Quantum security proofs using semi-classical oracles

Andris Ambainis[1], Mike Hamburg[2], and Dominique Unruh[3]

[1] University of Latvia
[2] Rambus Security Division
[3] University of Tartu

Abstract. We present an improved version of the one-way to hiding (O2H) Theorem by Unruh, J ACM 2015. Our new O2H Theorem gives higher flexibility (arbitrary joint distributions of oracles and inputs, multiple reprogrammed points) as well as tighter bounds (removing square-root factors, taking parallelism into account). The improved O2H Theorem makes use of a new variant of quantum oracles, semi-classical oracles, where queries are partially measured. The new O2H Theorem allows us to get better security bounds in several public-key encryption schemes.

Keywords: post-quantum cryptography, quantum random oracle model, one-way to hiding, public-key encryption, provable security

1 Introduction

Ever since it was first introduced in [6] as a proof technique for cryptographic proofs, the random oracle model has been widely used to analyze cryptographic schemes, especially when highly efficient, practical solutions are desired. In the post-quantum setting, however, we need to be careful how the random oracle is modeled. When the adversary makes a query, the input to the random oracle should not be measured [8]. That is, queries should be possible in superposition between different inputs (we then speak of a "quantum random oracle"). Otherwise, the random oracle model would be a very unrealistic idealization of the real world since a quantum adversary can evaluate, say, a hash function in superposition.

Unfortunately, proving the security in the quantum random oracle model is considerably more difficult than in the classical random oracle model. One example of a classical proof technique that is not easy to mimic is programming of the random oracle. In this technique, we run the adversary with access to a random oracle but we change the answer to certain queries during the execution. In a nutshell, as long as we can show that the probability of changing a value that the adversary has already queried is negligible, the adversary will not notice the programming, and the proof goes through. In the quantum setting, this does not make sense. The adversary could query the superposition of all inputs in its first query. Then any programming would change a value that has already been queried.

A technique that can solve this problem (at least in certain situations) is the One-Way to Hiding (O2H) Theorem from [33]. The O2H Theorem solves the reprogramming problem by showing, roughly speaking, that we can bound the probability that the adversary distinguishes between two oracles G and H (the original and the reprogrammed oracle) in terms of the probability that the adversary can guess the location where the oracle is reprogrammed (we speak of the "guessing game"). This conceptually simple theorem has proven powerful in a number of security proofs for post-quantum secure encryption schemes and other constructions (see our overview in Section 1.2). However, the O2H Theorem has a number of limitations that limit its applicability, or give bad bounds in concrete security proofs.

In this work, we present a new version of the O2H Theorem that improves on the state of the art in a number of aspects:

- **Non-uniform random oracles.** The random oracle that is reprogrammed does not have to be a uniformly random function. We allow any distribution of oracles, e.g., invertible permutations, ideal ciphers, etc.
- **Multiple reprogrammed points.** We can reprogram the oracle in more than a single point. That is, we can reprogram the random oracle at a set of positions S and then bound the probability that the adversary detects this reprogramming with a single application of the O2H Theorem.
- **Arbitrary joint distributions.** We allow the distribution of reprogrammed locations and of the adversary's input to be arbitrarily correlated with the distribution of the random oracle. This is especially important if the reprogrammed location depends on the random oracle (e.g., reprogramming $H(x)$ where $x := H(r)$ for random r).
- **Tighter bounds for guessing games.** Our O2H Theorem bounds the difference of the square-roots of the adversary probabilities between two games. In many cases involving guessing games (i.e., where we intend to show that the probability of a certain event is negligible) this leads to bounds that are quadratically better.
- **Tighter bounds using semi-classical oracles.** We introduce a new technique, called semi-classical oracles. By applying the O2H Theorem to games involving semi-classical oracles, we can again get better bounds in some cases. (Whether some advantage is gained depends very much on the specific proof in which the O2H Theorem is used.)
- **Query depth.** Our O2H Theorem distinguishes query number q and query depth d. Thus, for cases in which the adversary has a high parallelism, we get better bounds (and for sequential adversaries nothing is lost by setting $d := q$).

One crucial novelty in our O2H Theorem is the use of "semi-classical oracles". In a nutshell, a semi-classical oracle is an oracle that only measures whether the adversary queried a given "forbidden" input, but does not measure anything beyond that. (In contrast, a quantum oracle does not measure anything, and a classical oracle measures everything.) So, for example, if the adversary queries a superposition of non-measured inputs, nothing is measured.

Our O2H Theorem bounds the distinguishing probability between two oracles G and H again in terms of the success probability in a "guessing game" where the adversary has to query an oracle on one of the forbidden inputs on which G and H differ. But in contrast to the original O2H Theorem, the adversary is given a semi-classical oracle in the guessing game! (In the original O2H Theorem, the adversary is given a quantum oracle.) Using a semi-classical oracle, the guessing game can be expressed more simply since it is well-defined whether the forbidden input has been queried or not. (In the original O2H Theorem, we instead have to stop at a random query and measure whether that particular query queries the forbidden input. This makes the description of the game more complex, and the random selection of a single query is the reason why the original O2H Theorem gives worse bounds.)

We stress that the semi-classical oracles are purely a proof technique and occur in intermediate games in proofs involving the new O2H Theorem. The final security results still hold in the quantum random oracle model, not in some "semi-classical random oracle model".

In this work, we introduce semi-classical oracles, state and prove the new O2H Theorem (together with a query complexity result about searching in semi-classical oracles), and demonstrate its usefulness by elementary examples and by exploring the impact on the security bounds of existing encryption schemes.

Organization. In Section 1.1 we shortly discuss some related work, and in Section 1.2 we discuss the impact of our result on existing cryptographic schemes. Section 2 presents basic notation. Our notion of semi-classical oracles is introduced in Section 3. We also state our main theorems in Section 3, the proofs are deferred to Section 5 (after the examples). We present examples how to use the new technique in Section 4.

1.1 Related work

Variants of the O2H Theorem. Variants of the O2H Theorem were introduced in [33,31,32,22,14], see the beginning of Section 1.2 for more details.

Other proof techniques for the quantum random oracle model. [10] showed that Grover search is optimal with respect to worst-case complexity ([36] when parallelism is considered). [32,21] generalized this to the average-case which implies that finding preimages of the random oracle is hard. [8] introduced "history-free reductions" which basically amounts to replacing the random oracle by a different function right from the start. [38] showed that random oracles can be simulated using $2q$-wise independent functions. Based on this, [32] introduces a technique for extracting preimages of the random oracle. [38] introduces the "semi-constant distributions" technique that allows us to program the random oracle in many random locations with a challenge value without the adversary noticing. [37] improves on this with the "small-range distribution" technique that allows us to simulate random oracles using random looking functions with a small range. [39] shows that random oracles are indistinguishable from random permutations, and

as a consequence that random oracles are collision resistant (this is generalized by [29,15,4] to the case of non-uniformly distributed functions). Collision-resistance of the random oracle is generalized to the "collapsing property" which allows us to show that measuring the output of the random oracle effectively measures the input. More general methods for problems in quantum query complexity (not limited to random oracles) include the polynomial method [5] and the adversary method [1]. [3] shows that the difficulties of using the quantum random oracle are not just a matter of missing proof techniques, but that in certain cases classically secure schemes are not secure in the quantum random oracle model.

Cryptosystems whose security proof is based on O2H Theorems. See Section 1.2.

1.2 Impact on existing cryptosystems

Above, we explained why our new O2H Theorem can lead to better bounds. We will also illustrate that point with a few simple examples in Section 4. However, to better judge the impact on realistic cryptosystems, we need to ask the question how the bounds achieved by existing security proofs improve.

We are aware of the following results in the quantum random oracle model that employ some variant of the original O2H Theorem from [33]: [33] introduced the O2H Theorem to build revocable timed-release encryption schemes, [31] introduced an "adaptive" version of the O2H Theorem[4] to analyze a quantum position verification protocol, [32] made the O2H Theorem even more adaptive and used this for the design of non-interactive zero-knowledge proof systems and signature schemes (and this in turn is the basis for various follow-up schemes such as [35,18,11,13,12,9]). [34] uses the O2H variant from [32] to prove security of Fiat-Shamir [16], both as a proof system and as a signature scheme. [14] uses a variant of the O2H Theorem for proving security of Leighton-Micali signatures [25] (their variant generalizes [33] in some aspects but only works when the position where the oracle is programmed is information-theoretically hidden). [28] uses the O2H Theorem for constructing PRFs and MACs. [30] was the first paper to employ the O2H Theorem for designing public key encryption schemes: it proved the security of variants of the Fujisaki-Okamoto transform [17] and the OAEP transform [7] (introducing one extra hash value in the ciphertext for "key confirmation"). [19] modularized and improved the Fujisaki-Okamoto variant from [30], also using key confirmation. [27] proved security of a construction without key confirmation, still using the O2H Theorem. [22] introduced a variant of the O2H Theorem that allows some of the oracles and inputs given to the adversary to be non-uniformly distributed, subject to the independence and uniformity of certain random variables, and uses it to prove the security of further public-key encryption schemes. (Since our O2H Theorem can also handle non-uniform inputs, it might be that it can serve as a drop-in replacement in the proofs in [22] removing the necessity to check the independence conditions.) [24]

[4] Which allows to reprogram the random oracle at a location that is influenced by the adversary.

proves security of public-key encryption schemes with explicit rejection; an earlier version [23] of [24] used the O2H Theorem from [22], the current version uses our new O2H Theorems. [20] analyzes public-key encryption and authenticated key exchange schemes, using the original O2H Theorem from [33] in the first revision, but improving the bounds using our new O2H Theorem.

Thus, O2H Theorems might be one of the most widely used proof technique for cryptosystems involving quantum random oracles. We expect that our improvement of the O2H Theorem allows us to derive better security bounds for most of the above schemes. To give some evidence to this hypothesis, we report on the advantages gained by using our improvement in three of the works above, namely Targhi-Unruh [30], Hövelmanns-Kiltz-Schäge-Unruh [20], and Jiang-Zhang-Ma [24].

In case of [24], an earlier draft [23] used the O2H variant from [22], while the current version [24] already uses our new O2H Theorem. Since the O2H variant from [22] was introduced to handle the case where not all oracles and adversary inputs are independent, this demonstrates that our O2H Theorem can handle this case, too. (Besides giving tighter bounds.) Similarly, the first eprint version of [20] used the original O2H Theorem from [33], while the second version was updated to use our new O2H Theorem.

