1, 1/2\}$, define*

$$\mathsf{Adv}^{\mathsf{hide}}_{\mathsf{P}_0,\mathsf{P}_1,e}(\mathcal{A}) := \left| \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{P}_0}(\mathcal{A})\right] \right)^{e} - \left(\Pr\left[\mathbf{G}^{\mathsf{hide}}_{\mathsf{P}_1}(\mathcal{A})\right] \right)^{e} \right|.$$

Then we have

$$\mathsf{Adv}^{\mathsf{hide}}_{\mathsf{P}_0,\mathsf{P}_1,e}(\mathcal{A}) \leq \sqrt{2q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_0})\right]} + \sqrt{2q \cdot \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_1})\right]}.$$

Here the above probabilities are taken over $z \leftarrow \mathfrak{D}_z$.

Proof. We first prove that Theorem 5 implies Theorem 4. Given that in Theorem 4, G and H are functions from $\{0,1\}^m$ to $\{0,1\}^n$, we define two fixed permutations P_0 and P_1 as follows:

$$\begin{aligned}
\mathsf{P}_0 &: (x, y, f, f') \mapsto (x, y \oplus f(x), f, f'), \\
\mathsf{P}_1 &: (x, y, f, f') \mapsto (x, y \oplus f'(x), f, f').
\end{aligned}$$

Here $f, f' \in \mathrm{Fcs}[m, n]$. At this point, we construct a distinguisher $\mathcal{B}$ in Theorem 5, which has input $(\mathsf{G}, \mathsf{H}, z)$ and invokes the distinguisher $\mathcal{A}$ in Theorem 4. In more detail, $(\mathsf{G}, \mathsf{H}, z)$ is sampled according to the distribution $\mathfrak{D}$, and $\mathcal{B}$ directly queries permutation oracles P_0 (resp. P_1) with query state $|x, y, \mathsf{G}, \mathsf{H}\rangle$ to simulate function oracles $\mathsf{G}[\oplus]$ (resp. $\mathsf{H}[\oplus]$) for $\mathcal{A}$ with query state $|x, y\rangle$. It follows that

$$\mathsf{Adv}^{\mathsf{hide}}_{e}(\mathcal{A}) = \mathsf{Adv}^{\mathsf{hide}}_{\mathsf{P}_0,\mathsf{P}_1,e}(\mathcal{B}),\ \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{G}[\oplus]})\right] = \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{B}^{\mathsf{P}_0})\right],$$

$$\textit{and } \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{G}[\oplus],\mathsf{H}[\oplus]}(\mathcal{A}^{\mathsf{H}[\oplus]})\right] = \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{B}^{\mathsf{P}_1})\right].$$

Hence, Theorem 5 immediately yields the result stated in Theorem 4.

Now we prove that Theorem 4 implies Theorem 5. First, we denote the input and output space of permutations P_0 and P_1 as $\{0,1\}^n$. Then, by employing the same proof strategy as in [14], we can define the following two functions based on P_0 and P_1:

$$\mathsf{P}^0_{\pm} : \begin{cases} (1,x) \mapsto \mathsf{P}_0(x), \\ (0,x) \mapsto \mathsf{P}_0^{-1}(x). \end{cases} \quad \mathsf{P}^1_{\pm} : \begin{cases} (1,x) \mapsto \mathsf{P}_1(x), \\ (0,x) \mapsto \mathsf{P}_1^{-1}(x). \end{cases} \tag{10}$$

Recall that $\mathsf{P}^0_{\pm}[\oplus]$ and $\mathsf{P}^1_{\pm}[\oplus]$ denote the function-like permutations corresponding to $\mathsf{P}^0_{\pm}$ and $\mathsf{P}^1_{\pm}$, respectively. At this point, for a distinguisher $\mathcal{A}$ with input $z(\leftarrow \mathfrak{D}_z)$ in Theorem 5, we accordingly construct a distinguisher $\mathcal{B}$ in Theorem 4 as follows, which invokes $\mathcal{A}$ to distinguish functions $\mathsf{P}^0_{\pm}$ and $\mathsf{P}^1_{\pm}$.

1. $\mathcal{B}$ prepares two registers B and Y, and sets the initial state of B to $|1\rangle$ and the initial state of Y to $|0^n\rangle$.
2. Whenever $\mathcal{A}$ queries its oracle P_b $(b \in \{0,1\})$ with the query state $|x\rangle_{\mathsf{X}}$, $\mathcal{B}$ executes the following steps to reply:
   (a) $\mathcal{B}$ queries its oracle $\mathsf{P}^b_{\pm}[\oplus]$ with the query state $|1,x,0^n\rangle_{\mathsf{BXY}}$, where BX is the input register and Y is the output register.
   (b) $\mathcal{B}$ flips the state of B to $|0\rangle$ and queries $\mathsf{P}^b_{\pm}[\oplus]$ again with the input/output register BY/X.
   (c) $\mathcal{B}$ flips the state of B again to restore it to the state $|1\rangle$.
   (d) $\mathcal{B}$ swaps the state of the registers X and Y.

It can be observed that $\mathcal{B}$ queries its oracle $2q$ times, and that steps $2(a)$ to $2(d)$ yield the state $|1,\mathsf{P}_b(x),0\rangle_{\mathsf{BXY}}$, which shows that $\mathcal{B}$ perfectly simulates the permutation oracle P_b $(b \in \{0,1\})$ for $\mathcal{A}$. Hence, we have

$$\mathsf{Adv}^{\mathsf{hide}}_{e}(\mathcal{B}) = \mathsf{Adv}^{\mathsf{hide}}_{\mathsf{P}_0,\mathsf{P}_1,e}(\mathcal{A}). \tag{11}$$

Here the left distinguishing advantage is based on a specific joint distribution of $(\mathsf{P}^0_{\pm}[\oplus], \mathsf{P}^1_{\pm}[\oplus], z)$, where $\mathsf{P}^b_{\pm}[\oplus]$ and $\mathsf{P}^b_{\pm}[\oplus]$ are fixed since P_0 and P_1 are fixed and z is sampled according to $\mathfrak{D}_z$.

It should be noted that when $\mathcal{B}$ simulates the oracle P_b $(b \in \{0,1\})$ for $\mathcal{A}$, although it makes two queries to the $\mathsf{P}^b_{\pm}[\oplus]$ oracle, it does not change the state of the query register X. Thus, in essence, the value of the register X from the result of randomly choosing one of $\mathcal{B}$'s $\mathsf{P}^b_{\pm}[\oplus]$ queries for measurement has the same distribution as that from the result of randomly choosing one of $\mathcal{A}$'s P_b queries for measurement. This actually implies that

$$\begin{aligned} \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_0})\right] &= \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}^0_{\pm}[\oplus],\mathsf{P}^1_{\pm}[\oplus]}(\mathcal{B}^{\mathsf{P}^0_{\pm}[\oplus]})\right], \\ \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_1})\right] &= \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}^0_{\pm}[\oplus],\mathsf{P}^1_{\pm}[\oplus]}(\mathcal{B}^{\mathsf{P}^1_{\pm}[\oplus]})\right]. \end{aligned} \tag{12}$$

A detailed proof of Eq. (12) can be found in Supplementary Material B. Combining Eq. (11) and Eq. (12) with the fact that $\mathcal{B}$ queries its oracle $2q$ times, it follows that Theorem 4 implies Theorem 5 with an additional factor of at most $\sqrt{2}$ in the security bound. In fact, a direct emulation of the proof for Theorem 4 gives the bound claimed in Theorem 5, and we omit the details for brevity. This completes the proof of Theorem 5. □

4 Tighter GHHM Adaptive Reprogramming Theorem

Here we prove a tighter Grilo, Hövelmanns, Hülsing and Majenz (GHHM) adaptive reprogramming theorem [13]. In Section 4.1 we first review the GHHM reprogramming framework and GHHM reprogramming theorem. Then, in Section 4.2 we prove a tighter GHHM reprogramming theorem by using our DOFP-O2H.

4.1 GHHM Reprogramming Framework and Theorem

The GHHM framework defines a specific adaptive reprogramming setting in the QROM. The adversary $\mathcal{A}$ is given quantum oracle access to H_b, which is either the original random oracle when $b = 0$ or its reprogrammed version when $b = 1$. The goal of $\mathcal{A}$ is to guess b. An important feature of this framework is that $\mathcal{A}$ receives both the (potential) reprogramming point x and the side information x' generated when x is sampled. Consistent with [14], we restate the GHHM framework (Fig. 3) in terms of the Pan-Zeng framework [20], and we retain the notation in [14].

Game $\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},b}(\mathcal{A})$ $(b \in \{0,1\})$

1. $(\mathsf{H}_0, \mathsf{H}_1) \leftarrow \mathsf{Init}$
2. $y \leftarrow \mathcal{A}_0^{\mathsf{H}_b[\oplus]}$
3. **for** $i = 1, \ldots, r$ **do**
4. $\quad (z, x, y') \leftarrow \mathsf{Orac}(y)$
5. $\quad \mathsf{H}_1 \leftarrow \mathsf{Repro}(x, y', \mathsf{H}_1)$
6. $\quad y \leftarrow \mathcal{A}_i^{\mathsf{H}_b[\oplus]}(z)$
7. **return** $[\![y = 1]\!]$

Init

1. $\mathsf{H} \leftarrow_{\$} \mathrm{Fcs}[m, n]$
2. $\mathsf{H}_0 := \mathsf{H}, \mathsf{H}_1 := \mathsf{H}$
3. **return** $(\mathsf{H}_0, \mathsf{H}_1)$

$\mathsf{Repro}(x, y', \mathsf{H}_1)$

1. **return** $\mathsf{H}_1[x \mapsto y']$

$\mathsf{Orac}(y)$

1. Interpret y as a distribution p over $\{0,1\}^m \times \{0,1\}^*$
2. $(x, x') \leftarrow p, y' \leftarrow_{\$} \{0,1\}^n$
3. **return** $((x, x'), x, y')$

Fig. 3. The GHHM framework based on the environment $\mathcal{E} := (\mathsf{Init}, \mathsf{Orac}, \mathsf{Repro})$. Here, the adversary $\mathcal{A}$ is divided into $r + 1$ stages, namely, $\mathcal{A} = (\mathcal{A}_0, \ldots, \mathcal{A}_r)$.

The GHHM environment is denoted as $\mathcal{E} := (\mathsf{Init}, \mathsf{Orac}, \mathsf{Repro})$, where the three algorithms are specified in Fig. 3. Accordingly, the GHHM theorem provided in [13] is restated as follows:

Theorem 6 (GHHM Adaptive Reprogramming, [13, Theorem 1]). *Let $\mathcal{E}$ be a GHHM environment and $\mathcal{A} = (\mathcal{A}_0, \ldots, \mathcal{A}_r)$ be an adversary for which $\mathcal{A}_i$ makes q_i oracle queries. Define*

$$\mathsf{Adv}^{\mathsf{ghhm}}_{\mathcal{E}}(\mathcal{A}) := \left| \Pr\left[\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},0}(\mathcal{A})\right] - \Pr\left[\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},1}(\mathcal{A})\right] \right|.$$

Let p^i denote the i-th distribution interpreted by Orac. Denote $p^i_{\{0,1\}^m}$ as the marginal distribution of p^i over $\{0,1\}^m$. Define $p^i_{\max} := \mathbb{E}[\max_x p^i_{\{0,1\}^m}(x)]$, where the expectation is taken over $\mathcal{A}$'s behaviour until the i-th query to Orac. Then we have

$$\mathsf{Adv}^{\mathsf{ghhm}}_{\mathcal{E}}(\mathcal{A}) \leq \sum_{i=1}^{r} \left(\sqrt{\hat{q}_i \cdot p^i_{\max}} + \frac{1}{2} \cdot \hat{q}_i \cdot p^i_{\max} \right).$$

Here $\hat{q}_i := q_0 + \ldots + q_{i-1}$.

In the simple but common case where $p^i_{\max}$ is bounded by a constant $p_{\max}$ for $\forall i \in [r]$, we obtain the following corollary. Notably, this corollary is precisely the one used in [13] to prove the $\mathsf{EUF\text{-}CMA}$ security of the Fiat–Shamir transform in the QROM.

