Quantum security proofs using semi-classical oracles

Andris Ambainis
University of Latvia

Mike Hamburg
Rambus Security Division

Dominique Unruh
University of Tartu

September 24, 2018

Abstract

We present an improved version of the one-way to hiding (O2H) lemma by Unruh, J ACM 2015. Our new O2H lemma 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).

1 Introduction

> [This is a draft. While the technical parts are finished, the introduction lacks a discussion of the background, motivation, and related work. It is suitable mainly for readers who are already familiar with the idea of O2H lemmas. We make this version available at this stage to give researchers developing proofs using O2H lemmas to use our lemmas already now. Comments are highly welcome.]

We present an improved version of the one-way to hiding (O2H) lemma from [Unr15]. Our improved O2H lemma has the following features:

- **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 lemma.
- **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 lemma 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 lemma 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 lemma is used.) Introducing semi-classical oracles is a bit more complicated (or at least involves some non-standard oracles), so we give variants of our lemma both with and without semi-classical oracles to give the user the flexibility to chose.
- **Query depth.** Our O2H lemma 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 $d := q$).

2 Preliminaries

For basics of quantum computing, we refer to a standard textbook such as [NC00].

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 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 $Q_1, \dots, Q_n$ and $R_1, \dots, R_n$ 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 $i = 1, \dots, n$. (I.e., if $\mathcal{O}$ is implemented by U_f, we apply $U_f \otimes \dots \otimes U_f$ 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.

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 $H(x)$. The following lemma shows why this is called "puncturing": when Find doesn't 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.

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

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

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

Then

$$|P_{\text{left}} - P_{\text{right}}| \leq 2\sqrt{d \cdot P_{\text{find}}} \quad \textit{and} \quad \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \leq 2\sqrt{d \cdot P_{\text{find}}} \tag{2}$$

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

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

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

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

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

$$P_{\text{right}} := \Pr[b = 1 \lor \mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. \tag{7}$$

The proof is given in Appendix B.

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 at most q queries 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 $i \overset{\$}{\leftarrow} \{1, \dots, d\}$; 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

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_{S}^{SC}}(z)] \leq 4d \cdot \Pr[S \cap T \neq \varnothing : T \leftarrow B(z)] \tag{8}$$

The proof is given in Appendix C.

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*

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_S^{SC}}(z)] \le 4q \cdot P_{\max}. \tag{9}$$

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

Proof. A makes q queries at depth d, so $\mathrm{Exp}[\mathrm{card}\,(T) : T \leftarrow B(z)] \le q/d$ by definition. Therefore

$$\Pr[S \cap T \ne \varnothing : T \leftarrow B(z)] \le (q/d) \cdot P_{\max}$$

Then by Theorem 2,

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_S^{SC}}(z)] \le 4d \cdot (q/d) \cdot P_{\max} = 4q \cdot P_{\max}. \qquad \square$$

3.1 Regular O2H, revisited

Note that the use of semi-classical oracles in Theorem 1 is entirely optional. If we use variant (2) and apply Theorem 2 to P_{find}, we get a variant of Theorem 1 that does not involve semi-classical oracles. The result is essentially the following Theorem 3. However, proving Theorem 3 directly gives a better bound: $2d\sqrt{P_{guess}}$ instead of $4d\sqrt{P_{guess}}$.

Theorem 3 (One-way to hiding, probabilities) *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 quantum oracle algorithm with query depth d (not necessarily unitary).

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

Let

$$\begin{aligned} P_{\mathrm{left}} &:= \Pr[b = 1 : b \leftarrow A^H(z)] \\ P_{\mathrm{right}} &:= \Pr[b = 1 : b \leftarrow A^G(z)] \\ P_{\mathrm{guess}} &:= \Pr[S \cap T \ne \varnothing : T \leftarrow B^H(z)] \end{aligned}$$

Then

$$|P_{\text{left}} - P_{\text{right}}| \leq 2d\sqrt{P_{\text{guess}}} \qquad \text{and} \qquad \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \leq 2d\sqrt{P_{\text{guess}}}$$

The same result holds with B^G instead of B^H in the definition of P_{guess}.

As we said already, except for a factor of 2 in the bound, this is an immediate corollary of Theorem 1 and Theorem 2. To get the slightly better bound in Theorem 3, we use a direct proof. See Appendix D.