The old and new bounds are are summarized in Figure 1. The figure lists the advantages against IND-CCA security for different settings. Since it is difficult to compare the various formulas, in the column "queries", we summarize the relationship between query number and attack probability: Assuming that the terms involving ε, the advantage against the underlying public-key encryption scheme, dominate all other terms, how many queries does one have to make to break the scheme (with constant probability)? E.g., given an advantage $q\sqrt{\varepsilon}$, we need $q \approx \varepsilon^{-1/2}$ queries for a successful attack, so we write $q^2 \approx 1/\varepsilon$ in that case.

Furthermore, in the full version [2], we reprove the security of the Fujisaki-Okamoto variant from [30] using our O2H Theorem. That result is particularly interesting because of its heavy use of the O2H Theorem. This allows us to make use of several of the new features of our O2H Theorem.

- It uses "nested invocations" of the O2H Theorem. That is, first the O2H Theorem is applied as usual to a pair of games, leading to a guessing game in which we need to show that the guessing probability P_{guess} of the adversary is negligible. But then the O2H Theorem is applied again to prove this. Since the bound obtained by the O2H Theorem contains a square root over P_{guess}, the nested application of the O2H Theorem introduces nested square roots, i.e., a forth root. This leads to a particularly bad bound in [30].
  In contrast, our new O2H Theorem allows us to directly bound the difference of the square roots of the success probabilities of the adversary in two games. This means that in a nested invocation, when we analyze P_{guess}, the O2H Theorem directly tells us how $\sqrt{P_{\text{guess}}}$ changes (instead of how P_{guess} changes). This avoids the nested square root.
- It uses the adaptive version of the O2H Theorem (from [31]). While our O2H Theorem is not adaptive (in the sense that the input where the oracle is

Setting	Bound	Queries
Targhi-Unruh [30]		
old O2H, one-way	$\varepsilon_{sym} + q^{9/5}2^{-\gamma/5} + q^{3/2}\varepsilon^{1/4} + q^{3/2}2^{-n_1/4}$	$q^6 \approx 1/\varepsilon$
new O2H, IND-CPA	$\varepsilon_{sym} + q^{9/5}2^{-\gamma/5} + qq_{dec}^{1/2}\varepsilon^{1/2} + q^{3/2}q_{dec}2^{-n/2}$	$q^2q_{dec} \approx 1/\varepsilon$
new O2H, one-way	$\varepsilon_{sym} + q^{9/5}2^{-\gamma/5} + q^{3/2}q_{dec}\varepsilon^{1/2}$	$q^3q_{dec}^2 \approx 1/\varepsilon$
Hövelmanns-Kiltz-Schäge-Unruh [20]		
old O2H, IND-CPA	$q\varepsilon^{1/2} + q2^{-n/2}$	$q^2 \approx 1/\varepsilon$
new O2H, IND-CPA	$q^{1/2}\varepsilon^{1/2} + q2^{-n/2}$	$q \approx 1/\varepsilon$
Jiang-Zhang-Ma [24]		
old O2H, one-way	$q\varepsilon^{1/2}$	$q^2 \approx 1/\varepsilon$
new O2H, one-way	$q\varepsilon^{1/2}$	$q^2 \approx 1/\varepsilon$
new O2H, IND-CPA	$q^{1/2}\varepsilon^{1/2} + q2^{-n/2} + q2^{-n'}$	$q \approx 1/\varepsilon$

The "setting" column says whether the proof uses the old/new O2H and whether it is based on one-wayness or IND-CPA security of the underlying public-key encryption scheme.
The "bound" column gives the bound on the advantage of the adversary against IND-CCA security, up to constant factor. (In the case of [30] a hybrid public-key encryption scheme is constructed, in the other cases a KEM.) ε is the advantage of the reduced adversary against the one-wayness or IND-CPA security of the underlying public-key scheme, respectively. (A complete description would contain the runtime of that adversary. For this overview this is not relevant since in all cases, that runtime did not change when switching to the new O2H Theorem.) ε_{sym} is the advantage against the underlying symmetric encryption scheme. q is the number of queries (random oracle + decryption queries), q_{dec} only the decryption queries. γ is the min-entropy of ciphertexts, n the plaintext length of the underlying public-key scheme, and n' is the length of the additional hash appended to the ciphertext in [24].
The "queries" column summarizes the effect of queries compared to the security of the underlying public-key scheme (see the explanation in the text, higher exponent is worse).
For simplicity, we give the bounds for the case where no decryption errors occur.

Fig. 1. Security bounds of different Fujisaki-Okamoto variants with new and old O2H Theorems.

reprogrammed has to be fixed at the beginning of the game), it turns out that in the present case our new O2H Theorem can replace the adaptive one. This is because our new O2H Theorem allows us to reprogram the oracle at a large number of inputs (not just a single one). It turns out we do not need to adaptively choose the one input to reprogram, we just reprogram all potential inputs. At least in the proof from [30], this works without problems.

We restate (in [2]) the proof from [30] both under the assumption that the underlying public-key encryption scheme is one-way and under the assumption that it is IND-CPA secure. While in the original proof, we get essentially the same bound no matter which of the two assumptions we use, with the new O2H Theorem, the resulting bounds are much better when using IND-CPA security (but there is also an improvement in the one-way case).

The resulting bounds are given in Figure 1 as well. We see that the biggest improvement is in the case of IND-CPA security, where the dependence on the query number changed from the sixth power to cubic.

We also noticed a mistake in the proof,[5] which we fixed in our proof. (We do not know if the fix carries over to the original proof.)

But our analysis also shows some potential for future research on the O2H Theorem. The proof from [30] constructs a plaintext extractor Dec^{**} that is relatively inefficient because it iterates through a large number of possible candidate keys. Thus the number of oracle queries performed by Dec^{**} (namely, $O(qq_{dec})$) by far outweighs the number of oracle queries performed by the adversary (namely, $O(q)$). This large number of queries negatively influences the bounds obtained when applying the new O2H Theorem. However, the $O(qq_{dec})$ queries performed by Dec^{**} are all classical, only $O(q)$ quantum queries are made. Our O2H Theorem treats classical and quantum queries the same. A variant of the O2H Theorem that gives better bounds when only a small fraction of the queries are quantum would lead to improvements in the bounds obtained here. We leave this as a problem for future work.

2 Preliminaries

For basics of quantum computing, we refer to a standard textbook such as [26].

Given a function $f : X \to Y$, we model a quantum-accessible oracle $\mathcal{O}$ for f as a unitary transformation U_f operating on two registers Q, R with spaces $\mathbb{C}^X$ and $\mathbb{C}^Y$, respectively, where $U_f : |q, r\rangle \mapsto |q, r \oplus f(x)\rangle$, where $\oplus$ is some involutive group operation (e.g., XOR if Y is a set of bitstrings).

A quantum oracle algorithm is an algorithm that can perform classical and quantum computations, and that can query classical and/or quantum-accessible oracles. We allow an oracle algorithm A to perform oracle queries in parallel. We say A is a q-query algorithm if it performs at most q oracle queries (counting parallel queries as separate queries), and has query depth d if it invokes the oracle at most d times (counting parallel queries as one query). For example, if A performs 5 parallel queries followed by 7 parallel queries, we have $q = 12$ and $d = 2$.

The distinction between query number and query depth is important because realistic brute-force attacks are highly parallel. It's easy to do 2^{64} hash queries on parallel machines — the Bitcoin network does this several times a minute — but it would take millennia to do them sequentially. Query depth is also important because early quantum computers are likely to lose coherency quickly, limiting them to shallow circuits. Our model does not capture this limitation because it does not differentiate between a deep quantum computation and several shallow

[5] In Game 7 in [30], a secret δ^* is encrypted using a one-time secure encryption scheme, and the final step in the proof concludes that therefore δ^* cannot be guessed. However, Game 7 contains an oracle Dec^{**} that in turn accesses δ^* directly, invalidating that argument.

ones with measurements between. But we hope that future work can account for coherency using a notion of query depth.

We will make use of the well-known fact that any quantum oracle algorithm $A^{\mathcal{O}}(z)$ can be transformed into a *unitary* quantum oracle algorithm with constant factor computational overhead and the same query number and query depth. Such an algorithm has registers Q_A (for its state), and $Q_1, \ldots, Q_n$ and $R_1, \ldots, R_n$ for query inputs and outputs, respectively. It starts with an initial state $|\Psi\rangle$ (that may depend on the input z). Then, A alternatingly applies a fixed unitary U on all registers (independent of z and $\mathcal{O}$), and performs parallel queries. Parallel queries apply the oracle $\mathcal{O}$ to Q_i, R_i for each $i = 1, \ldots, n$. (I.e., if $\mathcal{O}$ is implemented by U_f, we apply $U_f \otimes \cdots \otimes U_f$ between U-applications.) Finally, the classical output of $A^{\mathcal{O}}(z)$ is the result of a projective measurement on the final state of A. This implies that in many situations, we can assume our algorithms to be unitary without loss of generality.

3 Semi-classical oracles

Classical oracles measure both their input and their output, whereas quantum-accessible oracles measure neither. We define semi-classical oracles, which measure their output but not their input. Formally, a semi-classical oracle $\mathcal{O}_f^{SC}$ for a function f with domain X and codomain Y is queried with two registers: an input register Q with space $\mathbb{C}^X$ and an output register R with space $\mathbb{C}^Y$.

When queried with a value $|x\rangle$ in Q, the oracle performs a measurement of $f(x)$. Formally, it performs the measurements corresponding to the projectors $M_y : y \in Y$ where $M_y := \sum_{x \in S : f(x) = y} |x\rangle\langle x|$. The oracle then initializes the R register to $|y\rangle$ for the measured y.

In this paper, the function f is always the indicator function f_S for a set S, where $f_S(x) = 1$ if $x \in S$ and 0 otherwise. For brevity, we overload the notation $\mathcal{O}_S^{SC}$ to be the semiclassical oracle for this index function.

To illustrate this, let us see what happens if the adversary performs the same query with a quantum oracle, a classical oracle, and a semi-classical oracle implementing the indicator function for S, respectively: Say the adversary sends the query $\sum_x 2^{-n/2}|x\rangle|0\rangle$, and say $S = \{x_0\}$. When querying a quantum oracle, the oracle returns the state $\sum_x 2^{-n/2}|x\rangle|f_S(x)\rangle = 2^{-n/2}|x\rangle|1\rangle + \sum_{x \neq x_0} 2^{-n/2}|x\rangle|0\rangle$. When querying a classical oracle, the resulting state will be $|x\rangle|f_S(x)\rangle$ for a uniformly random x. But when querying a semi-classical oracle, with probability $1 - 2^{-n}$, the resulting state is $\sum_{x \neq x_0} \frac{1}{\sqrt{2^n - 1}}|x\rangle|0\rangle$, and with probability 2^{-n}, the resulting state is $|x_0\rangle|1\rangle$. In particular, the superposition between all $|x\rangle$ that are not in S is preserved!