Corollary 3. *Suppose that $\mathcal{A}$ makes a total of q (quantum) queries to its oracle, hence $\hat{q}_i \leq q$ for all $i \in [r]$. Further, assume that $p^i_{\max} \leq p_{\max}$ for all $i \in [r]$. Then we have $\mathsf{Adv}^{\mathsf{ghhm}}_{\mathcal{E}}(\mathcal{A}) \leq 1.5r\sqrt{q \cdot p_{\max}}$.*

4.2 Tighter GHHM Reprogramming Theorem

Next, by applying our DOFP-O2H theorem, we establish the following variant of Theorem 6, which achieves a tighter upper bound.

Theorem 7. *Let $\mathcal{E}$ be a GHHM environment and $\mathcal{A} = (\mathcal{A}_0, \ldots, \mathcal{A}_r)$ be an adversary for which $\mathcal{A}_i$ makes q_i oracle queries. For $e \in \{1, 1/2\}$, define*

$$\mathsf{Adv}^{\mathsf{ghhm}}_{\mathcal{E},e}(\mathcal{A}) := \left| \left(\Pr\left[\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},0}(\mathcal{A})\right]\right)^e - \left(\Pr\left[\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},1}(\mathcal{A})\right]\right)^e \right|.$$

Define $\hat{q}_i$ and $p^i_{\max}$ as in Theorem 6, we have

$$\mathsf{Adv}^{\mathsf{ghhm}}_{\mathcal{E},e}(\mathcal{A}) \leq 2\sqrt{2} \cdot \sqrt{\sum_{i=1}^{r} \hat{q}_i \cdot p^i_{\max}}.$$

Proof. Here, for the sake of clarity, we introduce a sequence of hybrid games $\mathbf{G_0}$ through $\mathbf{G_3}$ as shown in Fig. 4 to complete the proof.

Game $\mathbf{G_0}$: This game reformulates game $\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},0}(\mathcal{A})$ defined in Fig. 3. The key idea behind this reformulation is as follows.

– We omit the algorithm Init and directly sample a function H from the set $\mathrm{Fcs}[m,n]$. Then, we introduce registers $\mathsf{H}_{0^m}\dots\mathsf{H}_{1^m}$ and initialize them to the state $|\mathsf{H}(0^m)\rangle_{\mathsf{H}_{0^m}}\dots|\mathsf{H}(1^m)\rangle_{\mathsf{H}_{1^m}}$ to store the truth table of H. Additionally, we introduce registers $\mathsf{Z}_1\dots\mathsf{Z}_r$, initializing them to the all-zero state $|0^n\rangle_{\mathsf{Z}_1}\dots|0^n\rangle_{\mathsf{Z}_r}$, and $\mathsf{H}_{0^m}\dots\mathsf{H}_{1^m}\mathsf{Z}_1\dots\mathsf{Z}_r$ are viewed as the internal registers and hence are hidden from the adversary.
– The quantum-accessible random oracle $\mathsf{H}_0[\oplus]$ is equivalently replaced by the impurified standard oracle StO' defined in Section 2.2.
– The algorithm Repro in game $\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},0}(\mathcal{A})$ is now modeled as a quantum oracle. It takes as input the query state $|i,x,y'\rangle$ in register R, where i is the loop index and (x,y') is generated by Orac. Since the execution of algorithm Repro in game $\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},0}(\mathcal{A})$ does not affect the adversary's behavior, we set the quantum oracle Repro in game $\mathbf{G}_0$ as a trivial oracle that only performs identity operator.

Clearly, Repro is queried r times in game $\mathbf{G_0}$ and we have

$$\Pr\left[\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},0}(\mathcal{A})\right]=\Pr\left[\mathbf{G_0}\right]. \tag{13}$$

Game $\mathbf{G_1}$: This game is identical to game $\mathbf{G_0}$ except that registers $\mathsf{H}_{0^m}\dots\mathsf{H}_{1^m}$ are initialized to the all-zero state $|0^n\rangle_{\mathsf{H}_{0^m}}\dots|0^n\rangle_{\mathsf{H}_{1^m}}$, and the oracle StO' is replaced with $\mathsf{H}^{\mathsf{Y}}\circ\mathsf{FourierPhsO}\circ\mathsf{H}^{\mathsf{Y}}$. Here H^{Y} is the Hadamard operation on register Y and $\mathsf{FourierPhsO}$ is the Fourier phase oracle defined in Section 2.2.

GAMES $\mathbf{G_0}$-$\mathbf{G_3}$

1. $\mathsf{H}\leftarrow_{\$}\mathrm{Fcs}[m,n]$ //$\mathbf{G_0}$-$\mathbf{G_3}$
2. Set $|\mathsf{H}(0^m)\rangle_{\mathsf{H}_{0^m}}\dots|\mathsf{H}(1^m)\rangle_{\mathsf{H}_{1^m}}$ //$\mathbf{G_0}$
   Set $|0^n\rangle_{\mathsf{H}_{0^m}}\dots|0^n\rangle_{\mathsf{H}_{1^m}}$ //$\mathbf{G_1}$-$\mathbf{G_2}$
   Set $|0^n\rangle_{\mathsf{Z}_1}\dots|0^n\rangle_{\mathsf{Z}_r}$ //$\mathbf{G_0}$-$\mathbf{G_2}$
3. $y\leftarrow\mathcal{A}_0^{\mathsf{StO}'}$ //$\mathbf{G_0}$
   $y\leftarrow\mathcal{A}_0^{\mathsf{H}^{\mathsf{Y}}\circ\mathsf{FourierPhsO}\circ\mathsf{H}^{\mathsf{Y}}}$ //$\mathbf{G_1}$-$\mathbf{G_2}$
   $y\leftarrow\mathcal{A}_0^{\mathsf{H}[\oplus]}$ //$\mathbf{G_3}$
4. **for** $i=1,\dots,r$ **do** //$\mathbf{G_0}$-$\mathbf{G_3}$
   $(z,x,y')\leftarrow\mathsf{Orac}(y)$ //$\mathbf{G_0}$-$\mathbf{G_3}$
   query $\mathsf{Repro}(|i,x,y'\rangle_{\mathsf{R}})$ //$\mathbf{G_0}$-$\mathbf{G_1}$
   query $\mathsf{Repro}'(|i,x,y'\rangle_{\mathsf{R}})$ //$\mathbf{G_2}$
   query $\mathsf{Repro}^{\diamond}(x,y',\mathsf{H})$ //$\mathbf{G_3}$
   $y\leftarrow\mathcal{A}_i^{\mathsf{StO}'}(z)$ //$\mathbf{G_0}$
   $y\leftarrow\mathcal{A}_i^{\mathsf{H}^{\mathsf{Y}}\circ\mathsf{FourierPhsO}\circ\mathsf{H}^{\mathsf{Y}}}(z)$ //$\mathbf{G_1}$-$\mathbf{G_2}$
   $y\leftarrow\mathcal{A}_i^{\mathsf{H}[\oplus]}(z)$ //$\mathbf{G_3}$
5. **return** $[\![y=1]\!]$ //$\mathbf{G_0}$-$\mathbf{G_3}$

$\mathsf{StO}'(|x,y\rangle_{\mathsf{XY}}|\mathsf{H}(0^m)\rangle_{\mathsf{H}_{0^m}}\dots|\mathsf{H}(1^m)\rangle_{\mathsf{H}_{1^m}})$ //$\mathbf{G_0}$

1. **return** $|x,y\oplus\mathsf{H}(x)\rangle_{\mathsf{XY}}|\mathsf{H}(0^m)\rangle_{\mathsf{H}_{0^m}}\dots|\mathsf{H}(1^m)\rangle_{\mathsf{H}_{1^m}}$

$\mathsf{FourierPhsO}(|x,y\rangle_{\mathsf{XY}}|y_{0^m}\rangle_{\mathsf{H}_{0^m}}\dots|y_{1^m}\rangle_{\mathsf{H}_{1^m}})$ //$\mathbf{G_1}$-$\mathbf{G_2}$

1. **return** $|x,y\rangle_{\mathsf{XY}}|y_{0^m}\rangle_{\mathsf{H}_{0^m}}\dots|y_x\oplus y\rangle_{\mathsf{H}_x}\dots|y_{1^m}\rangle_{\mathsf{H}_{1^m}}$

$\mathsf{Orac}(y)$ //$\mathbf{G_0}$-$\mathbf{G_3}$

1. Interpret y as a distribution p over $\{0,1\}^m\times\{0,1\}^*$
2. $(x,x')\leftarrow p, y'\leftarrow_{\$}\{0,1\}^n$
3. **return** $((x,x'),x,y')$

$\mathsf{Repro}(|i,x,y'\rangle_{\mathsf{R}})$ //$\mathbf{G_0}$-$\mathbf{G_1}$

1. perform identity operator on $\mathsf{RH}_{0^m}\dots\mathsf{H}_{1^m}\mathsf{Z}_1\dots\mathsf{Z}_r$

$\mathsf{Repro}'(|i,x,y'\rangle_{\mathsf{R}})$ //$\mathbf{G_2}$

1. perform a swap on $\mathsf{RH}_{0^m}\dots\mathsf{H}_{1^m}\mathsf{Z}_1\dots\mathsf{Z}_r$ which swaps the state of H_x and Z_i

$\mathsf{Repro}^{\diamond}(x,y',\mathsf{H})$ //$\mathbf{G_3}$

1. $\mathsf{H}\leftarrow\mathsf{H}[x\mapsto y']$

Fig. 4. Games $\mathbf{G_0}$ to $\mathbf{G_3}$ in the proof of Theorem 7.

According to Corollary 1, StO' is equivalent to $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$, hence we have

$$\Pr\left[\mathbf{G_0}\right] = \Pr\left[\mathbf{G_1}\right]. \tag{14}$$

Game $\mathbf{G_2}$: This game is identical to $\mathbf{G_1}$, except that the quantum oracle Repro is replaced with Repro'. Here, when queried with the state $|i, x, y'\rangle_{\mathsf{R}}$, Repro' swaps the state of the registers H_x and Z_i.

Next, we apply Theorem 5 to evaluate the difference between the success probabilities of game $\mathbf{G_1}$ and $\mathbf{G_2}$. We begin by defining two fixed permutations on the set $[r] \times \{0,1\}^m \times \{0,1\}^n \times \{0,1\}^{n2^m} \times \{0,1\}^{nr}$ as follows:

$$\begin{aligned}
&\mathsf{P}\left(i, x, y', y_{0^m}, \ldots, y_{1^m}, z_1, \ldots, z_r\right) = \left(i, x, y', y_{0^m}, \ldots, y_{1^m}, z_1, \ldots, z_r\right), \\
&\mathsf{P}'\left(i, x, y', y_{0^m}, \ldots, y_{1^m}, z_1, \ldots, z_r\right) = \begin{pmatrix} i, x, y', y_{0^m}, \ldots, y_{x-1}, z_i, y_{x+1}, \ldots, y_{1^m}, \\ z_1, \ldots, z_{i-1}, y_x, z_{i+1} \ldots, z_r \end{pmatrix}.
\end{aligned}$$

Then, by denoting $w := (i, x, y', y_{0^m}, \ldots, y_{1^m}, z_1, \ldots, z_r)$, the action of the oracles is clear: for the joint state $|w\rangle$ of registers $\mathsf{RH}_{0^m} \ldots \mathsf{H}_{1^m} \mathsf{Z}_1 \ldots \mathsf{Z}_r$, Repro maps $|w\rangle$ into $|\mathsf{P}(w)\rangle$ and Repro' maps $|w\rangle$ into $|\mathsf{P}'(w)\rangle$. By applying Theorem 5, for $e \in \{1, 1/2\}$, we have

$$\begin{aligned}
\left|\left(\Pr\left[\mathbf{G}_1\right]\right)^e - \left(\Pr[\mathbf{G}_2]\right)^e\right| \leq & \sqrt{2r \cdot \Pr\left[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{B}^{\mathsf{Repro}}\right]} \\
& + \sqrt{2r \cdot \Pr\left[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{B}^{\mathsf{Repro}'}\right]}.
\end{aligned} \tag{15}$$

The definition of $\mathcal{B}$ is given in Fig. 5. We first stress that regardless of which oracle (Repro or Repro') $\mathcal{B}$ has access to, its output $w = (j, x, y', y_{0^m}, \ldots, y_{1_m}, z_1, \ldots, z_r)$ must satisfy $z_j = 0^n$. This can be easily deduced, since the first component of w is the loop index j, and by the construction of $\mathcal{B}$ it follows that $\mathcal{B}$ never queries its oracle using j as the first element. At this point, for the output w of $\mathcal{B}$, we have that $\mathsf{P}(w) \neq \mathsf{P}'(w)$ if and only if $y_x \neq 0^n$.