The original O2H [Unr15, Lemma 6.2] is an immediate consequence of Theorem 3: Pick H uniformly, pick x, y uniformly, set $G(\cdot) := y$, $I := \{x\}$ and $z := (x, H(x))$. Then P_{left} and P_{guess} are as in the original O2H. P_{right} is $\Pr[b = 1 : b \leftarrow A^{H(x:=y)}(x, H(x))]$, but this is the same as $\Pr[b = 1 : b \leftarrow A^H(x, y)]$.

This also implies that using Theorem 1 and Theorem 2 instead can never give worse bounds than the original O2H, except by a factor of 2.

4 Examples how to use the O2H lemmas

To illustrate the use of the lemmas 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 lemma (Theorem 1), variant (5)[1] to this game, where $G := 0$ (the constant zero function) and $S := \{x : H(x) = 1\}$. This gives us:

$$\left|\sqrt{\underbrace{\Pr[b = 1 : \text{Game 2}]}_{P_{\text{left}}}} - \sqrt{\underbrace{\Pr[b = 1 \wedge \neg\mathsf{Find} : \text{Game 3}]}_{P_{\text{right}}}}\right| \\ \leq \sqrt{(d+2)\underbrace{\Pr[\mathsf{Find} : \text{Game 3}]}_{P_{\text{find}}}} \tag{10}$$

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 lemma 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 this specific case, it is all zero. The difference between the two games is bounded by (10) 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 lemma: 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., [Unr15]). First, prior O2H lemmas 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, (10) 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 [Unr15]). However, as we will see in the next section, in some cases semi-classical oracles give better bounds.

[1] Theorem 1 gives us different options how to define the right game. Conceptually simplest is variant (2) (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 our case, we trivially have $\Pr[G(x)=1 \land \neg\mathsf{Find} : \text{Game 4}] = 0$ 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. Appendix E 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 $\Pr[\mathsf{Find} : \text{Game 4}] \le 4(q+1)\lambda$.

Plugging this into (10), we get

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

Without the square roots on the left-hand side of (10), 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 $\Pr[H(x)=1] \le \lambda$ for all x. Let A be a q-query adversary with query depth d. Then $\Pr[H(x)=1 : b \leftarrow A^H()] \le 4(d+2)(q+1)\lambda$.*

While this is a simple consequence of our O2H technique, we are not aware that this bound was already presented in the literature. While [Zal99] 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 [HRS16, Theorem 1] which we improve upon by a factor of 2.

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 lemmas 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 subsection 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 (5), we find that

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

Unlike in subsection 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 subsection 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 (5) as before gives

$$|\sqrt{w_3} - \sqrt{w_2}| \leq \sqrt{(d+2) \Pr\left[\mathsf{Find} : \text{Game } 3\right]} \tag{11}$$

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 (11) and squaring both sides gives:

$$w_0 = w_2 \le (d+2)\Pr\left[\mathsf{Find} : \text{Game 3}\right] \tag{12}$$

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 don't care about the output of $\hat{A}$, but only whether it* Find*s.*

Clearly $\Pr\left[\mathsf{Find} : \text{Game 4}\right] = \Pr\left[\mathsf{Find} : \text{Game 3}\right]$. 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

$$\mathrm{Adv}_{\mathrm{IND}-E}(B) = \left|\Pr\left[\mathsf{Find} : \text{Game 5}\right] - \Pr\left[\mathsf{Find} : \text{Game 4}\right]\right| \tag{13}$$

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,

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

Combining this with (12) and (13), we have

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

This is a much better bound than we would have gotten without using semi-classical oracles (i.e., using Theorem 3 or the O2H lemma from [Unr15]). 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 [Unr15], 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 Theorem 3 would only require E to be one-way.

5 Acknowledgements

Thanks to Daniel Kane and Eike Kiltz for helpful discussions. 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), and the Mobilitas Plus grant MOBERC12 of the Estonian Research Council.

References

[HRS16] Andreas Hülsing, Joost Rijneveld, and Fang Song. Mitigating multi-target attacks in hash-based signatures. In Chen-Mou Cheng, Kai-Min Chung, Giuseppe Persiano, and Bo-Yin Yang, editors, *PKC 2016, Part I*, volume 9614 of *LNCS*, pages 387–416. Springer, Heidelberg, March 2016. `doi:10.1007/978-3-662-49384-7_15`.