In the execution of a quantum algorithm $A^{\mathcal{O}_S^{SC}}$, let Find be the event that $\mathcal{O}_S^{SC}$ ever returns $|1\rangle$. This is a well-defined classical event because $\mathcal{O}_S^{SC}$ measures its output. This event is called Find because if it occurs, the simulator could immediately stop execution and measure the input register Q to obtain a value $x \in S$. If H is some other quantum-accessible oracle with domain X and codomain Y, we define $H \setminus S$ ("H punctured on S") as an oracle which, on input x, first

queries $\mathcal{O}_S^{SC}(x)$ and then $H(x)$. We call this "puncturing" for the following reason: when Find does not occur, the outcome of $A^{H\setminus S}$ is independent of $H(x)$ for all $x \in S$. Those values are effectively removed from H's domain. The following lemma makes this fact formal.

Lemma 1. *Let $S \subseteq X$ be random. Let $G, H : X \to Y$ be random functions satisfying $\forall x \notin S.\ G(x) = H(x)$. Let z be a random bitstring. (S, G, H, z may have arbitrary joint distribution.)*

Let A be a quantum oracle algorithm (not necessarily unitary).

Let E be an arbitrary (classical) event.

Then $\Pr[E \wedge \neg\mathsf{Find} : x \leftarrow A^{H\setminus S}(z)] = \Pr[E \wedge \neg\mathsf{Find} : x \leftarrow A^{G\setminus S}(z)]$.

Unruh's "one-way to hiding" (O2H) Theorem [33] is a key ingredient in most post-quantum security analyses. This theorem bounds how much a quantum adversary's behavior can change when the random oracle changes on a set S, based on the probability that measuring a random query would give a result in S, which we call the "guessing probability". Semi-classical oracles allow us to split the O2H Theorem into two parts. The first part bounds how much a quantum adversary's behavior changes when a random oracle is punctured on S based on $\Pr[\mathsf{Find}]$:

Theorem 1 (Semi-classical O2H). *Let $S \subseteq X$ be random. Let $G, H : X \to Y$ be random functions satisfying $\forall x \notin S.\ G(x) = H(x)$. Let z be a random bitstring. (S, G, H, z may have arbitrary joint distribution.)*

Let A be an oracle algorithm of query depth d (not necessarily unitary).

Let

$$\begin{aligned} P_{\text{left}} &:= \Pr[b = 1 : b \leftarrow A^H(z)] \\ P_{\text{right}} &:= \Pr[b = 1 : b \leftarrow A^G(z)] \\ P_{\text{find}} &:= \Pr[\mathsf{Find} : A^{G\setminus S}(z)] \overset{\text{Lem. 1}}{=} \Pr[\mathsf{Find} : A^{H\setminus S}(z)] \end{aligned} \tag{1}$$

Then

$$|P_{\text{left}} - P_{\text{right}}| \le 2\sqrt{(d+1)\cdot P_{\text{find}}} \quad \textit{and} \quad \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \le 2\sqrt{(d+1)\cdot P_{\text{find}}}$$

The theorem also holds with bound $\sqrt{(d+1)P_{\text{find}}}$ for the following alternative definitions of P_{right}:

$$P_{\text{right}} := \Pr[b = 1 : b \leftarrow A^{H\setminus S}(z)], \tag{2}$$

$$P_{\text{right}} := \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)], \tag{3}$$

$$P_{\text{right}} := \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)], \tag{4}$$

$$P_{\text{right}} := \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{H\setminus S}(z)], \tag{5}$$

$$P_{\text{right}} := \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. \tag{6}$$

In this theorem, we give A only access to a single oracle (G or H). In many settings, there may be additional oracles that A has access to. It may not be obvious at the first glance, but Theorem 1 applies in that case, too. Since there is no assumption on the runtime of A, or on the size of z, nor on the number of queries made to the additional oracles, additional oracles can simply be encoded as part of z. That is, if we want to consider an adversary $A^{H,F}()$, we can instead write $A^H(F)$ where F is a complete (exponential size) description of F.

The proof of Theorem 1 is given in Section 5.2.

The second part relates $\Pr[\mathsf{Find}]$ to the guessing probability:

Theorem 2 (Search in semi-classical oracle). *Let A be any quantum oracle algorithm making some number of queries at depth at most d to a semi-classical oracle with domain X. Let $S \subseteq X$ and $z \in \{0,1\}^*$. (S, z may have arbitrary joint distribution.)*

Let B be an algorithm that on input z chooses $i \xleftarrow{\$} \{1,\dots,d\}$; runs $A^{\mathcal{O}^{SC}_{\varnothing}}(z)$ until (just before) the i-th query; then measures all query input registers in the computational basis and outputs the set T of measurement outcomes.

Then

$$\Pr[\mathsf{Find} : A^{\mathcal{O}^{SC}_{S}}(z)] \leq 4d \cdot \Pr[S \cap T \neq \varnothing : T \leftarrow B(z)] \tag{7}$$

The proof is given in Section 5.3.

In the simple but common case that the input of A is independent of S, we get the following corollary:

Corollary 1. *Suppose that S and z are independent, and that A is a q-query algorithm. Let $P_{\max} := \max_{x \in X} \Pr[x \in S]$. Then*

$$\Pr[\mathsf{Find} : A^{\mathcal{O}^{SC}_{S}}(z)] \leq 4q \cdot P_{\max}. \tag{8}$$

For example, for uniform $x \in \{1,\dots,N\}$, $A^{\mathcal{O}^{SC}_{\{x\}}}$ finds x with probability $\leq 4q/N$.

Proof. Since the query depth of A does not occur in the lemma, we can assume that A does not perform parallel queries. Then the output T of B in Theorem 2 has $|T| \leq 1$, and $d = q$. Thus $\Pr[S \cap T \neq \varnothing : T \leftarrow B(z)]$ is simply the probability that $B(z)$ outputs an element of S. Hence $\Pr[S \cap T \neq \varnothing : T \leftarrow B(z)] \leq P_{\max}$. Then by Theorem 2, $\Pr[\mathsf{Find} : A^{\mathcal{O}^{SC}_{S}}(z)] \leq 4q \cdot P_{\max}$. □

Note that Corollary 1 is essentially optimal (we cannot improve on the factor 4, see Appendix A). Thus, searching in a semi-classical oracle is still slightly easier than in a classical one.

4 Examples how to use the O2H Theorems

To illustrate the use of the theorems from the previous section, we give two illustrative examples: hardness of searching in a sparse random function, and hardness of inverting a random oracle with leakage (in the sense that an only computationally secret encryption of the preimage is given to the adversary).

4.1 Hardness of searching in a sparse random function

Consider the following setting: $H : X \to \{0,1\}$ is a random function where for each x, $H(x) = 1$ with probability $\leq \lambda$ (not necessarily independently). What is the probability to find x with $H(x) = 1$ in q queries? We will prove an upper bound.

We solve this problem using the semi-classical O2H technique introduced by Theorem 1. Let A be a q-query algorithm with depth d. We want to bound $\Pr[H(x) = 1 : x \leftarrow A^H()]$. We do this by a series of games.

Game 1 *$x \leftarrow A^H()$. Measure x. Then A wins if $H(x) = 1$.*

We would like to apply Theorem 1 to this game. But it doesn't work well to apply it to A^H because H is also used outside of A. Therefore, we use a different but obviously equivalent game:

Game 2 *Define $\hat{A}^H()$ to run $x \leftarrow A^H()$; measure x; and return $b := H(x)$. Game 2 runs $b \leftarrow \hat{A}^H()$. Then A wins if $b = 1$.*

Note that $\hat{A}$ is a $(q+1)$-query algorithm with depth $d+1$.

We can apply the semi-classical O2H Theorem (Theorem 1), variant (4)[6] to this game, where $G := 0$ (the constant zero function) and $S := \{x : H(x) = 1\}$. This gives us:

$$\left|\sqrt{\underbrace{\Pr[b = 1 : \text{Game 2}]}_{P_{\text{left}}}} - \sqrt{\underbrace{\Pr[b = 1 \wedge \neg\mathsf{Find} : \text{Game 3}]}_{P_{\text{right}}}}\right| \leq \sqrt{(d+2)\underbrace{\Pr[\mathsf{Find} : \text{Game 3}]}_{P_{\text{find}}}} \quad (9)$$

with

Game 3 *Run $b \leftarrow \hat{A}^{G\setminus S}()$. Then A wins if $b = 1$ and not* Find.

which is equivalent to

Game 4 *$x \leftarrow A^{G\setminus S}()$; set $b \leftarrow (G \setminus S)(x)$. Then A wins if $b = 1$ and not* Find.

What has happened so far? We have used the O2H Theorem to rewrite a game with access to an oracle H (Game 1) into the same game with a different oracle $G = 0$ (Game 4) ("right game"). The new oracle is considerably simpler: in

[6] Theorem 1 gives us different options how to define the right game. Conceptually simplest is variant (1) (it does not involve a semi-classical oracle in the right game), but it does not apply in all situations. The basic idea behind all variants is the same, namely that the adversary gets access to an oracle G that behaves differently on the set S of marked elements.

In the present proof, we use specifically variant (4) because then Game 4 will be of a form that is particularly easy to analyze (the adversary has winning probability 0 there).

this specific case, it is all zero. The difference between the two games is bounded by (9) in terms of how hard it is to find an element in the set S (the "marked elements"), i.e., a position where G and H differ (the "finding game"). This is the typical way of applying an O2H Theorem: Replace the oracle H by something simpler, continue the game-based proof from the right game, and additionally perform a second game-based proof to bound the probability of finding a marked element in the finding game.

However, there are several crucial differences to the use of prior O2H lemmas (e.g., [33]). First, prior O2H Theorems required G and H to be uniformly random functions, and to differ only at a single location x. But here H is not assumed to be uniform, and it differs from G at more than a single input (i.e. at the entire set S). This allows us to analyze search problems with multiple targets.

Second, (9) has square roots on the left-hand side. This is optional: Theorem 1 also gives a bound without square roots. In our example, since P_{right} is very small, the square-root variant gives smaller bounds for P_{left}.