That is to say, if $\mathcal{B}^{\mathsf{Repro}}$ or $\mathcal{B}^{\mathsf{Repro}'}$ outputs a w such that $\mathsf{P}(w) \neq \mathsf{P}'(w)$, we can directly guess x by measuring the registers $\mathsf{H}_{0^m} \ldots \mathsf{H}_{1^m}$ even if x is independently sampled from a distribution with the min-entropy $p^j_{\max}$. The detailed guessing process is as follows.

- Run $\mathcal{B}^{\mathcal{O}}$ ($\mathcal{O} \in \{\mathsf{Repro}, \mathsf{Repro}'\}$) until it aborts, measure $\mathsf{H}_{0^m} \ldots \mathsf{H}_{1^m}$ to get $(y_{0^m}, \ldots, y_{1^m})$, then uniformly sample an element $\hat{x}$ from the set $\{x \in \{0,1\}^m : y_x \neq 0^n\}$, and finally guess that $\hat{x}$ is the value x independently sampled from the distribution $p^j_{\{0,1\}^m}$.

During the execution of $\mathcal{B}^{\mathcal{O}}$ ($\mathcal{O} \in \{\mathsf{Repro}, \mathsf{Repro}'\}$, only queries to $\mathsf{FourierPhsO}$ or Repro' may change the state of registers $\mathsf{H}_{0^m} \ldots \mathsf{H}_{1^m}$. Notably, according to the definition of $\mathsf{FourierPhsO}$ given in Fig. 4, one query to $\mathsf{FourierPhsO}$ can change the state of at most one register in $\mathsf{H}_{0^m} \ldots \mathsf{H}_{1^m}$ to a non-zero state. Moreover, for Repro', $\mathcal{B}$ in Fig. 5 queries Repro' sequentially on the states $|1, x, y'\rangle_{\mathsf{R}}, \ldots, |j-1, x, y'\rangle_{\mathsf{R}}$. From the definition of Repro' in Fig. 4, a single query to Repro' changes

the state of at most one register in $\mathsf{H}_{0^m} \dots \mathsf{H}_{1^m}$ to the zero state. Consequently, regardless of which oracle $\mathcal{B}$ queries, the set $\{x \in \{0,1\}^m : y_x \neq 0^n\}$ considered in the above guess contains at most $\hat{q}_j = q_0 + \dots + q_{j-1}$ elements. Hence, if $\mathcal{B}$ outputs some w such that $\mathsf{P}(w) \neq \mathsf{P}'(w)$, the success probability of the above guess is bounded by $\hat{q}_j \cdot p^j_{\max}$, where j is sampled uniformly from $[r]$. Therefore,

$$\begin{aligned} \Pr\left[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{B}^{\mathsf{Repro}}\right] &\leq \sum_{j=1}^{r} \frac{1}{r} \cdot \hat{q}_j \cdot p^j_{\max}, \\ \Pr\left[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{B}^{\mathsf{Repro}'}\right] &\leq \sum_{j=1}^{r} \frac{1}{r} \cdot \hat{q}_j \cdot p^j_{\max}. \end{aligned} \tag{16}$$

By combining Eq. (15) with Eq. (16), for $e \in \{1, 1/2\}$ we can obtain

$$|(\Pr[\mathbf{G}_1])^e - (\Pr[\mathbf{G}_2])^e| \leq 2\sqrt{2} \cdot \sqrt{\sum_{j=1}^{r} \hat{q}_j \cdot p^j_{\max}}. \tag{17}$$

$\mathcal{B}^{\mathcal{O}}$ $(\mathcal{O} \in \{\mathsf{Repro}, \mathsf{Repro}'\})$

1. Set $|0^n\rangle_{\mathsf{H}_{0^m}} \dots |0^n\rangle_{\mathsf{H}_{1^m}}$, Set $|0^n\rangle_{\mathsf{Z}_1} \dots |0^n\rangle_{\mathsf{Z}_r}$
2. $j \leftarrow_{\$} [r]$
3. **if** $j = 1$
   - $y \leftarrow \mathcal{A}_0^{\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}}$
   - abort, execute $(z, x, y') \leftarrow \mathsf{Orac}(y)$, set the state of R as $|j, x, y'\rangle$
   - measure $\mathsf{RH}_{0^m} \dots \mathsf{H}_{1^m} \mathsf{Z}_1 \dots \mathsf{Z}_r$ to get $w = (j, x, y', y_{0^m}, \dots, y_{1^m}, z_1, \dots, z_r)$

   else
   - $y \leftarrow \mathcal{A}_0^{\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}}$
   - **for** $i = 1, \dots, j-1$ **do**
     - $(z, x, y') \leftarrow \mathsf{Orac}(y)$
     - query $\mathcal{O}(|i, x, y'\rangle_{\mathsf{R}})$
     - $y \leftarrow \mathcal{A}_i^{\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}}(z)$
   - abort, execute $(z, x, y') \leftarrow \mathsf{Orac}(y)$, set the state of R as $|j, x, y'\rangle$
   - measure $\mathsf{RH}_{0^m} \dots \mathsf{H}_{1^m} \mathsf{Z}_1 \dots \mathsf{Z}_r$ to get $w = (j, x, y', y_{0^m}, \dots, y_{1^m}, z_1, \dots, z_r)$
4. **return** w

Fig. 5. Quantum oracle algorithm $\mathcal{B}^{\mathcal{O}}$ $(\mathcal{O} \in \{\mathsf{Repro}, \mathsf{Repro}'\})$ in the proof of Theorem 7. The $\mathsf{FourierPhsO}$, Orac, Repro, and Repro' used by $\mathcal{B}$ are defined in Fig. 4.

Game $\mathbf{G_3}$: This game is the same as $\mathbf{G_2}$, except that the oracle $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$ is replaced with $\mathsf{H}[\oplus]$, and Repro' is replaced with $\mathsf{Repro}^{\diamond}$. Here, $\mathsf{H}[\oplus]$ maps $|x, y\rangle_{\mathsf{XY}}$ into $|x, y \oplus \mathsf{H}(x)\rangle_{\mathsf{XY}}$, and $\mathsf{Repro}^{\diamond}$ reprograms H at point x.

Note that in game $\mathbf{G}_2$, Repro' performs the reprogramming of $\mathsf{H}^{\mathsf{Y}} \circ \mathsf{FourierPhsO} \circ \mathsf{H}^{\mathsf{Y}}$ at point x. Consequently, according to Corollary 2, we can first convert the simulation of the quantum-accessible random oracle back to StO'. Meanwhile, the reprogramming oracle is correspondingly converted to $\mathsf{Reprogram}_0$ in Fig.1. Then, based on the equivalence between StO' and $\mathsf{H}[\oplus]$ as well as $\mathsf{Reprogram}_0$ and $\mathsf{Repro}^{\diamond}$, we obtain game $\mathbf{G_3}$. Meanwhile, game $\mathbf{G_3}$ is identical to game $\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},1}(\mathcal{A})$ defined in Fig. 3, and thus we have

$$\Pr\left[\mathbf{G}_2\right] = \Pr\left[\mathbf{G}_3\right] = \Pr\left[\mathbf{G}^{\mathsf{ghhm}}_{\mathcal{E},1}(\mathcal{A})\right]. \tag{18}$$

By combining Eq. (13), Eq. (14), Eq. (17), and Eq. (18), we finally get

$$\mathsf{Adv}^{\mathsf{ghhm}}_{\mathcal{E},e}(\mathcal{A}) \leq 2\sqrt{2} \cdot \sqrt{\sum_{j=1}^{r} \hat{q}_j \cdot p^j_{\max}}$$

for $e \in \{1, 1/2\}$. This completes the proof of Theorem 7. □

Similar to Corollary 3, in the common case where a constant $p_{\max}$ bounds $p^i_{\max}$ for all $i \in [r]$, Theorem 7 implies the following tighter corollary:

Corollary 4. *Suppose that $\mathcal{A}$ makes a total of q (quantum) queries to its oracle, hence $\hat{q}_i \leq q$ for all $i \in [r]$. Further, assume that $p^i_{\max} \leq p_{\max}$ for all $i \in [r]$. Then we have $\mathsf{Adv}^{\mathsf{ghhm}}_{\mathcal{E},e}(\mathcal{A}) \leq 2\sqrt{2}\sqrt{rq \cdot p_{\max}}$ for $e \in \{1, 1/2\}$.*

5 The ABKM Resampling Theorem is Implied by O2H

Alagic, Bai, Katz, and Majenz [1] proved a resampling theorem for two-way accessible random permutations. This theorem can be viewed as an extension of the GHHM adaptive reprogramming theorem (Theorem 6) to the setting of random permutations. In this section, we prove that this theorem is implied by our DOFP-O2H theorem.

Let P be a permutation on $\{0,1\}^n$, and let P^{-1} denote its inverse. Let $\mathsf{Perm}(n)$ denote the set of all permutations on $\{0,1\}^n$. Let $S_{s,s'}$ be the permutation swapping s and s', i.e., $S_{s,s'}(s) = s'$, $S_{s,s'}(s') = s$, and $S_{s,s'}(x) = x$ for $\forall x \notin \{s, s'\}$. Then the ABKM resampling game is defined in Fig. 6, and the following theorem was proved by Alagic et al. [1]:

Theorem 8 (ABKM Resampling, [1, Lemma 5]). *Let $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ be a two-stage adversary where $\mathcal{A}_0$ makes q_0 oracle queries. Define*

$$\mathsf{Adv}^{\mathsf{abkm}}(\mathcal{A}) := \left|\Pr\left[\mathbf{G}^{\mathsf{abkm}}_0(\mathcal{A})\right] - \Pr\left[\mathbf{G}^{\mathsf{abkm}}_1(\mathcal{A})\right]\right|.$$

Then

$$\mathsf{Adv}^{\mathsf{abkm}}(\mathcal{A}) \leq 4 \cdot \sqrt{\frac{q_0}{2^n}}.$$

Game $\mathbf{G}_b^{\mathsf{abkm}}(\mathcal{A})$ $(b \in \{0,1\})$

1. $\mathsf{P}_0 \leftarrow_{\$} \mathsf{Perm}(n)$
2. $\mathcal{A}_0^{\mathsf{P}_0[\oplus],\mathsf{P}_0^{-1}[\oplus]}$
3. $s \leftarrow_{\$} \{0,1\}^n, s' \leftarrow_{\$} \{0,1\}^n$
4. $\mathsf{P}_1 := \mathsf{P}_0 \circ S_{s,s'}$
5. $b' \leftarrow \mathcal{A}_1^{\mathsf{P}_b[\oplus],\mathsf{P}_b^{-1}[\oplus]}(s,s')$
6. **return** $[\![b'=1]\!]$

Fig. 6. The ABKM resampling game. Here, the adversary $\mathcal{A}$ is divided into two stages, namely, $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$.

In the following, we reprove Theorem 8 by applying our DOFP-O2H theorem. The new proof achieves the same upper bound and is also valid for the difference between square-root success probabilities.

Theorem 9. *Let $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ be an adversary for which $\mathcal{A}_0$ makes q_0 oracle queries. For $e \in \{1, 1/2\}$, define*

$$\mathsf{Adv}_e^{\mathsf{abkm}}(\mathcal{A}) := \left| \left(\Pr\left[\mathbf{G}_0^{\mathsf{abkm}}(\mathcal{A})\right]\right)^e - \Pr\left[\left(\mathbf{G}_1^{\mathsf{abkm}}(\mathcal{A})\right]\right)^e \right|.$$

Then we have

$$\mathsf{Adv}_e^{\mathsf{abkm}}(\mathcal{A}) \leq 4 \cdot \sqrt{\frac{q_0}{2^n}}.$$

Proof. Here, we introduce a sequence of hybrid games $\mathbf{G_0}$ through $\mathbf{G_3}$ as shown in Fig. 7 to complete the proof.