[NC00] M. Nielsen and I. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, first edition, 2000.

[Unr15] Dominique Unruh. Revocable quantum timed-release encryption. *Journal of the ACM*, 62(6):49:1–76, 2015. Preprint on IACR ePrint 2013/606.

[Zal99] Christof Zalka. Grover's quantum searching algorithm is optimal. *Phys. Rev. A*, 60:2746–2751, Oct 1999. URL: `https://arxiv.org/abs/quant-ph/9711070`, `doi:10.1103/PhysRevA.60.2746`.

A Auxiliary lemmas

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

$$\sqrt{P_1P_2} + \sqrt{(1-P_1)(1-P_2)} \geq F(\rho_1, \rho_2) \tag{14}$$

Also,

$$\left|\sqrt{P_1} - \sqrt{P_2}\right| \leq B(\rho_1, \rho_2). \tag{15}$$

Furthermore,

$$|P_1 - P_2| \leq \mathrm{TD}(\rho_1, \rho_2) \leq B(\rho_1, \rho_2). \tag{16}$$

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 $\operatorname{tr}\rho = 1$, $\mathcal{E}(\rho) = \begin{pmatrix} p & 0 \\ 0 & \bar{p}\end{pmatrix}$ where p is the probability that M returns 1 when measuring ρ.

Then $\rho_i' := \mathcal{E}(\rho_i) = \begin{pmatrix} P_i & 0 \\ 0 & \bar{P}_i\end{pmatrix}$ 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 [NC00, Thm. 9.6], and $(**)$ is the definition of fidelity. This shows (14). To prove (15), we compute:

$$\begin{aligned}\left(\sqrt{P_1}-\sqrt{P_2}\right)^2 &= P_1+P_2-2\sqrt{P_1P_2}\\ &\leq P_1+P_2-2\sqrt{P_1P_2}+\left(\sqrt{\bar P_1}-\sqrt{\bar P_2}\right)^2\\ &= 2-2\sqrt{P_1P_2}-2\sqrt{\bar P_1\bar P_2}\overset{(14)}{\leq} 2-2F(\rho_1,\rho_2)\overset{(*)}{=}B(\rho_1,\rho_2)^2\end{aligned}$$

where $(*)$ is by definition of the Bures distance. This implies (15).

The first inequality in (16) is well-known (e.g., [NC00, 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 [NC00, (9.101)], and $(**)$ is because $0\leq F(\rho,\tau)\leq 1$. □

B 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:*

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

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, $\big\|(I\otimes(I-|0\rangle\langle 0|))|\Psi_{\mathrm{right}}\rangle\big\|^2$.)

Then

$$\big\| |\Psi_{\text{left}}\rangle \otimes |0\rangle - |\Psi_{\text{right}}\rangle \big\|^2 \le (d+1)\tilde P_{\text{find}}.$$

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

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

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

$$|\Psi_{\text{left}}\rangle = \sum_{i=0}^{d} |\Psi'_i\rangle. \tag{18}$$

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

$$|\Psi_{\text{right}}\rangle = \sum_{l\in\{0,1\}^q} |\Psi_l\rangle|l\rangle_L \tag{19}$$

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 (17) and (19), the result of that projection is $|\Psi_0\rangle|0\rangle_L$ and $|\Psi'_0\rangle|0\rangle_C$, respectively. Hence

$$|\Psi_0\rangle = |\Psi'_0\rangle. \tag{20}$$

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

$$\big\||\Psi_0\rangle\big\|^2 = 1 - \tilde{P}_{\text{find}}. \tag{21}$$

We have

$$\sum_{i=0}^{d}\big\||\Psi'_i\rangle\big\|^2 = \sum_{i=0}^{d}\big\||\Psi'_i\rangle|i\rangle_C\big\|^2 = \Big\|\sum_{i=0}^{d}|\Psi'_i\rangle|i\rangle_C\Big\|^2 \overset{(17)}{=} \big\||\Psi_{\text{count}}\rangle\big\|^2 = 1.$$

$$\sum_{l\in\{0,1\}^d}\big\||\Psi_l\rangle\big\|^2 = \sum_{l\in\{0,1\}^d}\Big\||\Psi_l\rangle|l\rangle_L\Big\|^2 = \Big\|\sum_{l\in\{0,1\}^d}|\Psi_l\rangle|l\rangle_L\Big\|^2 \overset{(19)}{=} \big\||\Psi_{\text{right}}\rangle\big\|^2 = 1.$$