Third, the finding game is expressed using semi-classical oracles. This is never a limitation because we can always replace the semi-classical oracles by quantum-accessible ones using Theorem 2 (which then gives bounds comparable to the O2H from [33]). However, as we will see in the next section, in some cases semi-classical oracles give better bounds.

In our case, we trivially have $\Pr[G(x) = 1 \wedge \neg\mathsf{Find} : \text{Game } 4] = 0$ since $G = 0$.

However, analyzing $\Pr[\mathsf{Find} : \text{Game } 3]$ is less trivial. At the first glance, it seems that having access to the oracle $G = 0$ yields no information about S, and thus finding an element of S is down to pure luck, and cannot succeed with probability greater than $(q+1)\lambda$. But in fact, computing $G \setminus S$ requires measuring whether each query is in S. The measurement process can leak information about S. Section A shows that at least in some cases, it is possible to find elements of S with greater probability than $(q+1)\lambda$. Fortunately, we have a result for this situation, namely Corollary 1, which shows that $\Pr[\mathsf{Find} : \text{Game } 4] \leq 4(q+1)\lambda$.

Plugging this into (9), we get

$$\Pr[H(x) = 1 : \text{Game } 1] \leq 4(d+2)(q+1)\lambda.$$

Without the square roots on the left-hand side of (9), we would get only the bound $\sqrt{4(d+2)(q+1)\lambda}$.

We summarize what we have proven in the following lemma:

Lemma 2 (Search in unstructured function). *Let H be a random function, drawn from a distribution such that $\Pr[H(x) = 1] \leq \lambda$ for all x. Let A be a q-query adversary with query depth d. Then $\Pr[H(x) = 1 : b \leftarrow A^H()] \leq 4(d+2)(q+1)\lambda$.*

While this is a simple consequence of our O2H technique, we are not aware that this bound was already presented in the literature. While [36] already showed a trade-off between parallelism and query number in unstructured quantum search. However, our result gives an explicit (and tight) success probability and applies even to functions whose outputs are not i.i.d. For the special case of no-parallelism ($d = q$) and i.i.d. functions, the best known bound was [21, Theorem 1] which

we improve upon by a factor of 2. Additionally, our lemma allows the different outputs of H to be correlated while prior results require them to be independent.

4.2 Hardness of inverting a random oracle with leakage

The previous example considered a pure query-complexity problem, searching in a random function. It can easily be solved with other techniques (giving slightly different bounds). Where O2H Theorems shine is the combination of computational hardness and random oracles. The following example illustrates this.

Let E be a randomized algorithm taking input from a space X, such that it is difficult to distinguish the distributions

$$\mathcal{D}_1 := \{(x, E(x)) : x \xleftarrow{\$} X\} \quad \text{and} \quad \mathcal{D}_0 := \{(x_1, E(x_2)) : x_1, x_2 \xleftarrow{\$} X\}$$

For a quantum algorithm B, define its E-distinguishing advantage as

$$\mathrm{Adv}_{\mathrm{IND}-E}(B) := \left| \begin{array}{l} \Pr\left[1 \leftarrow B(x,e) : (x,e) \leftarrow \mathcal{D}_1\right] \\ \quad - \Pr\left[1 \leftarrow B(x,e) : (x,e) \leftarrow \mathcal{D}_0\right] \end{array} \right|$$

For example, E could be IND-CPA-secure encryption. Let $H : X \to Y$ be a random oracle which is independent of E. How hard is it to invert H with a leakage of E? That is, given a quantum oracle algorithm A, we want to bound

$$\mathrm{Adv}_{\text{OW-LEAK-}E}(A) := \Pr\left[A^H(H(x), E(x)) = x : x \xleftarrow{\$} X\right]$$

We can do this using a series of games. For brevity, we will go into slightly less detail than in Section 4.1. Let w_i be the probability that the adversary wins Game i.

Game 0 (Original) $x \xleftarrow{\$} X; x' \leftarrow A^H(H(x), E(x))$. *The adversary wins if* $x' = x$.

Now choose a random $y \xleftarrow{\$} Y$, and set a different random oracle $G := H(x := y)$ which is the same as H on every input except $S := \{x\}$. We can define a new game where the adversary has access to $G \setminus S$:

Game 1 (Punctured, first try) $x \xleftarrow{\$} X; x' \leftarrow A^{G\setminus\{x\}}(H(x), E(x))$. *The adversary wins if* $x' = x$ *and not* Find.

Applying Theorem 1 variant (4),[7] we find that

$$\left| \sqrt{\underbrace{\Pr[x' = x : \text{Game } 0]}_{P_{\text{left}}=w_0}} - \sqrt{\underbrace{\Pr[x' = x \wedge \neg\mathsf{Find} : \text{Game } 1]}_{P_{\text{right}}=w_1}} \right| \leq \sqrt{\underbrace{(d+1)\Pr\left[\mathsf{Find} : \text{Game } 1\right]}_{P_{\text{find}}}}$$

[7] Choosing a different variant here would slightly change the formula below but lead to the same problems.

Unlike in Section 4.1, this time we do not have a trivial bound for w_1. We could bound it in terms of distinguishing advantage against E. But let's instead try to make this game more like the ones in Section 4.1: we can cause the adversary to Find instead of winning. To do this, we just apply an extra hash operation. Let $\hat{A}^H(y, e)$ be the algorithm which runs $x' \leftarrow A^H(y, e)$; computes $H(x')$ and ignores the result; and then returns x'. Then $\hat{A}$ performs $q + 1$ queries at depth $d + 1$. This gives us a new game:

Game 2 (Original with extra hash) $x \xleftarrow{\$} X; x' \leftarrow \hat{A}^H(H(x), E(x))$. *The adversary wins if* $x' = x$.

Clearly $w_2 = w_0$. The new punctured game is also similar:

Game 3 (Punctured, extra hash) $x \xleftarrow{\$} X; x' \leftarrow \hat{A}^{G\setminus\{x\}}(H(x), E(x))$. *The adversary wins if* $x' = x$ *and not* Find.

Applying Theorem 1 variant (4)[8] as before gives

$$|\sqrt{w_3} - \sqrt{w_2}| \leq \sqrt{(d+2)\Pr[\mathsf{Find} : \text{Game } 3]} \tag{10}$$

But the adversary cannot win Game 3: the extra hash query triggers Find if $x' = x$, and the adversary does not win if Find. Therefore $w_3 = 0$. Plugging this into (10) and squaring both sides gives:

$$w_0 = w_2 \leq (d+2)\Pr[\mathsf{Find} : \text{Game } 3] \tag{11}$$

It remains to bound the right-hand side. We first note that in Game 3, the value $H(x)$ is only used once, since the adversary does not have access to $H(x)$: it only has access to G, which is the same as H everywhere except x. So Game 3 is the same as if $H(x)$ is replaced by a random value:

Game 4 (No $H(x)$) *Set* $x \xleftarrow{\$} X; y \xleftarrow{\$} Y; \hat{A}^{G\setminus\{x\}}(y, E(x))$. *We do not care about the output of* $\hat{A}$*, but only whether it* Finds.

Clearly $\Pr[\mathsf{Find} : \text{Game } 4] = \Pr[\mathsf{Find} : \text{Game } 3]$. Finally, we apply the indistinguishability assumption by comparing to the following game:

Game 5 (IND-E challenge) $(x_1, x_2) \xleftarrow{\$} X; y \xleftarrow{\$} Y; \hat{A}^{G\setminus\{x_1\}}(y, E(x_2))$.

Let $B(x, e)$ be an algorithm which chooses $y \xleftarrow{\$} Y$; runs $\hat{A}^{G\setminus\{x\}}(y, e)$; and returns 1 if Find and 0 otherwise. Then B runs in about the same time as A plus $(q + 1)$ comparisons. If (y, e) are drawn from $\mathcal{D}_1$, then this experiment is equivalent to Game 4, and it they are drawn from $\mathcal{D}_0$ then it is equivalent to Game 5. Therefore B is a distinguisher for E with advantage exactly

$$\mathrm{Adv}_{\mathrm{IND}-E}(B) = |\Pr[\mathsf{Find} : \text{Game } 5] - \Pr[\mathsf{Find} : \text{Game } 4]| \tag{12}$$

[8] The reason for choosing this particular variant is that same as in footnote 6.

Furthermore, in Game 5, the oracle G is punctured at x_1, which is uniformly random and independent of everything else in the game. So by Theorem 2,

$$\Pr[\mathsf{Find} : \text{Game } 5] \leq 4(q+1)/\mathrm{card}\,(X)$$

Combining this with (11) and (12), we have

$$\mathrm{Adv}_{\text{OW-LEAK-}E}(A) \leq (d+2)\mathrm{Adv}_{\mathrm{IND}-E}(B) + \frac{4(d+2)(q+1)}{\mathrm{card}\,(X)}$$

This is a much better bound than we would have gotten without using semi-classical oracles (i.e., the O2H Theorem from [33]). In front of $\mathrm{Adv}_{\mathrm{IND}-E}(B)$, we only have the factor $d+2$. In contrast, if we had applied Theorem 2 directly after using Theorem 1, then we would have gotten a factor of $O(qd)$ in front of $\mathrm{Adv}_{\mathrm{IND}-E}(B)$. If we had used the O2H from [33], then we would have gotten an even greater bound of $O(q\sqrt{\mathrm{Adv}_{\mathrm{IND}-E}(B) + 1/\mathrm{card}\,(X)})$. However, this bound with semi-classical oracles assumes indistinguishability, whereas an analysis with the original O2H Theorem would only require E to be one-way.

5 Proofs

5.1 Auxiliary lemmas

The fidelity $F(\sigma,\tau)$ between two density operators is $\mathrm{tr}\sqrt{\sqrt{\sigma}\tau\sqrt{\sigma}}$, the trace distance $\mathrm{TD}(\sigma,\tau)$ is defined as $\frac{1}{2}\mathrm{tr}|\sigma-\tau|$, and the Bures distance $B(\tau,\sigma)$ is $\sqrt{2-2F(\tau,\sigma)}$.