Game $\mathbf{G_0}$: This game is the same as game $\mathbf{G}_0^{\mathsf{abkm}}(\mathcal{A})$ defined in Fig. 6, except that we sample P_0, s, s' and P_1 at the beginning. It then follows that

$$\Pr\left[\mathbf{G}_0^{\mathsf{abkm}}(\mathcal{A})\right] = \Pr\left[\mathbf{G_0}\right]. \tag{19}$$

Game $\mathbf{G_1}$: In this game, the $\mathcal{A}_0$ with quantum oracle access to $\mathsf{P}_0[\oplus]$ and $\mathsf{P}_0^{-1}[\oplus]$ is replaced by $\mathcal{B}$ with quantum oracle access to PERM_0. Here, PERM_0 maps $|x, y, d, \mathsf{P}_0, s, s'\rangle$ into $|\mathsf{P}(x, y, d, \mathsf{P}_0, s, s')\rangle$, where P is a permutation defined as

$$\mathsf{P}(x, y, d, \mathsf{P}_0, s, s') = \begin{cases} (x, y \oplus \mathsf{P}_0(x), d, \mathsf{P}_0, s, s') & \text{if } d = 1, \\ (x, y \oplus \mathsf{P}_0^{-1}(x), d, \mathsf{P}_0, s, s') & \text{if } d = 0. \end{cases}$$

$\mathcal{B}$ runs $\mathcal{A}_0$ and simulates $\mathsf{P}_0[\oplus]$ and $\mathsf{P}_0^{-1}[\oplus]$ by querying PERM_0. That is, when $\mathcal{A}_0$ queries $\mathsf{P}_0[\oplus]$ (resp. $\mathsf{P}_0^{-1}[\oplus]$) with the query state $|x, y\rangle$, $\mathcal{B}$ sets $d = 1$ (resp. $d = 0$) and queries PERM_0 with the query state $|x, y, 1, \mathsf{P}_0, s, s'\rangle$ (resp. $|x, y, 0, \mathsf{P}_0, s, s'\rangle$).

Game $\mathbf{G_0}$	**Game $\mathbf{G_1}$**
1. $s \leftarrow_\$ \{0,1\}^n, s' \leftarrow_\$ \{0,1\}^n$ $\mathsf{P}_0 \leftarrow_\$ \mathsf{Perm}(n), \mathsf{P}_1 := \mathsf{P}_0 \circ S_{s,s'}$	1. $s \leftarrow_\$ \{0,1\}^n, s' \leftarrow_\$ \{0,1\}^n$ $\mathsf{P}_0 \leftarrow_\$ \mathsf{Perm}(n), \mathsf{P}_1 := \mathsf{P}_0 \circ S_{s,s'}$
2. $\mathcal{A}_0^{\mathsf{P}_0[\oplus],\mathsf{P}_0^{-1}[\oplus]}$	2. $\mathcal{B}^{\mathsf{PERM}_0}$
3. $b' \leftarrow \mathcal{A}_1^{\mathsf{P}_0[\oplus],\mathsf{P}_0^{-1}[\oplus]}(s,s')$	3. $b' \leftarrow \mathcal{A}_1^{\mathsf{P}_0[\oplus],\mathsf{P}_0^{-1}[\oplus]}(s,s')$
4. **return** $[\![b'=1]\!]$	4. **return** $[\![b'=1]\!]$
Game $\mathbf{G_2}$	**Game $\mathbf{G_3}$**
1. $s \leftarrow_\$ \{0,1\}^n, s' \leftarrow_\$ \{0,1\}^n$ $\mathsf{P}_0 \leftarrow_\$ \mathsf{Perm}(n), \mathsf{P}_1 := \mathsf{P}_0 \circ S_{s,s'}$	1. $s \leftarrow_\$ \{0,1\}^n, s' \leftarrow_\$ \{0,1\}^n$ $\mathsf{P}_0 \leftarrow_\$ \mathsf{Perm}(n), \mathsf{P}_1 := \mathsf{P}_0 \circ S_{s,s'}$
2. $\mathcal{B}^{\mathsf{PERM}_1}$	2. $\mathcal{A}_0^{\mathsf{P}_1[\oplus],\mathsf{P}_1^{-1}[\oplus]}$
3. $b' \leftarrow \mathcal{A}_1^{\mathsf{P}_0[\oplus],\mathsf{P}_0^{-1}[\oplus]}(s,s')$	3. $b' \leftarrow \mathcal{A}_1^{\mathsf{P}_0[\oplus],\mathsf{P}_0^{-1}[\oplus]}(s,s')$
4. **return** $[\![b'=1]\!]$	4. **return** $[\![b'=1]\!]$

Fig. 7. The ABKM resampling game. Here, the adversary $\mathcal{A}$ is divided into two stages, namely, $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$.

It is easy to see that $\mathcal{B}$ perfectly simulates $\mathcal{A}_0$ and $\mathcal{B}$ makes q_0 queries to PERM_0. Thus we have

$$\Pr[\mathbf{G_0}] = \Pr[\mathbf{G_1}]. \tag{20}$$

Game $\mathbf{G_2}$: This game is the same as $\mathbf{G_1}$, except that the PERM_0 is replaced with PERM_1, which maps $|x, y, d, \mathsf{P}_0, s, s'\rangle$ into $|\mathsf{P}'(x, y, d, \mathsf{P}_0, s, s')\rangle$ where P' is a permutation defined as

$$\mathsf{P}'(x, y, d, \mathsf{P}_0, s, s') = \begin{cases} (x, y \oplus \mathsf{P}_0 \circ S_{s,s'}(x), d, \mathsf{P}_0, s, s') & \text{if } d = 1, \\ (x, y \oplus (\mathsf{P}_0 \circ S_{s,s'})^{-1}(x), d, \mathsf{P}_0, s, s') & \text{if } d = 0. \end{cases}$$

It can be observed that P and P' defined above are two fixed permutations, and they differ only on the input $(x, y, d, \mathsf{P}_0, s, s')$ where

$$(x, d) \in \{(s, 1), (s', 1), (\mathsf{P}_0(s), 0), (\mathsf{P}_0(s'), 0)\}. \tag{21}$$

Hence, by applying Theorem 5, for $e \in \{1, 1/2\}$ we have

$$\begin{aligned} |(\Pr[\mathbf{G}_1])^e - (\Pr[\mathbf{G}_2])^e| \leq & \sqrt{2q_0 \cdot \Pr[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{C}^{\mathsf{PERM}_0}]} \\ & + \sqrt{2q_0 \cdot \Pr[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{C}^{\mathsf{PERM}_1}]}. \end{aligned} \tag{22}$$

Here $\mathcal{C}^{\mathcal{O}}$ ($\mathcal{O} \in \{\mathsf{PERM}_0, \mathsf{PERM}_1\}$ is a quantum oracle algorithm that first samples $i \leftarrow_\$ [q_0]$, then runs $\mathcal{B}$ until (just before) its i-th query to $\mathcal{O}$, and finally measures

$\mathcal{C}^{\mathsf{PERM}_0}$

1. $i \leftarrow_{\$} [q_0], s \leftarrow_{\$} \{0,1\}^n, s' \leftarrow_{\$} \{0,1\}^n, \mathsf{P}_0 \leftarrow_{\$} \mathsf{Perm}(n), \mathsf{P}_1 := \mathsf{P}_0 \circ S_{s,s'}$
2. run $\mathcal{A}_0^{\mathsf{P}_0[\oplus],\mathsf{P}_0^{-1}[\oplus]}$ until (just before) its i-th query to $\mathsf{P}_0[\oplus]$ or $\mathsf{P}_0^{-1}[\oplus]$
   measure the query register to get (x, y)
   if (x, y) is a query to $\mathsf{P}_0[\oplus]$
      return $w = (x, y, 1, \mathsf{P}_0, s, s')$
   else return $w = (x, y, 0, \mathsf{P}_0, s, s')$

$\mathcal{C}^{\mathsf{PERM}_1}$

1. $i \leftarrow_{\$} [q_0], s \leftarrow_{\$} \{0,1\}^n, s' \leftarrow_{\$} \{0,1\}^n, \mathsf{P}_0 \leftarrow_{\$} \mathsf{Perm}(n), \mathsf{P}_1 := \mathsf{P}_0 \circ S_{s,s'}$
2. run $\mathcal{A}_0^{\mathsf{P}_1[\oplus],\mathsf{P}_1^{-1}[\oplus]}$ until (just before) its i-th query to $\mathsf{P}_1[\oplus]$ or $\mathsf{P}_1^{-1}[\oplus]$
   measure the query register to get (x, y)
   if (x, y) is a query to $\mathsf{P}_1[\oplus]$
      return $w = (x, y, 1, \mathsf{P}_0, s, s')$
   else return $w = (x, y, 0, \mathsf{P}_0, s, s')$

Fig. 8. Quantum oracle algorithms $\mathcal{C}^{\mathsf{PERM}_0}$ and $\mathcal{C}^{\mathsf{PERM}_1}$.

this query to get its output w. Then, based on the definition of $\mathcal{B}$ given earlier, $\mathcal{C}^{\mathsf{PERM}_0}$ and $\mathcal{C}^{\mathsf{PERM}_1}$ can be equivalently reformulated as shown in Fig. 8.

For the output $w = (x, y, d, \mathsf{P}_0, s, s')$, it follows from Eq. (21) that $\mathsf{P}(w) \neq \mathsf{P}'(w)$ implies $x \in \{s, s'\}$ if $d = 1$ and $x \in \{\mathsf{P}_0(s), \mathsf{P}_0(s')\}$ if $d = 0$. Hence, if $\mathcal{C}^{\mathsf{PERM}_0}$ indeed outputs a $w = (x, y, d, \mathsf{P}_0, s, s')$ satisfying $\mathsf{P}(w) \neq \mathsf{P}'(w)$, we can deduce that the element x in w belongs to a special set: x must be either in $\{s, s'\}$ if (x, y) is a $\mathsf{P}_0[\oplus]$ query, or in $\{\mathsf{P}_0(s), \mathsf{P}_0(s')\}$ if (x, y) is a $\mathsf{P}_0^{-1}[\oplus]$ query. This allows us to correctly guess an element from one of these two sets, even though the internal adversary $\mathcal{A}_0$ is independent of them. Since s and s' are uniformly sampled from $\{0,1\}^n$, by a simple probability analysis we have

$$\Pr\left[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{C}^{\mathsf{PERM}_0}\right] \leq \frac{2}{2^n}. \tag{23}$$

As for $\mathcal{C}^{\mathsf{PERM}_1}$, according to the definition of PERM_1 one can notice that $\mathcal{C}^{\mathsf{PERM}_1}$ actually simulates the process of $\mathcal{A}_0$ querying $\mathsf{P}_1[\oplus]$ and $\mathsf{P}_1^{-1}[\oplus]$, where $\mathsf{P}_1 = \mathsf{P}_0 \circ S_{s,s'}$. Here we point out that the independence of $\mathcal{A}_0$ from $\{s, s'\}$ and $\{\mathsf{P}_0(s), \mathsf{P}_0(s')\}$ is preserved, since P_0 is drawn uniformly from $\mathsf{Perm}(n)$, and the mapping $\mathsf{P} \to \mathsf{P} \circ S_{s,s'}$ is a permutation of $\mathsf{Perm}(n)$ itself. Hence, similar to $\mathcal{C}^{\mathsf{PERM}_0}$, we also have

$$\Pr\left[\mathsf{P}(w) \neq \mathsf{P}'(w) : w \leftarrow \mathcal{C}^{\mathsf{PERM}_1}\right] \leq \frac{2}{2^n}. \tag{24}$$

Now by combining Eq. (22), Eq. (23), and Eq. (24), for $e \in \{1, 1/2\}$, we have

$$|(\Pr[\mathbf{G_1}])^e - (\Pr[\mathbf{G_2}])^e| \leq 4 \cdot \sqrt{\frac{q_0}{2^n}}. \tag{25}$$

Game $\mathbf{G_3}$: In this game, the $\mathcal{B}$ with quantum oracle access to PERM_1 is replaced with the $\mathcal{A}_0$ with quantum oracle access to $\mathsf{P}_1[\oplus]$ and $\mathsf{P}_1^{-1}[\oplus]$.