Thus

$$\sum_{i=1}^{d}\big\||\Psi'_i\rangle\big\|^2 = 1 - \big\||\Psi'_0\rangle\big\|^2 \overset{(21)}{=} \tilde{P}_{\text{find}}, \qquad \sum_{\substack{l\in\{0,1\}^d\\ l\neq 0}}\big\||\Psi_l\rangle\big\|^2 = 1 - \big\||\Psi_0\rangle\big\|^2 \overset{(21)}{=} \tilde{P}_{\text{find}}. \tag{22}$$

Therefore

$$\begin{aligned}
\Big\||\Psi_{\text{right}}\rangle - |\Psi_{\text{left}}\rangle|0\rangle_L\Big\|^2 &\overset{(19)}{=} \Big\|\big(|\Psi_0\rangle - |\Psi_{\text{left}}\rangle\big)|0\rangle + \sum_{\substack{l\in\{0,1\}^d\\ l\neq 0}}|\Psi_l\rangle|l\rangle\Big\|^2 \\
&= \Big\||\Psi_0\rangle - |\Psi_{\text{left}}\rangle\Big\|^2 + \sum_{\substack{l\in\{0,1\}^d\\ l\neq 0}}\big\||\Psi_l\rangle\big\|^2 \overset{(22)}{=} \Big\||\Psi_0\rangle - |\Psi_{\text{left}}\rangle\Big\|^2 + \tilde{P}_{\text{find}} \\
&\overset{(20),(18)}{=} \Big\|\sum_{i=1}^{d}|\Psi'_i\rangle\Big\|^2 + \tilde{P}_{\text{find}} \overset{(*)}{\le} \Big(\sum_{i=1}^{d}\Big\||\Psi'_i\rangle\Big\|\Big)^2 + \tilde{P}_{\text{find}} \overset{(**)}{\le} d\cdot\sum_{i=1}^{d}\Big\||\Psi'_i\rangle\Big\|^2 + \tilde{P}_{\text{find}} \\
&\overset{(22)}{=} d\tilde{P}_{\text{find}} + \tilde{P}_{\text{find}} = (d+1)\tilde{P}_{\text{find}}.
\end{aligned}$$

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

Lemma 6 (O2H in terms of mixed states) *Let H, I, z be random. (With some joint distribution.)*

Let A be an algorithm with query depth d. Let B and P_{find} be as in Theorem 1.

Let ρ_{left} denote the final state of A.

Let ρ_{right} denote the final state of Q_A, L, where Q_A is the register used for its state by B (or A), and L is a register that contains the log of the responses of $\mathcal{O}_I^{SC}$. If the i-th query to $\mathcal{O}_I^{SC}$ 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 $\big|\Psi_{\text{left}}^{HIz}\big\rangle$ be the state $\big|\Psi_{\text{left}}\big\rangle$ from Lemma 5 for specific values of H, I, z. And analogously for $\big|\Psi_{\text{right}}^{HIz}\big\rangle$ and $\tilde{P}_{\text{find}}^{HIz}$.

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

Furthermore, if we define $\rho'_{\text{right}} := \operatorname{Exp}_{HIz}[|\Psi_{\text{right}}^{HIz}\rangle\langle\Psi_{\text{right}}^{HIz}|]$, 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 $P_{\text{find}} = \operatorname{Exp}_{HIz}[\tilde{P}_{\text{find}}^{HIz}]$.

Then

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

Here $(*)$ follows from the monotonicity of the fidelity [NC00, Thm. 9.6], and $(**)$ follows from the joint concavity of the fidelity [NC00, (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 $B(\rho_{\text{left}}, \rho_{\text{right}}) \leq \sqrt{(d+1)P_{\text{find}}}$. □

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

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

Then

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

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

$$\begin{aligned}P_{\text{right}} &:= \Pr[b = 1 : b \leftarrow A^{H\setminus S}(z)], &(3)\\ P_{\text{right}} &:= \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)], &(4)\\ P_{\text{right}} &:= \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)], &(5)\\ P_{\text{right}} &:= \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{H\setminus S}(z)], &(6)\\ P_{\text{right}} &:= \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. &(7)\end{aligned}$$

Proof. We first prove the theorem using the definition of P_{right} from (3).