Lemma 3. *For states $|\Psi\rangle, |\Phi\rangle$ with $\||\Psi\rangle\| = \||\Phi\rangle\| = 1$, we have*

$$F(|\Psi\rangle\langle\Psi|, |\Phi\rangle\langle\Phi|) \geq 1 - \frac{1}{2}\||\Psi\rangle - |\Phi\rangle\|^2$$

so that

$$B(|\Psi\rangle\langle\Psi|, |\Phi\rangle\langle\Phi|) \leq \||\Psi\rangle - |\Phi\rangle\|$$

Proof. We have

$$\begin{aligned}\||\Psi\rangle - |\Phi\rangle\|^2 &= (\langle\Psi| - \langle\Phi|)(|\Psi\rangle - |\Phi\rangle) = \||\Psi\rangle\|^2 + \||\Phi\rangle\|^2 - \langle\Psi|\Phi\rangle - \langle\Phi|\Psi\rangle \\ &= 2 - 2\Re(\langle\Psi|\Phi\rangle) \geq 2 - 2|\langle\Psi|\Phi\rangle| \overset{(*)}{=} 2 - 2F(|\Psi\rangle\langle\Psi|, |\Phi\rangle\langle\Phi|)\end{aligned}$$

where $\Re$ denotes the real part, and $(*)$ is by definition of the fidelity F (for pure states). Thus $F(|\Psi\rangle\langle\Psi|, |\Phi\rangle\langle\Phi|) \geq 1 - \frac{1}{2}\||\Psi\rangle - |\Phi\rangle\|^2$ as claimed. The second inequality follows from the definition of Bures distance. □

Lemma 4 (Distance measures vs. measurement probabilities). *Let ρ_1, ρ_2 be density operators (with $\mathrm{tr}\,\rho_i = 1$). Let M be a binary measurement (e.g., represented as a POVM). Let P_i be the probability that M returns 1 when measuring ρ_i.*

Then

$$\sqrt{P_1P_2} + \sqrt{(1-P_1)(1-P_2)} \geq F(\rho_1, \rho_2) \tag{13}$$

Also,

$$\left|\sqrt{P_1} - \sqrt{P_2}\right| \leq B(\rho_1, \rho_2). \tag{14}$$

Furthermore,

$$|P_1 - P_2| \leq \mathrm{TD}(\rho_1, \rho_2) \leq B(\rho_1, \rho_2). \tag{15}$$

Proof. In this proof, given a probability P, let $\bar{P} := 1 - P$. Let $\mathcal{E}$ be the superoperator that maps ρ to the classical bit that contains the result of measuring ρ using M. That is, for every density operator ρ with $\operatorname{tr}\rho = 1$, $\mathcal{E}(\rho) = \left(\begin{smallmatrix} p & 0 \\ 0 & \bar{p} \end{smallmatrix}\right)$ where p is the probability that M returns 1 when measuring ρ.

Then $\rho_i' := \mathcal{E}(\rho_i) = \left(\begin{smallmatrix} P_i & 0 \\ 0 & \bar{P}_i \end{smallmatrix}\right)$ for $i = 1, 2$. We then have

$$\begin{aligned} F(\rho_1, \rho_2) &\overset{(*)}{\leq} F(\rho_1', \rho_2') \overset{(**)}{=} \left\| \sqrt{\rho_1'}\sqrt{\rho_2'} \right\|_{\mathrm{tr}} \\ &= \operatorname{tr} \begin{pmatrix} \sqrt{P_1P_2} & 0 \\ 0 & \sqrt{\bar{P}_1\bar{P}_2} \end{pmatrix} = \sqrt{P_1P_2} + \sqrt{\bar{P}_1\bar{P}_2} \end{aligned}$$

where $(*)$ is due to the the monotonicity of the fidelity [26, Thm. 9.6], and $(**)$ is the definition of fidelity. This shows (13). To prove (14), we compute:

$$\begin{aligned} \left(\sqrt{P_1} - \sqrt{P_2}\right)^2 &= P_1 + P_2 - 2\sqrt{P_1P_2} \\ &\leq P_1 + P_2 - 2\sqrt{P_1P_2} + \left(\sqrt{\bar{P}_1} - \sqrt{\bar{P}_2}\right)^2 \\ &= 2 - 2\sqrt{P_1P_2} - 2\sqrt{\bar{P}_1\bar{P}_2} \overset{(13)}{\leq} 2 - 2F(\rho_1, \rho_2) \overset{(*)}{=} B(\rho_1, \rho_2)^2 \end{aligned}$$

where $(*)$ is by definition of the Bures distance. This implies (14).

The first inequality in (15) is well-known (e.g., [26, Thm. 9.1]). For the second part, we calculate

$$\begin{aligned} \mathrm{TD}(\rho, \tau) &\overset{(*)}{\leq} \sqrt{1 - F(\rho, \tau)^2} = \sqrt{\frac{1 + F(\rho, \tau)}{2}} \cdot \sqrt{2 - 2F(\rho, \tau)} \\ &= \sqrt{\frac{1 + F(\rho, \tau)}{2}} \cdot B(\rho, \tau) \overset{(**)}{\leq} B(\rho, \tau) \end{aligned}$$

Here the inequality marked $(*)$ is shown in [26, (9.101)], and $(**)$ is because $0 \leq F(\rho, \tau) \leq 1$. □

5.2 Proof of Theorem 1

In the following, let $H : X \to Y$, $S \subseteq X$, $z \in \{0,1\}^*$.

Lemma 5 (O2H in terms of pure states). *Fix H, S, z. Let $A^H(z)$ be a unitary quantum oracle algorithm of query depth d. Let Q_A denote the register containing all of A's state.*

Let L be a quantum register with space $\mathbb{C}^{2^d}$ (for the "query log").

Let $B^{H,S}(z)$ be the unitary algorithm on registers Q_A, L that operates like $A^H(z)$, except:

- *It initializes the register L with $|0\dots0\rangle$.*
- *When A performs its i-th set of parallel oracle queries on input/output registers $(Q_1, R_1), \dots, (Q_n, R_n)$ that are part of Q_A, B instead first applies U_S on $(Q_1, \dots, Q_n, L)$ and then performs the oracle queries. Here U_S is defined by:*

$$U_S|x_1,\dots,x_n\rangle|l\rangle := \begin{cases} |x_1,\dots,x_n\rangle|l\rangle & (\text{every } x_j \notin S), \\ |x_1,\dots,x_n\rangle|\mathrm{flip}_i(l)\rangle & (\text{any } x_j \in S) \end{cases}$$

Let $|\Psi_{\mathrm{left}}\rangle$ denote the final state of $A^H(z)$, and $|\Psi_{\mathrm{right}}\rangle$ the final state of $B^{H,S}(z)$.

Let $\tilde{P}_{\mathrm{find}}$ be the probability that a measurement of L in the state $|\Psi_{\mathrm{right}}\rangle$ returns $\neq 0$. (Formally, $\left\|(I \otimes (I - |0\rangle\langle 0|))|\Psi_{\mathrm{right}}\rangle\right\|^2$.)

Then

$$\left\||\Psi_{\mathrm{left}}\rangle \otimes |0\rangle - |\Psi_{\mathrm{right}}\rangle\right\|^2 \leq (d+1)\tilde{P}_{\mathrm{find}}.$$

Proof. We first define a variant B_{count} of the algorithm B that, instead of keeping a log of the successful oracle queries (as B does in L), just counts the number of successful oracle queries (in a register C). Specifically:

Let C be a quantum register with space $\mathbb{C}^{\{0,\dots,d\}}$, i.e., C can store states $|0\rangle, \dots, |d\rangle$. Let $B^{H,S}_{\mathrm{count}}(z)$ be the unitary algorithm on registers Q_A, S that operates like $A^H(z)$, except:

- It initializes the register C with $|0\rangle$.
- When A performs its i-th set of parallel oracle queries on input/output registers $((Q_1, R_1), \dots)$ that are part of Q_A, B instead first applies U'_S on $(Q_1, \dots, Q_n), C$ and then performs the oracle queries. Here U'_S is defined by:

$$U'_S|x_1,\dots,x_n\rangle|c\rangle := \begin{cases} |x_1,\dots,x_n\rangle|c\rangle & (\text{every } x_j \notin S), \\ |x_1,\dots,x_n\rangle|c+1 \bmod d+1\rangle & (\text{any } x_j \in S) \end{cases}$$

Note that the $\bmod d+1$ part of the definition of U'_S has no effect on the behavior of $\tilde{B}$ because U_S is applies only d times. However, the $\bmod d+1$ is required so that U_S is unitary.

Consider the state $|\Psi_{\mathrm{count}}\rangle$ at the end of the execution $B^{H,S}_{\mathrm{count}}(z)$. This may be written

$$|\Psi_{\mathrm{count}}\rangle = \sum_{i=0}^{d} |\Psi'_i\rangle|i\rangle_C. \tag{16}$$

for some (non-normalized) states $|\Psi'_i\rangle$ on Q_A.

Consider the linear (but not unitary) map $N' : |x\rangle|y\rangle \mapsto |x\rangle|0\rangle$. Obviously, N' commutes with the oracle queries and with the unitary applied by A between queries (since those unitaries do not operate on C.) Furthermore $N'U'_S = N'$, and the initial state of B_{count} is invariant under N'. Thus $N'|\Psi_{\text{count}}\rangle$ is the same as the state we get if we execute B_{count} without the applications of U'_S. But that state is $|\Psi_{\text{left}}\rangle|0\rangle_C$ because the only difference between B_{count} and A is that B_{count} initializes C with $|0\rangle$ and applies U'_S to it.

So we have

$$\sum_{i=0}^{d} |\Psi'_i\rangle|0\rangle_C = N'|\Psi_{\text{count}}\rangle = |\Psi_{\text{left}}\rangle|0\rangle_C$$

and hence

$$|\Psi_{\text{left}}\rangle = \sum_{i=0}^{d} |\Psi'_i\rangle. \tag{17}$$

The state $|\Psi_{\text{right}}\rangle$ is a state on Q_A, L and thus can be written as

$$|\Psi_{\text{right}}\rangle = \sum_{l\in\{0,1\}^q} |\Psi_l\rangle|l\rangle_L \tag{18}$$

for some (non-normalized) states $|\Psi_l\rangle$ on Q_A.