Based on the definition of $\mathcal{B}$ given earlier, $\mathcal{B}^{\mathsf{PERM}_1}$ perfectly simulates $\mathsf{P}_1[\oplus]$ and $\mathsf{P}_1^{-1}[\oplus]$ for $\mathcal{A}_0$[6], and thus

$$\Pr[\mathbf{G_2}] = \Pr[\mathbf{G_3}]. \tag{26}$$

As for game $\mathbf{G_3}$, it is not hard to check that it is equivalent to game $\mathbf{G}_1^{\mathsf{abkm}}(\mathcal{A})$ defined in Fig. 6. Hence

$$\Pr[\mathbf{G_3}] = \Pr\left[\mathbf{G}_1^{\mathsf{abkm}}(\mathcal{A})\right]. \tag{27}$$

Finally, by combining Eq. (19), Eq. (20), Eq. (25), and Eq. (27), we can obtain

$$\mathsf{Adv}_e^{\mathsf{abkm}}(\mathcal{A}) \leq 4 \cdot \sqrt{\frac{q_0}{2^n}}$$

for $e \in \{1, 1/2\}$. This completes the proof of Theorem 9. □

6 Acknowledgement

We thank the anonymous reviewers, especially the shepherd of this article, for helpful comments that significantly improved the quality of this paper. We also thank Yu-Hsuan Huang for insightful discussions on generalizing the analysis of the matching attack to reprogramming multiple points. This work is supported by National Natural Science Foundation of China (62302496, 62172405), and the Key Research Program of the Chinese Academy of Science, Grant No. ZDRW-XX-2022-1.

[6] This property is also utilized when reformulating $\mathcal{C}^{\mathsf{PERM}_1}$ as shown in Fig. 8.

References

1. Alagic, G., Bai, C., Katz, J., Majenz, C.: Post-quantum security of the even-mansour cipher. 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. Lecture Notes in Computer Science, vol. 13277, pp. 458–487. Springer (2022). https://doi.org/10.1007/978-3-031-07082-2_17
2. Alagic, G., Bai, C., Katz, J., Majenz, C., Struck, P.: Post-quantum security of tweakable even-mansour, and applications. In: Advances in Cryptology - EUROCRYPT 2024 - 43rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zurich, Switzerland, May 26-30, 2024, Proceedings, Part I. Lecture Notes in Computer Science, vol. 14651, pp. 310–338. Springer (2024). https://doi.org/10.1007/978-3-031-58716-0_11
3. 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
4. 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
5. 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
6. 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
7. Carolan, J.: Compressed permutation oracles. IACR Cryptol. ePrint Arch. p. 1734 (2025), https://eprint.iacr.org/2025/1734
8. Devevey, J., Fallahpour, P., Passelègue, A., Stehlé, D.: A detailed analysis of fiat-shamir with aborts. In: Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part V. Lecture Notes in Computer Science, vol. 14085, pp. 327–357. Springer (2023). https://doi.org/10.1007/978-3-031-38554-4_11
9. Don, J., Fehr, S., Huang, Y., Struck, P.: On the (in)security of the BUFF transform. In: Advances in Cryptology - CRYPTO 2024 - 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2024, Proceedings, Part I. Lecture Notes in Computer Science, vol. 14920, pp. 246–275. Springer (2024). https://doi.org/10.1007/978-3-031-68376-3_8
10. Eaton, E.: Leighton-micali hash-based signatures in the quantum random-oracle model. In: Adams, C., Camenisch, J. (eds.) Selected Areas in Cryptography - SAC 2017 - 24th International Conference, Ottawa, ON, Canada, August 16-18, 2017, Revised Selected Papers. Lecture Notes in Computer Science, vol. 10719, pp. 263–280. Springer (2017). https://doi.org/10.1007/978-3-319-72565-9_13

11. Fallahpour, P., Fehr, S., Huang, Y.: Tighter quantum security for fiat-shamir-with-aborts and hash-and-sign-with-retry signatures. IACR Cryptol. ePrint Arch. p. 985 (2025), https://eprint.iacr.org/2025/985
12. Ge, J., Liao, H., Xue, R.: Measure-rewind-extract: Tighter proofs of one-way to hiding and CCA security in the quantum random oracle model. In: Advances in Cryptology - ASIACRYPT 2024 - 30th International Conference on the Theory and Application of Cryptology and Information Security, Kolkata, India, December 9-13, 2024, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 15487, pp. 3–34. Springer (2024). https://doi.org/10.1007/978-981-96-0894-2_1
13. Grilo, A.B., Hövelmanns, K., Hülsing, A., Majenz, C.: Tight adaptive reprogramming in the QROM. 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. 637–667. Springer (2021). https://doi.org/10.1007/978-3-030-92062-3_22
14. Jaeger, J.: Nonadaptive one-way to hiding implies adaptive quantum reprogramming. Cryptology ePrint Archive, Paper 2024/797, to appear in Proceedings of ASIACRYPT 2025, https://eprint.iacr.org/2024/797
15. Kosuge, H., Xagawa, K.: Probabilistic hash-and-sign with retry in the quantum random oracle model. In: Public-Key Cryptography - PKC 2024 - 27th IACR International Conference on Practice and Theory of Public-Key Cryptography, Sydney, NSW, Australia, April 15-17, 2024, Proceedings, Part I. Lecture Notes in Computer Science, vol. 14601, pp. 259–288. Springer (2024). https://doi.org/10.1007/978-3-031-57718-5_9
16. 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
17. Liao, H., Ge, J., Cao, S., Xue, R.: New proof for plain OAEP: post-quantum security without parameter restrictions or collision-resistance. IACR Cryptol. ePrint Arch. p. 1288 (2025), https://eprint.iacr.org/2025/1288
18. Morimae, T., Yamakawa, T.: Classically verifiable NIZK for QMA with preprocessing. 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. Lecture Notes in Computer Science, vol. 13794, pp. 599–627. Springer (2022). https://doi.org/10.1007/978-3-031-22972-5_21
19. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information (10th Anniversary edition). Cambridge University Press (2016)
20. Pan, J., Zeng, R.: Selective opening security in the quantum random oracle model, revisited. In: Public-Key Cryptography - PKC 2024 - 27th IACR International Conference on Practice and Theory of Public-Key Cryptography, Sydney, NSW, Australia, April 15-17, 2024, Proceedings, Part III. pp. 92–122. Springer (2024). https://doi.org/10.1007/978-3-031-57725-3_4
21. Unruh, D.: Quantum position verification in the random oracle model. In: Garay, J.A., Gennaro, R. (eds.) Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part II. Lecture Notes in Computer Science, vol. 8617, pp. 1–18. Springer (2014). https://doi.org/10.1007/978-3-662-44381-1_1

22. Unruh, D.: Non-interactive zero-knowledge proofs in the quantum random oracle model. In: Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9057, pp. 755–784. Springer (2015). https://doi.org/10.1007/978-3-662-46803-6_25
23. Unruh, D.: Revocable quantum timed-release encryption. J. ACM **62**(6), 49:1–49:76 (2015). https://doi.org/10.1145/2817206
24. Unruh, D.: Compressed permutation oracles (and the collision-resistance of sponge/sha3). IACR Cryptol. ePrint Arch. p. 62 (2021), https://eprint.iacr.org/2021/062
25. Unruh, D.: Towards compressed permutation oracles. In: Advances in Cryptology - ASIACRYPT 2023 - 29th International Conference on the Theory and Application of Cryptology and Information Security, Guangzhou, China, December 4-8, 2023, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 14441, pp. 369–400. Springer (2023). https://doi.org/10.1007/978-981-99-8730-6_12
26. Yuan, Q., Sun, C., Takagi, T.: Revisiting the security of fiat-shamir signature schemes under superposition attacks. In: Information Security and Privacy - 29th Australasian Conference, ACISP 2024, Sydney, NSW, Australia, July 15-17, 2024, Proceedings, Part II. Lecture Notes in Computer Science, vol. 14896, pp. 164–184. Springer (2024). https://doi.org/10.1007/978-981-97-5028-3_9
27. 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

A 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. A quantum state on Q is a vector on $\mathcal{H}_Q$. 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 state if $\||\psi\rangle\| = 1$. The norm of an operator A is defined as $\|A\| := \max_{|\psi\rangle} \|A|\psi\rangle\|$, where max is over all unit states on Q. Sometimes we also write $|\psi\rangle$ as $|\psi\rangle_Q$ to emphasize that $|\psi\rangle$ is a quantum state on Q.

Composite quantum system. Given quantum systems Q_1 and Q_2, we call tensor product $Q_1 \otimes Q_2$ to be the composite quantum system and the product state to be $|\psi_1\rangle \otimes |\psi_2\rangle \in Q_1 \otimes Q_2$, 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 a 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 $|\psi'\rangle = \mathrm{U}|\psi\rangle$. For any unitary operation U, we have $\mathrm{U}\mathrm{U}^\dagger = \mathrm{I}$, where $\mathrm{U}^\dagger$ 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_S ($S \subseteq X$) applies the operation $\sum_{y\in S} |y\rangle\langle y|$ to $|\psi\rangle$ to get the new state $\sum_{y\in S} |y\rangle\langle y|\psi\rangle$. 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 is obtained, the state $|\psi\rangle$ collapses to state $|x\rangle$. Moreover, we explain an important special class of measurements called projective measurement which is defined by a set of projectors $\mathrm{M}_1, \ldots, \mathrm{M}_n$ 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 i ($i \in \{1, \ldots, n\}$) is $\langle\psi|\mathrm{M}_i|\psi\rangle$ ($= \|\mathrm{M}_i|\psi\rangle\|^2$). Given that result i is obtained, the state $|\psi\rangle$ collapses to the state $\mathrm{M}_i|\psi\rangle/\sqrt{\langle\psi|\mathrm{M}_i|\psi\rangle}$. In our paper, unless otherwise specified, we only use measurements in the computational basis.

Quantum oracle algorithm. 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 $b \in \{0,1\}$. We say that a quantum oracle algorithm $\mathcal{A}^{\mathsf{H}}(z)$ is a unitary quantum oracle algorithm if its entire execution can be described as

$$\mathbb{M}_{\mathcal{A}} \circ \mathrm{U}_q \circ \mathrm{O}_{\mathsf{H}} \circ \mathrm{U}_{q-1} \circ \mathrm{O}_{\mathsf{H}} \cdots \mathrm{U}_2 \circ \mathrm{O}_{\mathsf{H}} \circ \mathrm{U}_1 \circ \mathrm{O}_{\mathsf{H}} |\psi_z\rangle.$$

Here $|\psi_z\rangle$ is an initial pure state that may depend on input z, $\mathrm{U}_1, \ldots, \mathrm{U}_q$ are the fixed unitary operations applied between oracle queries, O_{H} is the unitary operation used to implement the oracle H. $\mathbb{M}_{\mathcal{A}} := \{\mathrm{M}_0, \mathrm{M}_1\}$ 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 $\mathcal{A}$. Next, we introduce the following well-known fact about the quantum oracle algorithm. This fact implies that we can directly assume the quantum oracle algorithms to be unitary.

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

B A Detailed Proof of Eq. (12)

Let $\mathbf{G}_{\mathsf{ow},i}(\mathcal{B}^{\mathsf{P}^0_{\pm}[\oplus]})$ ($i \in [2q]$) be the sub-game that internally runs $\mathcal{B}^{\mathsf{P}^0_{\pm}[\oplus]}$ until (just before) its i-th query to $\mathsf{P}^0_{\pm}[\oplus]$, and then measures the query register to get $(b_i, x_i, y_i) \in \{0,1\} \times \{0,1\}^n \times \{0,1\}^n$ and finally outputs $[\![\mathsf{P}^0_{\pm}[\oplus](b_i, x_i, y_i)) \neq \mathsf{P}^1_{\pm}[\oplus](b_i, x_i, y_i))]\!]$. Hence, we have

$$\begin{aligned}\Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}^0_{\pm}[\oplus],\mathsf{P}^1_{\pm}[\oplus]}(\mathcal{B}^{\mathsf{P}^0_{\pm}[\oplus]})\right] &= \sum_{i=1}^{2q} \frac{1}{2q} \cdot \Pr\left[\mathbf{G}_{\mathsf{ow},i}(\mathcal{B}^{\mathsf{P}^0_{\pm}[\oplus]})\right] \\ &\overset{(a)}{=} \sum_{i=1}^{2q} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}^0_{\pm}[\oplus](b_i, x_i, y_i) \neq \mathsf{P}^1_{\pm}[\oplus](b_i, x_i, y_i)\right].\end{aligned}$$

Here (a) is merely a re-representation intended to enhance the clarity. Specifically, it allows us to explicitly express the measurement result (b_i, x_i, y_i) in the probability expression. Similarly, for $i \in [q]$ we can define the sub-game $\mathbf{G}_{\mathsf{ow},i}(\mathcal{A}^{\mathsf{P}_0})$ and obtain that

$$\Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_0})\right] = \sum_{i=1}^{q} \frac{1}{q} \cdot \Pr\left[\mathbf{G}_{\mathsf{ow},i}(\mathcal{A}^{\mathsf{P}_0})\right].$$

From the construction of $\mathcal{B}$, it follows that $\mathcal{B}$ makes $2q$ queries to its oracle $\mathsf{P}^0_{\pm}[\oplus]$, and these queries have the following two properties:

1. The q queries at odd positions have input/output register BX/Y, while the state of register B must be $|1\rangle$ and the state of register Y must be $|0^n\rangle$.
2. The other q queries at even positions have input/output register BY/X, while the state of register B must be $|0\rangle$ and the state of registers YX must be $\sum_x |\mathsf{P}_0(x), x\rangle$. That is to say, the value of X must be the inverse (under permutation P_0) of that of Y.

Based on the above two properties, and by defining two sets $[2q_1] := \{1, 3, \cdots 2q-1\}$ and $[2q_0] := \{2, 4, \cdots, 2q\}$, we can compute

$$\begin{aligned}&\Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}^0_{\pm}[\oplus],\mathsf{P}^1_{\pm}[\oplus]}(\mathcal{B}^{\mathsf{P}^0_{\pm}[\oplus]})\right] \\ =&\sum_{i=1}^{2q} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}^0_{\pm}[\oplus](b_i, x_i, y_i) \neq \mathsf{P}^1_{\pm}[\oplus](b_i, x_i, y_i)\right] \\ =&\sum_{i\in[2q_1]} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}^0_{\pm}[\oplus](1, x_i, 0^n) \neq \mathsf{P}^1_{\pm}[\oplus](1, x_i, 0^n)\right] \\ &+\sum_{i\in[2q_0]} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}^0_{\pm}[\oplus](0, \mathsf{P}_0(x_i), x_i) \neq \mathsf{P}^1_{\pm}[\oplus](0, \mathsf{P}_0(x_i), x_i)\right]\end{aligned}$$

$$
\begin{aligned}
&\overset{(a)}{=} \sum_{i\in[2q_1]} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}_0(x_i) \neq \mathsf{P}_1(x_i) : \begin{array}{l}(1, x_i, 0^n) \textit{ is the measurement}\\ \textit{result of } \mathcal{B}\textit{'s } i\textit{-th } \mathsf{P}^0_{\pm}[\oplus] \textit{ query.}\end{array}\right] \\
&\quad + \sum_{i\in[2q_0]} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}_0(x_i) \neq \mathsf{P}_1(x_i) : \begin{array}{l}(0, \mathsf{P}_0(x_i), x_i) \textit{ is the measurement}\\ \textit{result of } \mathcal{B}\textit{'s } i\textit{-th } \mathsf{P}^0_{\pm}[\oplus] \textit{ query.}\end{array}\right] \\
&\overset{(b)}{=} \sum_{i=1}^{q} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}_0(x_i) \neq \mathsf{P}_1(x_i) : \begin{array}{l}x_i \textit{ is the measurement}\\ \textit{result of } \mathcal{A}\textit{'s } i\textit{-th } \mathsf{P}_0 \textit{ query.}\end{array}\right] \\
&\quad + \sum_{i=1}^{q} \frac{1}{2q} \cdot \Pr\left[\mathsf{P}_0(x_i) \neq \mathsf{P}_1(x_i) : \begin{array}{l}x_i \textit{ is the measurement}\\ \textit{result of } \mathcal{A}\textit{'s } i\textit{-th } \mathsf{P}_0 \textit{ query.}\end{array}\right] \\
&\overset{(c)}{=} \sum_{i=1}^{q} \frac{1}{q} \cdot \Pr\left[\mathbf{G}_{\mathsf{ow},i}(\mathcal{A}^{\mathsf{P}_0})\right] = \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_0})\right].
\end{aligned}
$$

Here (a) follows from the definition of $\mathsf{P}^0_{\pm}$ and $\mathsf{P}^1_{\pm}$ given in Eq. (10), and (c) follows from the definition of $\mathbf{G}_{\mathsf{ow},i}(\mathcal{A}^{\mathsf{P}_0})$. As for the equation (b), first, it can be observed that the two probability expressions derived from (a) are actually insensitive to the values of registers BY; they only focus on whether the value x_i of register X satisfies $\mathsf{P}_0(x_i) \neq \mathsf{P}_1(x_i)$. Furthermore, according to the construction of $\mathcal{B}$, it perfectly simulates a P_0 query (with query register X) of $\mathcal{A}$ through two consecutive $\mathsf{P}^0_{\pm}[\oplus]$ queries, and these two queries must share a identical state of register X. This implies that the result obtained by measuring the i-th ($i \in [q]$) P_0 query of $\mathcal{A}$ and the result obtained by measuring the $(2i-1)$-th or $2i$-th $\mathsf{P}^0_{\pm}[\oplus]$ queries of $\mathcal{B}$ have the same distribution in terms of the value of register X. Consequently, equation (b) holds, meaning that under the flag event $\mathsf{P}_0(x_i) \neq \mathsf{P}_1(x_i)$, we can directly convert the q measurements for $\mathcal{B}$ corresponding to sets $[2q_1]$ or $[2q_0]$ into q measurements for $\mathcal{A}$ on register X.

So we finally prove that

$$
\Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_0})\right] = \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}^0_{\pm}[\oplus],\mathsf{P}^1_{\pm}[\oplus]}(\mathcal{B}^{\mathsf{P}^0_{\pm}[\oplus]})\right].
$$

As for the proof of

$$
\Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}_0,\mathsf{P}_1}(\mathcal{A}^{\mathsf{P}_1})\right] = \Pr\left[\mathbf{G}^{\mathsf{ow}}_{\mathsf{P}^0_{\pm}[\oplus],\mathsf{P}^1_{\pm}[\oplus]}(\mathcal{B}^{\mathsf{P}^1_{\pm}[\oplus]})\right],
$$

it is almost the same and we omit it for the sake of simplicity. This completes the proof of Eq. (12). □

C A more detailed discussion of the matching attack given in [13]

In [13] it is claimed that the GHHM reprogramming theorem is tight, and a matching attack is given. In this appendix, we provide a detailed discussion about the advantage of that concrete attacker, and then claim that our new bound does not conflict with that matching attack.

C.1 Recalling the attack matching the single-reprogramming case ($r = 1$)

We begin by restating the concrete attacker D from the original GHHM paper [13]. As is shown in Fig. 9, D makes q (quantum) queries to O and q (quantum) queries to O', where O/O' denotes the oracle before/after the potential reprogramming. σ is a fixed cyclic permutation on $\{0,1\}^n$. For the reprogramming point x^*, first define the set $S = \left\{x^*, \sigma^{-1}(x^*), \ldots, \sigma^{-(q-1)}(x^*)\right\}$ and its component $\bar{S} = \{0,1\}^n \backslash S$. The measurement is then defined by two projectors Π_0 and Π_1, where $\Pi_0 = \frac{1}{2}(|S\rangle + |\bar{S}\rangle)(\langle S| + \langle \bar{S}|)$ and $\Pi_1 = \mathrm{I} - \Pi_0$[7]. Here $|S\rangle$ and $|\bar{S}\rangle$ are the uniform superpositions over S and $\bar{S}$.

Before potential reprogramming:	After potential reprogramming
1. Prepare registers XY in $\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}\|x,0\rangle$.	1. Query O' using registers XY.
2. Query O using registers XY.	2. **for** $i = q-2,\ldots,0$
3. **for** $i = 1,\ldots,q-1$:	Apply σ^{-1} on register X,
Apply σ on register X,	Query O using registers XY.
Query O using registers XY.	3. Measure X register according to $\{\Pi_0,\Pi_1\}$.
	Output b if the state projects onto Π_b.

Fig. 9. (The original Fig.13) Distinguisher for a Single Reprogrammed Point. In this figure, we have adjusted the parameters to ensure that the number of oracle queries before and after the potential reprogramming is q in both cases.

Invoking the results from the original GHHM paper, the distinguishing advantage of the concrete distinguisher D can be lower-bounded as follows:

Theorem 10 (The Original Theorem 7). *For every $1 \le q \le 2^{n-3}$, the attack described in Fig. 9 can be implemmented in quantum polynomial-time. Performing q queries each before and after the potential reprogramming, it detects the reprogramming of a random oracle $\mathsf{O} : \{0,1\}^n \to \{0,1\}^m$ at a single point with probability at least $\Omega(\sqrt{\frac{q}{2^n}})$.*

[7] Consistent with the original GHHM paper, S, Π_0 and Π_1 are functions of x^*, and we also omit the dependence here since it can be assumed that x^* is fixed.

C.2 Extending the attack to the multi-reprogramming case ($r > 1$)

The structure of the matching distinguisher. We begin by describing the distinguisher for the case $r > 1$. Consider the simpler case where both the number of potentially reprogrammed points, r, and the number of queries, q, are polynomial in n. Following the description in the original GHHM paper (page 48, penultimate paragraph), the structure of the matching distinguisher should be identical to that depicted in Fig. 9, except that the measurement is defined differently, and that oracles O and O' may differ at r points. We also name the distinguisher as D.

Let $\{x_1^*, x_2^*, \cdots, x_r^*\}$ be the set of r reprogramming points, and σ be the fixed cyclic permutation on $\{0,1\}^n$. For each $i \in [r]$ define the associated set $S_i := \{x_i^*, \sigma^{-1}(x_i^*), \ldots, \sigma^{-(q-1)}(x_i^*)\}$. We can assume that the reprogramming points are fixed and, are all uniformly chosen from $\{0,1\}^n$. Then, it follows from a direct probability argument that the sets $\{S_i\}_{i\in[r]}$ are pairwise disjoint with the probability of at least $1 - \frac{2r^2q}{2^n}$[9].

Remark 2. The observation above is consistent with the original GHHM paper (page 48), which states that "since all reprogrammed values are chosen uniformly at random, the probability that there exists some $i, j \in [q]$ and potentially reprogrammed points $x, x' \in \{0,1\}^n$ such that $f_k^i(x) = f_k^j(x')$ is exponentially small". Here we substitute "reprogrammed values" with "reprogramming points". The reason is that f_k is not explicitly defined in the original text, and if we interpret it as σ^{-1}, this substitution yields a more natural description of the collision event.

Define the total set as $S = \cup_{i=1}^r S_i$, then it follows that $|S| = rq$. Similar to the original paper, define the component set $\bar{S} = \{0,1\}^n \backslash S$. Based on the explanation in the original paper (page 48), we define the measurement by the following two projectors Π_0 and Π_1:

$$\Pi_0 = \frac{1}{2}\left(|S\rangle + |\bar{S}\rangle\right)\left(\langle S| + \langle\bar{S}|\right), \Pi_1 = \mathrm{I} - \Pi_0,$$

where $|S\rangle = \frac{1}{\sqrt{rq}}\sum_{x\in S}|x\rangle$ and $|\bar{S}\rangle = \frac{1}{\sqrt{2^n - rq}}\sum_{x\in\bar{S}}|x\rangle$ are the uniform superpositions over sets S and $\bar{S}$. Meanwhile, according to the definition we have: $|S\rangle = \frac{1}{\sqrt{r}}\sum_{i=1}^r |S_i\rangle, i \in [r]$, where $|S_i\rangle$ denotes the uniform superposition over S_i.