Furthermore, both $|\Psi_{\text{count}}\rangle$ and $|\Psi_{\text{right}}\rangle$, when projected onto $|0\rangle$ in register C/L, respectively, result in the same state, namely the state corresponding to no query to $\mathcal{O}_S^{SC}$ succeeding. By (16) and (18), the result of that projection is $|\Psi_0\rangle|0\rangle_L$ and $|\Psi'_0\rangle|0\rangle_C$, respectively. Hence

$$|\Psi_0\rangle = |\Psi'_0\rangle. \tag{19}$$

Furthermore, the probability that no query succeeds is the square of the norm of that state. Hence

$$\big\||\Psi_0\rangle\big\|^2 = 1 - \tilde{P}_{\text{find}}. \tag{20}$$

We have

$$\sum_{i=0}^{d} \big\||\Psi'_i\rangle\big\|^2 = \sum_{i=0}^{d} \big\||\Psi'_i\rangle|i\rangle_C\big\|^2 = \Big\|\sum_{i=0}^{d} |\Psi'_i\rangle|i\rangle_C\Big\|^2 \overset{(16)}{=} \big\||\Psi_{\text{count}}\rangle\big\|^2 = 1.$$

$$\sum_{l\in\{0,1\}^d} \big\||\Psi_l\rangle\big\|^2 = \sum_{l\in\{0,1\}^d} \Big\||\Psi_l\rangle|l\rangle_L\Big\|^2 = \Big\|\sum_{l\in\{0,1\}^d} |\Psi_l\rangle|l\rangle_L\Big\|^2 \overset{(18)}{=} \big\||\Psi_{\text{right}}\rangle\big\|^2 = 1.$$

Thus

$$\sum_{i=1}^{d} \big\||\Psi'_i\rangle\big\|^2 = 1 - \big\||\Psi'_0\rangle\big\|^2 \overset{(20)}{=} \tilde{P}_{\text{find}}, \qquad \sum_{\substack{l\in\{0,1\}^d \\ l\neq 0}} \big\||\Psi_l\rangle\big\|^2 = 1 - \big\||\Psi_0\rangle\big\|^2 \overset{(20)}{=} \tilde{P}_{\text{find}}. \tag{21}$$

Therefore

$$\begin{aligned}
&\Big\||\Psi_{\text{right}}\rangle - |\Psi_{\text{left}}\rangle|0\rangle_L\Big\|^2 \overset{(18)}{=} \Big\|\big(|\Psi_0\rangle - |\Psi_{\text{left}}\rangle\big)|0\rangle + \sum_{\substack{l\in\{0,1\}^d\\ l\neq 0}} |\Psi_l\rangle|l\rangle\Big\|^2 \\
&= \Big\||\Psi_0\rangle - |\Psi_{\text{left}}\rangle\Big\|^2 + \sum_{\substack{l\in\{0,1\}^d\\ l\neq 0}} \big\||\Psi_l\rangle\big\|^2 \overset{(21)}{=} \Big\||\Psi_0\rangle - |\Psi_{\text{left}}\rangle\Big\|^2 + \tilde{P}_{\text{find}} \\
&\overset{(19),(17)}{=} \Big\|\sum_{i=1}^d |\Psi_i'\rangle\Big\|^2 + \tilde{P}_{\text{find}} \overset{(*)}{\leq} \Big(\sum_{i=1}^d \Big\||\Psi_i'\rangle\Big\|\Big)^2 + \tilde{P}_{\text{find}} \overset{(**)}{\leq} d\cdot\sum_{i=1}^d \Big\||\Psi_i'\rangle\Big\|^2 + \tilde{P}_{\text{find}} \\
&\overset{(21)}{=} d\tilde{P}_{\text{find}} + \tilde{P}_{\text{find}} = (d+1)\tilde{P}_{\text{find}}.
\end{aligned}$$

Here $(*)$ uses the triangle inequality, and $(**)$ the AM-QM (or Jensen's) inequality. This is the inequality claimed in the lemma. □

Theorem 1 follows mechanically from Lemma 5 by applying Lemma 4 and Lemma 3 to each case.

Lemma 6 (O2H in terms of mixed states). *Let X, Y be sets, and let $H : X \to Y, S \subset X, z \in \{0,1\}^*$ be random. (With some joint distribution.)*

Let A be an algorithm which queries H at depth d. Let P_{find} be as in Theorem 1.

Let ρ_{left} denote the final state of $A^H(z)$.

Let ρ_{right} denote the final state of $A^{H\setminus S}$. This is the state of the registers Q_A and L, where Q_A is the state of A itself, and L is a register that contains the log of the responses of $\mathcal{O}_S^{SC}$. If the i-th query to $\mathcal{O}_S^{SC}$ returns ℓ_i, then L contains $|\ell_1 \dots \ell_q\rangle$ at the end of the execution of B.

Then $F(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}}) \geq 1 - \frac{1}{2}(d+1)P_{\text{find}}$ and $B(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}}) \leq \sqrt{(d+1)P_{\text{find}}}$.

Proof. Without loss of generality, we can assume that A is unitary: If A is not unitary, we can construct a unitary variant of A that uses an extra auxiliary register Z, and later trace out that register again from the states ρ_{left} and ρ_{right}.

Let $\big|\Psi_{\text{left}}^{HSz}\big\rangle$ be the state $\big|\Psi_{\text{left}}\big\rangle$ from Lemma 5 for specific values of H, S, z. And analogously for $\big|\Psi_{\text{right}}^{HSz}\big\rangle$ and $\tilde{P}_{\text{find}}^{HSz}$.

Then $\rho_{\text{left}} = \text{Exp}_{HSz}[\big|\Psi_{\text{left}}^{HSz}\big\rangle\big\langle\Psi_{\text{left}}^{HSz}\big|]$

Furthermore, if we define $\rho'_{\text{right}} := \text{Exp}_{HSz}[|\Psi_{\text{right}}^{HSz}\rangle\langle\Psi_{\text{right}}^{HSz}|]$, then $\rho_{\text{right}} = \mathcal{E}_L(\rho'_{\text{right}})$ where $\mathcal{E}_L$ is the quantum operation that performs a measurement in the computational basis on the register L.

And $P_{\text{find}} = \text{Exp}_{HSz}[\tilde{P}_{\text{find}}^{HSz}]$.

Then

$$\begin{aligned}
&F(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}}) \\
&\qquad = F\big(\mathcal{E}_L(\rho_{\text{left}} \otimes |0\rangle\langle 0|), \mathcal{E}_L(\rho'_{\text{right}})\big) \\
&\qquad \overset{(*)}{\geq} F\big(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho'_{\text{right}}\big) \\
&\qquad = F\left(\operatorname*{Exp}_{HSz}\Big[\big|\Psi^{HSz}_{\text{left}}\big\rangle\big\langle\Psi^{HSz}_{\text{left}}\big| \otimes |0\rangle\langle 0|\Big], \operatorname*{Exp}_{HSz}\Big[\big|\Psi^{HSz}_{\text{right}}\big\rangle\big\langle\Psi^{HSz}_{\text{right}}\big|\Big]\right) \\
&\qquad \overset{(**)}{\geq} \operatorname*{Exp}_{HSz}\Big[F\left(\big|\Psi^{HSz}_{\text{left}}\big\rangle\big\langle\Psi^{HSz}_{\text{left}}\big| \otimes |0\rangle\langle 0|, \big|\Psi^{HSz}_{\text{right}}\big\rangle\big\langle\Psi^{HSz}_{\text{right}}\big|\right)\Big] \\
&\qquad \overset{\text{Lem. 3}}{\geq} 1 - \tfrac{1}{2}\operatorname*{Exp}_{HSz}\Big[\big\||\Psi^{HSz}_{\text{left}}\rangle \otimes |0\rangle - |\Psi^{HSz}_{\text{right}}\rangle\big\|^2\Big] \\
&\qquad \overset{\text{Lem. 5}}{\geq} 1 - \frac{1}{2}\operatorname*{Exp}_{HSz}\Big[(d+1)\tilde{P}^{HSz}_{\text{find}}\Big] = 1 - \tfrac{1}{2}(d+1)P_{\text{find}}.
\end{aligned}$$

Here $(*)$ follows from the monotonicity of the fidelity [26, Thm. 9.6], and $(**)$ follows from the joint concavity of the fidelity [26, (9.95)]. This shows the first bound from the lemma.

The Bures distance B is defined as $B(\rho,\tau)^2 = 2(1 - F(\rho,\tau))$. Thus

$$\begin{aligned}
B(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}})^2 &= 2(1 - F(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}})) \\
&\leq 2(1 - (1 - \tfrac{1}{2}(d+1)P_{\text{find}})) = (d+1)P_{\text{find}},
\end{aligned}$$

hence $B(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}}) \leq \sqrt{(d+1)P_{\text{find}}}$. □

Theorem 1 (Semi-classical O2H – restated). *Let $S \subseteq X$ be random. Let $G, H : X \to Y$ be random functions satisfying $\forall x \notin S.\ G(x) = H(x)$. Let z be a random bitstring. (S, G, H, z may have arbitrary joint distribution.)*

Let A be an oracle algorithm of query depth d (not necessarily unitary).

Let

$$\begin{aligned}
P_{\text{left}} &:= \Pr[b = 1 : b \leftarrow A^H(z)] \\
P_{\text{right}} &:= \Pr[b = 1 : b \leftarrow A^G(z)] \\
P_{\text{find}} &:= \Pr[\mathsf{Find} : A^{G\setminus S}(z)] \overset{\text{Lem. 1}}{=} \Pr[\mathsf{Find} : A^{H\setminus S}(z)]
\end{aligned} \tag{1}$$

Then

$$|P_{\text{left}} - P_{\text{right}}| \leq 2\sqrt{(d+1)\cdot P_{\text{find}}} \quad \textit{and} \quad \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \leq 2\sqrt{(d+1)\cdot P_{\text{find}}}$$

The theorem also holds with bound $\sqrt{(d+1)P_{\text{find}}}$ for the following alternative definitions of P_{right}:

$$P_{\text{right}} := \Pr[b = 1 : b \leftarrow A^{H\setminus S}(z)], \tag{2}$$

$$P_{\text{right}} := \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)], \tag{3}$$

$$P_{\text{right}} := \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)], \tag{4}$$

$$P_{\text{right}} := \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{H\setminus S}(z)], \tag{5}$$

$$P_{\text{right}} := \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. \tag{6}$$

Proof. We first prove the theorem using the definition of P_{right} from (2).

Let M be the measurement that measures, given the the register Q_A, L, what the output b of A is. Here Q_A is the state space of A, and L is the additional register introduced in Lemma 6. (Since A obtains b by measuring Q_A, such a measurement M exists.)

Let $P_M(\rho)$ denote the probability that M returns 1 when measuring a state ρ.

Then $P_{\text{left}} = P_M(\rho_{\text{left}} \otimes |0\rangle\langle 0|)$ and $P_{\text{right}} = P_M(\rho_{\text{right}})$ where ρ_{left} and ρ_{right} are defined in Lemma 6.

Then

$$\begin{aligned}
\left|P_{\text{left}} - P_{\text{right}}\right| &= \left|P_M(\rho_{\text{left}} \otimes |0\rangle\langle 0|) - P_M(P_{\text{right}})\right| \\
&\overset{\text{Lem. 4}}{\leq} B(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}}) \\
&\overset{\text{Lem. 6}}{\leq} \sqrt{(d+1)P_{\text{find}}} \\
\left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| &= \left|\sqrt{P_M(\rho_{\text{left}} \otimes |0\rangle\langle 0|)} - \sqrt{P_M(P_{\text{right}})}\right| \\
&\overset{\text{Lem. 4}}{\leq} B(\rho_{\text{left}} \otimes |0\rangle\langle 0|, \rho_{\text{right}}) \\
&\overset{\text{Lem. 6}}{\leq} \sqrt{(d+1)P_{\text{find}}}.
\end{aligned}$$

This shows the theorem with the definition of P_{right} from (2).

Now we show the theorem using the definition of P_{right} from (3). Let M instead be the measurement that measures whether $b = 1$ and L contains $|0\rangle$ (this means Find did not happen). Then $P_{\text{left}} = P_M(\rho_{\text{left}} \otimes |0\rangle\langle 0|)$ and $P_{\text{right}} = P_M(\rho_{\text{right}})$, and the rest of the proof is as in the case of (2).

Now we show the theorem using the definition of P_{right} from (5). Let M instead be the measurement that measures whether $b = 1$ or L contains $|x\rangle$ for $x \neq 0$ (this means Find did happen). Then $P_{\text{left}} = P_M(\rho_{\text{left}} \otimes |0\rangle\langle 0|)$ and $P_{\text{right}} = P_M(\rho_{\text{right}})$, and the rest of the proof is as in the case of (2).

Now we show the theorem using the definition of P_{right} from (4). This follows immediately by case (3), and the fact that $\Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)] = \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]$ by Lemma 1.

Now we show the theorem using the definition of P_{right} from (6). By Lemma 1,

$$\Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)] = \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)] \tag{22}$$

$$\Pr[\text{true} \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)] = \Pr[\text{true} \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. \tag{23}$$

From (23), we get (by considering the complementary event):

$$\Pr[\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)] = \Pr[\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. \tag{24}$$

Adding (22) and (24), we get

$$\Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{H\setminus S}(z)] = \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. \tag{25}$$

Then case (6) follows from case (5) and the fact (25).

Now we show the theorem using the definition of P_{right} from (1). Let

$$\begin{aligned}
P_{\text{mid}} &:= \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)],\\
P'_{\text{mid}} &:= \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)],\\
P'_{\text{find}} &:= \Pr[\mathsf{Find} : A^{G\setminus S}(z)].
\end{aligned}$$

By the current lemma, case (3) (which we already proved), we have

$$|P_{\text{left}} - P_{\text{mid}}| \le \sqrt{(d+1)P_{\text{find}}}, \qquad |P_{\text{left}} - P_{\text{mid}}| \le \sqrt{(d+1)P_{\text{find}}},$$

and by case (4), we also get

$$|P_{\text{right}} - P'_{\text{mid}}| \le \sqrt{(d+1)P'_{\text{find}}}, \qquad |P_{\text{right}} - P'_{\text{mid}}| \le \sqrt{(d+1)P'_{\text{find}}},$$

Note that in the second case, we invoke the current lemma with G and H exchanged, and our P_{right} is their P_{left}.

By Lemma 1, $P_{\text{mid}} = P'_{\text{mid}}$ and by (24), $P_{\text{find}} = P'_{\text{find}}$. With this and the triangle inequality, we get

$$|P_{\text{left}} - P_{\text{right}}| \le 2\sqrt{(d+1)P_{\text{find}}}, \qquad |P_{\text{left}} - P_{\text{right}}| \le 2\sqrt{(d+1)P_{\text{find}}}.$$

as required. □

5.3 Proof of Theorem 2

In the following, let $S \subseteq X$, $z \in \{0,1\}^*$.

Lemma 7. *Fix S, z (S, z are not randomized in this lemma.) Let $A^H(z)$ be a unitary oracle algorithm with query depth d.*

Let B be an oracle algorithm that on input z does the following: pick $i \xleftarrow{\$} \{1,\dots,d\}$, runs $A^{\mathcal{O}_\varnothing^{SC}}(z)$ until (just before) the i-th query, measure all query input registers in the computational basis, output the set T of measurement outcomes.

Then

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_S^{SC}}(z)] \le 4d \cdot \Pr[S \cap T \neq \varnothing : T \leftarrow B(z)].$$

Proof. Let $|\Psi_i\rangle$ be the (non-normalized) state of $A^{\mathcal{O}_S^{SC}}(z)$ right after the i-th query in the case that the first i queries return 0. That is, $\||\Psi_i\rangle\|^2$ is the probability that the first i queries return 0, and $|\Psi_i\rangle/\||\Psi_i\rangle\|$ is the state conditioned on that outcome. Let $|\Psi'_i\rangle$ be the corresponding state of $A^{\mathcal{O}_\varnothing^{SC}}(z)$, that is, $|\Psi'_i\rangle$ is the state just after the ith query (or before, since queries to $\mathcal{O}_\varnothing^{SC}$ do not affect the state). Note that $|\Psi_0\rangle = |\Psi'_0\rangle$ is the initial state of $A(z)$ (independent of the oracle).

From the state $|\Psi_i\rangle$, the algorithm A first applies a fixed unitary U that depends only on A. Then it queries the semi-classical oracle $\mathcal{O}_S^{SC}$.

Let P_S be the orthogonal projector projecting the query input registers $Q_1, \ldots, Q_n$ onto states $|T\rangle$ with $S \cap T \neq \varnothing$, formally $P_S := \sum_{T \text{ s.t. } S \cap T \neq \varnothing} |T\rangle\langle T|$. Thus $\|P_S U|\Psi_i\rangle\|^2$ is the probability of measuring T with $S \cap T \neq \varnothing$ in registers $Q_1, \ldots, Q_n$ given the state $U|\Psi_i\rangle$.

Then the i-th query to $\mathcal{O}_S^{SC}$ applies $I - P_S$ to $|\Psi_i\rangle$. Therefore $|\Psi_{i+1}\rangle = (I - P_S)U|\Psi_i\rangle$.

Let $p_i = 1 - \||\Psi_i\rangle\|^2$ be the probability that one of the first i queries returns 1, and let

$$\begin{aligned} r_i &:= p_i + 2\||\Psi_i\rangle - |\Psi_i'\rangle\|^2 = 1 - \||\Psi_i\rangle\|^2 + 2\||\Psi_i\rangle\|^2 - 4\Re\langle\Psi_i'|\Psi_i\rangle + 2\underbrace{\||\Psi_i'\rangle\|^2}_{=1} \\ &= 3 - 4\Re\langle\Psi_i'|\Psi_i\rangle + \||\Psi_i\rangle\|^2. \end{aligned} \tag{26}$$

Notice that $r_0 = 0$ since $|\Psi_0\rangle = |\Psi_0'\rangle$ and $\||\Psi_0\rangle\| = 1$. During the $(i+1)$-st query, $U|\Psi_i\rangle$ is changed to $U|\Psi_i\rangle - P_S U|\Psi_i\rangle$, and $U|\Psi_i'\rangle$ stays the same, so that

$$\begin{aligned} |\Psi_{i+1}\rangle &= U|\Psi_i\rangle - P_S U|\Psi_i\rangle \\ |\Psi_{i+1}'\rangle &= U|\Psi_i'\rangle \end{aligned}$$

Therefore,

$$\begin{aligned} \||\Psi_{i+1}\rangle\|^2 &= \|U|\Psi_i\rangle\|^2 - \big\langle\Psi_i\big|U^\dagger P_S U\big|\Psi_i\big\rangle - \big\langle\Psi_i\big|U^\dagger P_S^\dagger U\big|\Psi_i\big\rangle + \big\langle\Psi_i\big|U^\dagger P_S^\dagger P_S U\big|\Psi_i\big\rangle \\ &= \||\Psi_i\rangle\|^2 - \big\langle\Psi_i\big|U^\dagger P_S U\big|\Psi_i\big\rangle \end{aligned} \tag{27}$$

because P_S is a projector and thus $P_S^\dagger P_S = P_S^\dagger = P_S$. Likewise,

$$\begin{aligned} \langle\Psi_{i+1}'|\Psi_{i+1}\rangle &= \big\langle\Psi_i'\big|U^\dagger U\big|\Psi_i\big\rangle - \big\langle\Psi_i'\big|U^\dagger P_S U\big|\Psi_i\big\rangle \\ &= \langle\Psi_i'|\Psi_i\rangle - \big\langle\Psi_i'\big|U^\dagger P_S U\big|\Psi_i\big\rangle \end{aligned} \tag{28}$$

Let

$$g_i := \langle\Psi_{i-1}'|U^\dagger P_S U|\Psi_{i-1}'\rangle = \big\|P_S U|\Psi_{i-1}'\rangle\big\|^2.$$

Then g_i is the probability that the algorithm B returns T with $S \cap T \neq \varnothing$ when measured at the i-th query.

We calculate

$$\begin{aligned} r_{i+1} - r_i &\overset{(26)}{=} -4\Re\langle\Psi_{i+1}'|\Psi_{i+1}\rangle + \||\Psi_{i+1}\rangle\|^2 + 4\Re\langle\Psi_i'|\Psi_i\rangle - \||\Psi_i\rangle\|^2 \\ &\overset{(27),(28)}{=} 4\Re\langle\Psi_i'|U^\dagger P_S U|\Psi_i\rangle - \langle\Psi_i|U^\dagger P_S U|\Psi_i\rangle \\ &= 4\langle\Psi_i'|U^\dagger P_S U|\Psi_i'\rangle - \underbrace{\langle 2\Psi_i' - \Psi|U^\dagger P_S U|2\Psi_i' - \Psi_i\rangle}_{\geq 0} \\ &\leq 4\langle\Psi_i'|U^\dagger P_S U|\Psi_i'\rangle = 4g_{i+1} \end{aligned}$$

Since $r_0 = 0$, by induction we have

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_S^{SC}}(z)] = p_d \leq r_d \leq 4\sum_{i=1}^{d} g_i = 4d \cdot \Pr\left[S \cap T \neq \varnothing : T \leftarrow B(z)\right]$$

as claimed. □

Theorem 2 (Search in semi-classical oracle – restated). *Let A be any quantum oracle algorithm making some number of queries at depth at most d to a semi-classical oracle with domain X. Let $S \subseteq X$ and $z \in \{0,1\}^*$. (S, z may have arbitrary joint distribution.)*

Let B be an algorithm that on input z chooses $i \xleftarrow{\$} \{1,\dots,d\}$; runs $A^{\mathcal{O}_\varnothing^{SC}}(z)$ until (just before) the i-th query; then measures all query input registers in the computational basis and outputs the set T of measurement outcomes.