(Efficient) implementation of of the matching attacker. It remains to demonstrate the efficiency of the matching attacker. Similar to the analysis in the original GHHM paper, the primary bottleneck lies in the implementation of the measurement on the basis (Π_0, Π_1). Following the methodology of the

[9] We emphasize that our subsequent analysis relies on this "pairwise disjoint" assumption, as it leads to a more streamlined argument.

original GHHM paper, to generate the desired superposition $|S\rangle$, we need to first construct the following state by computing σ^{-1}:

$$\frac{1}{\sqrt{rq}}\sum_{i=1}^{r}\sum_{j=0}^{q-1}|j\rangle|\sigma^{-j}(x_i^*)\rangle,$$

and subsequently erase the first register. The remaining steps proceed in a manner analogous to the original GHHM paper. Specifically, first ϵ-approximate the state $|\bar{S}\rangle$ in time $O(rq\log\frac{1}{\epsilon})$. Futhermore, construct an efficient circuit C that prepares the state $\frac{1}{\sqrt{2}}(|S\rangle+|\bar{S}\rangle)$, which finally allows us to ϵ-approximate the measurement $\{\Pi_0,\Pi_1\}$ in time $\mathsf{poly}(rq,\frac{1}{\epsilon})$.

The calculation and analysis of the distinguishing advantage. In this section, following the analysis in the original GHHM paper, we calculate the distinguishing advantage of D.

First, the state of registers XY before the final measurement is:

$$\begin{aligned}|\Psi_{XY}\rangle =&\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle|\bigoplus\nolimits_{j=0}^{q}(\mathsf{O}(\sigma^j(x))\oplus\mathsf{O}'(\sigma^j(x)))\rangle\\ \overset{(a)}{=}&\frac{1}{\sqrt{2^n}}\sum_{i=1}^{r}\sum_{x\in S_i}|x,\mathsf{O}(x_i^*)\oplus\mathsf{O}'(x_i^*)\rangle+\frac{1}{\sqrt{2^n}}\sum_{x\in\bar{S}}|x,0\rangle.\end{aligned}$$

Here equality (a) follows from the definition of D and the assumption that the sets $S_i(i\in[r])$ are pairwise disjoint (i.e., $S_i\cap S_j=\emptyset$ for any distinct $i,j\in[r]$). Furthermore, let O_i and O'_i denote $\mathsf{O}(x_i^*)$ and $\mathsf{O}'(x_i^*)$, respectively. Then we have:

$$|\Psi_{XY}\rangle=\sqrt{\frac{q}{2^n}}\sum_{i=1}^{r}|S_i,\mathsf{O}_i\oplus\mathsf{O}'_i\rangle+\sqrt{\frac{2^n-rq}{2^n}}|\bar{S},0\rangle. \tag{28}$$

To analyze the distinguishing advantage, we define E_k to denote the event that there are exactly k non-zero elements in the sequence $\{\mathsf{O}_i\oplus\mathsf{O}'_i\}_{i=1}^r$. Then it follows that:

$$\begin{aligned}\mathsf{Adv}(\mathsf{D})=&\Pr[0\leftarrow\mathsf{D}(\cdot)|\text{no reprogramming}]-\Pr[0\leftarrow\mathsf{D}(\cdot)|\text{reprogramming}]\\ =&\sum\nolimits_{k=0}^{r}\Pr\left[0\leftarrow\mathsf{D}(\cdot)\cap\mathrm{E}_k|\text{no reprogramming}\right]\\ &-\sum\nolimits_{k=0}^{r}\Pr\left[0\leftarrow\mathsf{D}(\cdot)\cap\mathrm{E}_k|\text{reprogramming}\right].\end{aligned} \tag{29}$$

For simplicity, we omit the absolute value bars on the right-hand side in the above equation.

In the no-reprogramming case, $\mathsf{O}_i=\mathsf{O}'_i$ for all $i\in[r]$, so it follows:

$$\begin{aligned}&\sum\nolimits_{k=0}^{r}\Pr\left[0\leftarrow\mathsf{D}(\cdot)\cap\mathrm{E}_k|\text{no reprogramming}\right]\\ =&\sum\nolimits_{k=0}^{r}\Pr\left[\mathrm{E}_k|\text{no reprogramming}\right]\cdot\Pr\left[0\leftarrow\mathsf{D}(\cdot)|\mathrm{E}_k\cap\text{no reprogramming}\right]\\ \overset{(a)}{=}&\Pr\left[0\leftarrow\mathsf{D}(\cdot)|\mathrm{E}_0\cap\text{no reprogramming}\right]=\Pr\left[0\leftarrow\mathsf{D}(\cdot)|\mathrm{E}_0\cap\text{reprogramming}\right],\end{aligned} \tag{30}$$

where (a) holds because in the no-reprogramming case, the event E_0 occurs with probability 1.

In the reprogramming case, for each k, there are $\binom{r}{k}$ possible combinations of positions for the non-zero elements. Denote E_k as $\bigcup_{j=1}^{\binom{r}{k}} \mathrm{E}_{k,j}$, where j indexes the distinct combinations. Then we can obtain:

$$\begin{aligned}
&\sum\nolimits_{k=0}^{r} \Pr\left[0 \leftarrow \mathsf{D}(\cdot) \cap \mathrm{E}_k \mid \text{reprogramming}\right] \\
=&\sum\nolimits_{k=0}^{r} \sum\nolimits_{j=1}^{\binom{r}{k}} \Pr\left[0 \leftarrow \mathsf{D}(\cdot) \cap \mathrm{E}_{k,j} \mid \text{reprogramming}\right] \\
=&\sum\nolimits_{k=0}^{r} \sum\nolimits_{j=1}^{\binom{r}{k}} \Pr\left[\mathrm{E}_{k,j} | \text{reprogramming}\right] \cdot \Pr\left[0 \leftarrow \mathsf{D}(\cdot) | \mathrm{E}_{k,j} \cap \text{reprogramming}\right] \\
\overset{(a)}{=}&\sum\nolimits_{k=0}^{r} \sum\nolimits_{j=1}^{\binom{r}{k}} \left(1-\frac{1}{2^m}\right)^k \left(\frac{1}{2^m}\right)^{r-k} \cdot \Pr\left[0 \leftarrow \mathsf{D}(\cdot) | \mathrm{E}_{k,j} \cap \text{reprogramming}\right].
\end{aligned} \tag{31}$$

Here (a) is due to the fact that all the reprogramming values O'_i are chosen uniformly at random from $\{0,1\}^m$.

In what follows, we proceed to calculate the probability $\Pr[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_{k,j} \cap$ reprogramming] in Eq. (31). Without loss of generality, let $\mathrm{E}_{k,1}$ denote the event that the first k elements in the sequence $\{\mathsf{O}_i \oplus \mathsf{O}'_i\}_{i=1}^r$ are non-zero. Then based on Eq. (28) and conditioned on $\mathrm{E}_{k,1}$, the state of registers XY just before the final measurement is given by:

$$|\Psi_{XY}^{(k,1)}\rangle = \sqrt{\frac{q}{2^n}} \sum_{i=1}^{k} |S_i, \mathsf{O}_i \oplus \mathsf{O}'_i\rangle + \sqrt{\frac{q}{2^n}} \sum_{i=k+1}^{r} |S_i, 0\rangle + \sqrt{\frac{2^n - rq}{2^n}} |\bar{S}, 0\rangle. \tag{32}$$

Denote the projector Π_0 as $\Pi_0 = |\Phi_{\Pi_0}\rangle\langle\Phi_{\Pi_0}|$, where

$$|\Phi_{\Pi_0}\rangle = \frac{1}{\sqrt{2}}\left(|S\rangle + |\bar{S}\rangle\right) = \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{r}} \sum_{i=1}^{r} |S_i\rangle + |\bar{S}\rangle\right).$$

For the case where $k = 0$,

$$|\Psi_{XY}^{(0,1)}\rangle = \left(\sqrt{\frac{q}{2^n}} \sum_{i=1}^{r} |S_i\rangle + \sqrt{\frac{2^n - rq}{2^n}} |\bar{S}\rangle\right) \otimes |0\rangle,$$

Then it is clear that the state on the X register can be denoted as:

$$|\Psi_{X}^{(0,1)}\rangle = \sqrt{\frac{q}{2^n}} \sum_{i=1}^{r} |S_i\rangle + \sqrt{\frac{2^n - rq}{2^n}} |\bar{S}\rangle = \sqrt{\frac{rq}{2^n}} |S\rangle + \sqrt{\frac{2^n - rq}{2^n}} |\bar{S}\rangle. \tag{33}$$

Since D outputs 0 is indeed the event that the measurement outcome is 0, it can be directly obtained that:

$$\begin{aligned}\Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_{0,1} \cap \text{reprogramming}\right] =& \left\| \langle \Phi_{\Pi_0} | \Psi_{XY}^{(0,1)} \rangle \right\|^2 \\ =& \frac{1}{2}\left(\sqrt{\frac{rq}{2^n}} + \sqrt{\frac{2^n - rq}{2^n}}\right)^2 \\ =& \frac{1}{2} + \sqrt{\frac{rq}{2^n}} \cdot \sqrt{1 - \frac{rq}{2^n}}.\end{aligned} \tag{34}$$

Note that for $k = 0$, there is only one choice for j, yielding:

$$\Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_0 \cap \text{reprogramming}\right] = \Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_{0,1} \cap \text{reprogramming}\right] \tag{35}$$

For the case where $k > 0$, the state $|\Psi_{XY}^{(k,1)}\rangle$ given in Eq. (32) exhibits entanglement between registers X and Y. Let $\rho_{XY}^{(k,1)} = |\Psi_{XY}^{(k,1)}\rangle\langle\Psi_{XY}^{(k,1)}|$ denote the density matrix of registers XY. Since the measurement is performed only on X register, in order to calculate the probability that the measurement outcome is 0, we first trace out Y register to obtain the reduced density matrix $\rho_X^{(k,1)}$ of register X. In the calculation process, $\rho_X^{(k,1)}$ differs according to whether the k non-zero elements $\{\mathsf{O}_i \oplus \mathsf{O}'_i\}_{i\in[k]}$ are pairwise distinct. Let F_k be the event that all the non-zero elements are pairwise distinct, and let $\neg\mathrm{F}_k$ denote its component. Then we have:

$$\begin{aligned}&\Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ =&\Pr\left[0 \leftarrow \mathsf{D}(\cdot) \cap \mathrm{F}_k|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ &+ \Pr\left[0 \leftarrow \mathsf{D}(\cdot) \cap \neg\mathrm{F}_k|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ =&\Pr\left[\mathrm{F}_k|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \cdot \Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{F}_k \cap \mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ &+ \Pr\left[\neg\mathrm{F}_k|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \cdot \Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\neg\mathrm{F}_k \cap \mathrm{E}_{k,1} \cap \text{reprogramming}\right].\end{aligned}$$

Here, the probability of the "collision event" $\neg\mathrm{F}_k$ can be bounded as:

$$\begin{aligned}&\Pr\left[\neg\mathrm{F}_k|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ =&\Pr\left[\exists i_1, i_2 \in [k], i_1 \neq i_2, \mathsf{O}_{i_1} \oplus \mathsf{O}'_{i_1} = \mathsf{O}_{i_2} \oplus \mathsf{O}'_{i_2}|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ =&\Pr\left[\cup_{i_1=1}^{k-1} \cup_{i_2=i_1+1}^{k} (\mathsf{O}_{i_1} \oplus \mathsf{O}'_{i_1} = \mathsf{O}_{i_2} \oplus \mathsf{O}'_{i_2})|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ \leq&\frac{k(k+1)}{2}\Pr\left[\mathsf{O}_1 \oplus \mathsf{O}'_1 = \mathsf{O}_2 \oplus \mathsf{O}'_2|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ \leq&k^2 \Pr\left[\mathsf{O}'_1 \oplus \mathsf{O}'_2 = \mathsf{O}_1 \oplus \mathsf{O}_2|\mathrm{E}_{k,1} \cap \text{reprogramming}\right] \\ \overset{(a)}{\leq}&k^2 \cdot \left(\frac{1}{2^m - 1}\right)^2 \cdot (2^m - 1) \\ \leq&\frac{r^2}{2^m - 1}.\end{aligned}$$

Here (a) holds because, under the condition $\mathrm{E}_{k,1} \cap \text{reprogramming}$, we strictly have $\mathsf{O}'_i \neq \mathsf{O}_i$ for $i \in [2]$. This restricts the sample space of each reprogrammed

value to size 2^m-1, yielding:

$$\begin{aligned}&\Pr\left[\mathsf{O}'_1 \text{ is chosen } \mid \mathrm{E}_{k,1}\cap \text{reprogramming}\right]\\ =&\Pr\left[\mathsf{O}'_2 \text{ is chosen } \mid \mathrm{E}_{k,1}\cap \text{reprogramming}\right]\\ =&\frac{1}{2^m-1}.\end{aligned}$$

Since r is polynomial bounded in n, $\frac{r^2}{2^m-1}$ is negligible. Therefore,

$$\Pr\left[\mathrm{F}_k \mid \mathrm{E}_{k,1}\cap \text{reprogramming}\right]\geq 1-\frac{r^2}{2^m-1}\approx 1, \tag{36}$$

and that

$$\begin{aligned}&\Pr\left[0\leftarrow \mathsf{D}(\cdot) \mid \mathrm{E}_{k,1}\cap \text{reprogramming}\right]\\ \approx&\Pr\left[0\leftarrow \mathsf{D}(\cdot) \mid \mathrm{F}_k\cap \mathrm{E}_{k,1}\cap \text{reprogramming}\right].\end{aligned} \tag{37}$$

Let $\rho^{(k,1)}_{X,\mathrm{F}_k}$ denote the reduced density matrix of register X conditioned on the event $\mathrm{F}_k\cap \mathrm{E}_{k,1}\cap$ reprogramming. It follows that:

$$\begin{aligned}\rho^{(k,1)}_{X,\mathrm{F}_k}=&\mathrm{Tr}_Y\left(\rho^{(k,1)}_{XY}\right)=\mathrm{Tr}_Y\left(|\Psi^{(k,1)}_{XY}\rangle\langle\Psi^{(k,1)}_{XY}|\right)\\ =&\frac{q}{2^n}\left(|S_1\rangle\langle S_1|+\cdots+|S_k\rangle\langle S_k|\right)\\ &+\frac{q}{2^n}\left(|S_{k+1}\rangle\langle S_{k+1}|+\cdots+|S_{k+1}\rangle\langle S_r|\right)+\frac{\sqrt{q(2^n-rq)}}{2^n}|S_{k+1}\rangle\langle\bar{S}|\\ &+\frac{q}{2^n}\left(|S_{k+2}\rangle\langle S_{k+1}|+\cdots+|S_{k+2}\rangle\langle S_r|\right)+\frac{\sqrt{q(2^n-rq)}}{2^n}|S_{k+2}\rangle\langle\bar{S}|\\ &+\cdots\\ &+\frac{q}{2^n}\left(|S_r\rangle\langle S_{k+1}|+\cdots+|S_r\rangle\langle S_r|\right)+\frac{\sqrt{q(2^n-rq)}}{2^n}|S_r\rangle\langle\bar{S}|\\ &+\frac{\sqrt{q(2^n-rq)}}{2^n}\left(|\bar{S}\rangle\langle S_{k+1}|+\cdots+|\bar{S}\rangle\langle S_r|\right)+\frac{2^n-rq}{2^n}|\bar{S}\rangle\langle\bar{S}|.\end{aligned} \tag{38}$$

Therefore we can obtain:

$$\begin{aligned}&\Pr\left[0\leftarrow \mathsf{D}(\cdot) \mid \mathrm{F}_k\cap \mathrm{E}_{k,1}\cap \text{reprogramming}\right]\\ =&\mathrm{Tr}\left(\Pi_0\rho^{(k,1)}_{X,\mathrm{F}_k}\right)\\ =&\langle\Phi_{\Pi_0}|\rho^{(k,1)}_{X,\mathrm{F}_k}|\Phi_{\Pi_0}\rangle\\ =&\frac{1}{\sqrt{2r}}\cdot\frac{rq}{2^n}\cdot\frac{1}{\sqrt{2r}}+\frac{1}{\sqrt{2r}}\cdot\frac{\sqrt{q(2^n-rq)}}{2^n}\cdot\frac{1}{\sqrt{2}}\\ &+(r-k)\left[(r-k-1)\cdot\frac{1}{\sqrt{2r}}\cdot\frac{q}{2^n}\cdot\frac{1}{\sqrt{2r}}+\frac{1}{\sqrt{2r}}\cdot\frac{\sqrt{q(2^n-rq)}}{2^n}\cdot\frac{1}{\sqrt{2}}\right]\\ &+(r-k)\cdot\frac{1}{\sqrt{2}}\cdot\frac{\sqrt{q(2^n-rq)}}{2^n}\cdot\frac{1}{\sqrt{2r}}+\frac{1}{\sqrt{2}}\cdot\frac{2^n-rq}{2^n}\cdot\frac{1}{\sqrt{2}}\\ =&\frac{1}{2}+\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}\cdot\frac{r-k+1}{2r}+\frac{q}{2^n}\cdot\frac{k^2+k-2kr}{r}.\end{aligned} \tag{39}$$

Analogously, for any other index j, following the same analysis, we have:

$$\begin{aligned}&\Pr\left[0 \leftarrow \mathsf{D}(\cdot) \mid \mathrm{F}_k \cap \mathrm{E}_{k,j} \cap \text{reprogramming}\right]\\ =&\Pr\left[0 \leftarrow \mathsf{D}(\cdot) | \mathrm{F}_k \cap \mathrm{E}_{k,1} \cap \text{reprogramming}\right].\end{aligned} \tag{40}$$

According to the above analysis, we can obtain:

$$\begin{aligned}&\mathsf{Adv}(\mathsf{D})\\ \overset{(a)}{=}&\sum\nolimits_{k=0}^{r}\Pr\left[0 \leftarrow \mathsf{D}(\cdot) \cap \mathrm{E}_k|\text{no reprogramming}\right]\\ &-\sum\nolimits_{k=0}^{r}\Pr\left[0 \leftarrow \mathsf{D}(\cdot) \cap \mathrm{E}_k|\text{reprogramming}\right]\\ \overset{(b)}{=}&\Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_0 \cap \text{reprogramming}\right]\\ &-\sum\nolimits_{k=0}^{r}\sum\nolimits_{j=1}^{\binom{r}{k}}\left(1-\frac{1}{2^m}\right)^k\left(\frac{1}{2^m}\right)^{r-k}\cdot\Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_{k,j} \cap \text{reprogramming}\right]\\ \overset{(c)}{\approx}&\Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{E}_{0,1} \cap \text{reprogramming}\right]\\ &-\sum\nolimits_{k=0}^{r}\binom{r}{k}\left(1-\frac{1}{2^m}\right)^k\left(\frac{1}{2^m}\right)^{r-k}\cdot 1\cdot\Pr\left[0 \leftarrow \mathsf{D}(\cdot)|\mathrm{F}_k \cap \mathrm{E}_{k,1} \cap \text{reprogramming}\right]\\ \overset{(d)}{=}&\frac{1}{2}+\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}\\ &-\frac{1}{2}\cdot\sum\nolimits_{k=0}^{r}\binom{r}{k}\left(1-\frac{1}{2^m}\right)^k\left(\frac{1}{2^m}\right)^{r-k}\\ &-\sum\nolimits_{k=0}^{r}\binom{r}{k}\left(1-\frac{1}{2^m}\right)^k\left(\frac{1}{2^m}\right)^{r-k}\cdot\left[\begin{array}{l}\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}\cdot\frac{r-k+1}{2r}\\ +\frac{q}{2^n}\cdot\frac{k^2+k-2kr}{r}\end{array}\right]\\ =&\frac{1}{2}\cdot\left[1-\sum\nolimits_{k=0}^{r}\binom{r}{k}\left(1-\frac{1}{2^m}\right)^k\left(\frac{1}{2^m}\right)^{r-k}\right]+\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}\\ &+\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}\cdot\sum\nolimits_{k=0}^{r}\binom{r}{k}\left(1-\frac{1}{2^m}\right)^k\left(\frac{1}{2^m}\right)^{r-k}\cdot\left[\frac{r+1}{2r}-\frac{1}{2r}\cdot k\right]\\ &-\frac{q}{2^n}\cdot\sum\nolimits_{k=0}^{r}\binom{r}{k}\left(1-\frac{1}{2^m}\right)^k\left(\frac{1}{2^m}\right)^{r-k}\cdot\left[\frac{1}{r}(k^2+k)-2k\right]\\ \overset{(e)}{=}&\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}-\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}\cdot\left(\frac{r+1}{2r}-\frac{1}{2r}\mathbb{E}(K)\right)\\ &-\frac{q}{2^n}\left[\frac{1}{r}\left(\mathbb{E}(K^2)+\mathbb{E}(K)\right)-2\mathbb{E}(K)\right]\\ \overset{(f)}{=}&\sqrt{\frac{rq}{2^n}}\cdot\sqrt{1-\frac{rq}{2^n}}\cdot(1-\frac{1}{2r}-\frac{1}{2^{m+1}})+\frac{q}{2^n}\cdot(1-\frac{1}{2^m})\cdot(r+\frac{r-1}{2^m})\\ \approx&\Omega(\sqrt{\frac{rq}{2^n}}).\end{aligned} \tag{41}$$

Here, (a) follows from Eq. (29); (b) is obtained by combining Eqs. (30) and (31); (c) utilizes Eqs. (35) and (37); (d) is derived from Eqs. (34) and (39); and (e) is based on interpreting the terms $\sum_{k=0}^{r}\binom{r}{k}(1-\frac{1}{2^m})^k(\frac{1}{2^m})^{r-k}\cdot k$ and $\sum_{k=0}^{r}\binom{r}{k}(1-\frac{1}{2^m})^k(\frac{1}{2^m})^{r-k}\cdot k^2$ as the expectation of K and K^2, respectively, where K is a

random variable representing the number of successes in r independent Bernoulli trials with success probability $1 - \frac{1}{2^m}$. Since K follows a binomial distribution $B(r, 1 - \frac{1}{2^m})$, (f) is derived using the standard properties: $\mathbb{E}(K) = r(1 - \frac{1}{2^m})$ and $\mathrm{Var}(K) = \mathbb{E}(K^2) - (\mathbb{E}(K))^2 = r \cdot (1 - \frac{1}{2^m}) \cdot \frac{1}{2^m}$.

In conclusion, the distinguishing advantage of D is lower-bounded as follows:

$$\mathsf{Adv}(\mathsf{D}) \approx \Omega(\sqrt{\frac{rq}{2^n}}). \tag{42}$$

Till now, we've generalized the original Theorem 7 in the GHHM paper to the multi-reprogramming case. Although our previous analysis assumed r and q to be polynomial in n, the results are applicable to a wider range of parameters:

- First, the above calculation relies on the assumption that the sets $\{S_i\}_{i\in[r]}$ are pairwise disjoint, which is introduced at the beginning of Section C.2. Here we note that, given $r^2q = c \cdot 2^n$ for a constant $c \in (0, \frac{1}{2})$, this assumption holds with probability at least $1 - 2c$. Consequently, the attacker's distinguishing advantage remains the order of $\Omega(\sqrt{\frac{rq}{2^n}})$.
- Second, for r, Eq. (36) holds as long as $r = o(2^{m/2})$.

Generally speaking we've proved the following theorem:

Theorem 11 (The Generalized Theorem 7). *Under the conditions that r is of order $o(2^{m/2})$ and $r^2q = c{\cdot}2^n$ for some constant $c \in (0, \frac{1}{2})$, the attack described in Fig. 9 detects the reprogramming of a random oracle $\mathsf{O} : \{0,1\}^n \to \{0,1\}^m$ at r points with the probability at least $\Omega(\sqrt{\frac{rq}{2^n}})$.*

Remark 3. Note that our Theorem 11 differs slightly from the original Theorem 7 in the GHHM paper, which specifies a concrete range for q. This does not pose a conflict, since the original analysis for the general case in the GHHM paper also assumes $q = o(2^n)$. Through a more refined analysis, we find that the constraint on r should be adjusted to $o(2^{m/2})$, and for q it should follow that $r^2q = c{\cdot}2^n$ for some constant $c \in (0, \frac{1}{2})$. Meanwhile, the above anlalysis acturally shows that, our new reprogramming theorem (Theorem 7 in this paper) is tight.