Then

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_S^{SC}}(z)] \leq 4d \cdot \Pr[S \cap T \neq \varnothing : T \leftarrow B(z)] \tag{7}$$

Proof. Immediate from Lemma 7 by using the fact that A can always be transformed into a unitary oracle algorithm, and by averaging. □

Acknowledgements. Thanks to Daniel Kane, Eike Kiltz, and Kathrin Hövelmanns for valuable discussions. Ambainis was supported by the ERDF project 1.1.1.5/18/A/020. Unruh was supported by institutional research funding IUT2-1 of the Estonian Ministry of Education and Research, the United States Air Force Office of Scientific Research (AFOSR) via AOARD Grant "Verification of Quantum Cryptography" (FA2386-17-1-4022), the Mobilitas Plus grant MOBERC12 of the Estonian Research Council, and the Estonian Centre of Exellence in IT (EXCITE) funded by ERDF.

A Optimality of Corollary 1

Lemma 8. *If $S = \{x\}$ where $x \xleftarrow{\$} \{1,\dots,N\}$, then there is a q-query algorithm $A^{\mathcal{O}_S^{SC}}$ such that*

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_S^{SC}}()] \geq \frac{4q-3}{N} - \frac{8q(q-1)}{N^2}$$

Proof. The algorithm is as follows:

- Make the first query with amplitude $1/\sqrt{N}$ in all positions.
- Between queries, transform the state by the unitary $U := 2E/N - I$ where E is the matrix containing 1 everywhere. That U is unitary follows since $U^\dagger U = 4E^2/N^2 - 4E/N + I = I$ using $E^2 = NE$.

One may calculate by induction that the final non-normalized state has amplitude

$$\left(1 - \frac{2}{N}\right)^{q-1} \cdot \frac{1}{\sqrt{N}}$$

in all positions except for the xth one (where the amplitude is 0), so its squared norm is

$$1 - \Pr[\mathsf{Find}] = \left(1 - \frac{2}{N}\right)^{2q-2} \cdot \frac{1}{N} \cdot (N-1) = \left(1 - \frac{2}{N}\right)^{2q-2} \cdot \left(1 - \frac{1}{N}\right)$$

As a function of $1/N$, this expression's derivatives alternate on $[0, 1/2]$, so it is below its second-order Taylor expansion:

$$1 - \Pr[\mathsf{Find}] \leq 1 - \frac{4q-3}{N} + \frac{8q(q-1)}{N^2}$$

This completes the proof. □

References

1. Ambainis, A.: Quantum lower bounds by quantum arguments. J. Comput. Syst. Sci. 64(4), 750–767 (Jun 2002), http://dx.doi.org/10.1006/jcss.2002.1826
2. Ambainis, A., Hamburg, M., Unruh, D.: Quantum security proofs using semi-classical oracles. IACR ePrint 2018/904 (2019), full version of this paper
3. Ambainis, A., Rosmanis, A., Unruh, D.: Quantum attacks on classical proof systems: The hardness of quantum rewinding. In: 55th FOCS. pp. 474–483. IEEE Computer Society Press (Oct 2014)
4. Balogh, M., Eaton, E., Song, F.: Quantum collision-finding in non-uniform random functions. In: Lange, T., Steinwandt, R. (eds.) Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018. pp. 467–486. Springer, Heidelberg (2018)
5. Beals, R., Buhrman, H., Cleve, R., Mosca, M., de Wolf, R.: Quantum lower bounds by polynomials. J. ACM 48(4), 778–797 (Jul 2001), http://doi.acm.org/10.1145/502090.502097
6. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Ashby, V. (ed.) ACM CCS 93. pp. 62–73. ACM Press (Nov 1993)
7. Bellare, M., Rogaway, P.: Optimal asymmetric encryption. In: Santis, A.D. (ed.) EUROCRYPT'94. LNCS, vol. 950, pp. 92–111. Springer, Heidelberg (May 1995)
8. Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 41–69. Springer, Heidelberg (Dec 2011)
9. Boneh, D., Eskandarian, S., Fisch, B.: Post-quantum EPID group signatures from symmetric primitives. Cryptology ePrint Archive, Report 2018/261 (2018), https://eprint.iacr.org/2018/261
10. Boyer, M., Brassard, G., Høyer, P., Tapp, A.: Tight bounds on quantum searching. Fortschritte der Physik 46(4-5), 493–505 (1998)
11. Chase, M., Derler, D., Goldfeder, S., Orlandi, C., Ramacher, S., Rechberger, C., Slamanig, D., Zaverucha, G.: Post-quantum zero-knowledge and signatures from symmetric-key primitives. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) ACM CCS 17. pp. 1825–1842. ACM Press (Oct / Nov 2017)
12. Chen, M.S., Hülsing, A., Rijneveld, J., Samardjiska, S., Schwabe, P.: SOFIA: $\mathcal{MQ}$-based signatures in the QROM. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part II. LNCS, vol. 10770, pp. 3–33. Springer, Heidelberg (Mar 2018)
13. Derler, D., Ramacher, S., Slamanig, D.: Post-quantum zero-knowledge proofs for accumulators with applications to ring signatures from symmetric-key primitives. In: Lange, T., Steinwandt, R. (eds.) Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018. pp. 419–440. Springer, Heidelberg (2018)
14. Eaton, E.: Leighton-Micali hash-based signatures in the quantum random-oracle model. In: Adams, C., Camenisch, J. (eds.) SAC 2017. LNCS, vol. 10719, pp. 263–280. Springer, Heidelberg (Aug 2017)

15. Ebrahimi, E.E., Unruh, D.: Quantum collision-resistance of non-uniformly distributed functions: upper and lower bounds. Quantum Information & Computation 18(15&16), 1332–1349 (2018), http://www.rintonpress.com/xxqic18/qic-18-1516/1332-1349.pdf
16. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO'86. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (Aug 1987)
17. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. Journal of Cryptology 26(1), 80–101 (Jan 2013)
18. Galbraith, S.D., Petit, C., Silva, J.: Identification protocols and signature schemes based on supersingular isogeny problems. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part I. LNCS, vol. 10624, pp. 3–33. Springer, Heidelberg (Dec 2017)
19. Hofheinz, D., Hövelmanns, K., Kiltz, E.: A modular analysis of the Fujisaki-Okamoto transformation. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 341–371. Springer, Heidelberg (Nov 2017)
20. Hövelmanns, K., Kiltz, E., Schäge, S., Unruh, D.: Generic authenticated key exchange in the quantum random oracle model. Cryptology ePrint Archive, Report 2018/928 (2018), https://eprint.iacr.org/2018/928
21. Hülsing, A., Rijneveld, J., Song, F.: Mitigating multi-target attacks in hash-based signatures. In: Cheng, C.M., Chung, K.M., Persiano, G., Yang, B.Y. (eds.) PKC 2016, Part I. LNCS, vol. 9614, pp. 387–416. Springer, Heidelberg (Mar 2016)
22. Jiang, H., Zhang, Z., Chen, L., Wang, H., Ma, Z.: IND-CCA-secure key encapsulation mechanism in the quantum random oracle model, revisited. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part III. LNCS, vol. 10993, pp. 96–125. Springer, Heidelberg (Aug 2018)
23. Jiang, H., Zhang, Z., Ma, Z.: Key encapsulation mechanism with explicit rejection in the quantum random oracle model, unpublished manuscript, first revision of [24], personal communication
24. Jiang, H., Zhang, Z., Ma, Z.: Key encapsulation mechanism with explicit rejection in the quantum random oracle model. Cryptology ePrint Archive, Report 2019/052 (2019), https://eprint.iacr.org/2019/052
25. Leighton, F.T., Micali, S.: Large provably fast and secure digital signature schemes based on secure hash functions. US Patent 5,432,852 (1995)
26. Nielsen, M., Chuang, I.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge, first edn. (2000)
27. Saito, T., Xagawa, K., Yamakawa, T.: Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part III. LNCS, vol. 10822, pp. 520–551. Springer, Heidelberg (Apr / May 2018)
28. Song, F., Yun, A.: Quantum security of NMAC and related constructions - PRF domain extension against quantum attacks. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part II. LNCS, vol. 10402, pp. 283–309. Springer, Heidelberg (Aug 2017)
29. Targhi, E.E., Tabia, G.N., Unruh, D.: Quantum collision-resistance of non-uniformly distributed functions. In: Takagi, T. (ed.) Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016. pp. 79–85. Springer, Heidelberg (2016)
30. Targhi, E.E., Unruh, D.: Quantum security of the Fujisaki-Okamoto and OAEP transforms. In: TCC 2016-B. LNCS, vol. 9986, pp. 192–216. Springer (2016)

31. Unruh, D.: Quantum position verification in the random oracle model. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part II. LNCS, vol. 8617, pp. 1–18. Springer, Heidelberg (Aug 2014)
32. Unruh, D.: Non-interactive zero-knowledge proofs in the quantum random oracle model. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part II. LNCS, vol. 9057, pp. 755–784. Springer, Heidelberg (Apr 2015)
33. Unruh, D.: Revocable quantum timed-release encryption. Journal of the ACM 62(6), 49:1–76 (2015), preprint on IACR ePrint 2013/606
34. Unruh, D.: Post-quantum security of Fiat-Shamir. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part I. LNCS, vol. 10624, pp. 65–95. Springer, Heidelberg (Dec 2017)
35. Yoo, Y., Azarderakhsh, R., Jalali, A., Jao, D., Soukharev, V.: A post-quantum digital signature scheme based on supersingular isogenies. In: Kiayias, A. (ed.) FC 2017. LNCS, vol. 10322, pp. 163–181. Springer, Heidelberg (Apr 2017)
36. Zalka, C.: Grover's quantum searching algorithm is optimal. Phys. Rev. A 60, 2746–2751 (Oct 1999), https://arxiv.org/abs/quant-ph/9711070
37. Zhandry, M.: How to construct quantum random functions. In: 53rd FOCS. pp. 679–687. IEEE Computer Society Press (Oct 2012)
38. Zhandry, M.: Secure identity-based encryption in the quantum random oracle model. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 758–775. Springer, Heidelberg (Aug 2012)
39. Zhandry, M.: A note on the quantum collision and set equality problems. Quantum Information and Computation 15(7&8) (2015)