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

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

This shows the theorem with the definition of P_{right} from (3).

Now we show the theorem using the definition of P_{right} from (4). 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 (3).

Now we show the theorem using the definition of P_{right} from (6). 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 (3).

Now we show the theorem using the definition of P_{right} from (5). This follows immediately by case (4), 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 (7). By Lemma 1,

$$\Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{H\setminus S}(z)] = \Pr[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{G\setminus S}(z)] \tag{23}$$

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

From (24), we get (by considering the complementary event):

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

Adding (23) and (25), we get

$$\Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{H\setminus S}(z)] = \Pr[b = 1 \vee \mathsf{Find} : b \leftarrow A^{G\setminus S}(z)]. \tag{26}$$

Then case (7) follows from case (6) and the fact (26).

Now we show the theorem using the definition of P_{right} from (2). 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 (4) (which we already proved), we have

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

and by case (5), we also get

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

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 (25), $P_{\text{find}} = P'_{\text{find}}$. With this and the triangle inequality, we get

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

as required. □

C 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

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

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 $Q_1, \dots, Q_n$ 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 $Q_1, \dots, Q_n$ 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

$$\begin{aligned} r_i &:= p_i + 2\||\Psi_i\rangle - |\Psi_i'\rangle\|^2 = 1 - \||\Psi_i\rangle\|^2 + 2\||\Psi_i\rangle\|^2 - 4\Re\langle\Psi_i'|\Psi_i\rangle + 2\underbrace{\||\Psi_i'\rangle\|^2}_{=1} \\ &= 3 - 4\Re\langle\Psi_i'|\Psi_i\rangle + \||\Psi_i\rangle\|^2 \end{aligned} \tag{27}$$

Notice that $r_0 = 0$ since $|\Psi_0\rangle = |\Psi_0'\rangle$ and $\||\Psi_0\rangle\| = 1$. During the 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,

$$\begin{aligned} \||\Psi_{i+1}\rangle\|^2 &= \|U|\Psi_i\rangle\|^2 - \big\langle\Psi_i\big|U^\dagger P_S U\big|\Psi_i\big\rangle - \big\langle\Psi_i\big|U^\dagger P_S^\dagger U\big|\Psi_i\big\rangle + \big\langle\Psi_i\big|U^\dagger P_S^\dagger P_S U\big|\Psi_i\big\rangle \\ &= \||\Psi_i\rangle\|^2 - \big\langle\Psi_i\big|U^\dagger P_S U\big|\Psi_i\big\rangle \end{aligned} \tag{28}$$

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

$$\begin{aligned} \langle\Psi_{i+1}'|\Psi_{i+1}\rangle &= \big\langle\Psi_i'\big|U^\dagger U\big|\Psi_i\big\rangle - \big\langle\Psi_i'\big|U^\dagger P_S U\big|\Psi_i\big\rangle \\ &= \langle\Psi_i'|\Psi_i\rangle - \big\langle\Psi_i'\big|U^\dagger P_S U\big|\Psi_i\big\rangle \end{aligned} \tag{29}$$

Let

$$g_i := \langle\Psi_i'|U^\dagger P_S U|\Psi_i'\rangle$$

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

We calculate

$$\begin{aligned} r_{i+1} - r_i &\overset{(27)}{=} -4\Re\langle\Psi_{i+1}'|\Psi_{i+1}\rangle + \||\Psi_{i+1}\rangle\|^2 + 4\Re\langle\Psi_i'|\Psi_i\rangle - \||\Psi_i\rangle\|^2 \\ &\overset{(28),(29)}{=} 4\Re\langle\Psi_i'|U^\dagger P_S U|\Psi_i\rangle - \langle\Psi_i|U^\dagger P_S U|\Psi_i\rangle \\ &= 4\langle\Psi_i'|U^\dagger P_S U|\Psi_i'\rangle - \underbrace{\langle 2\Psi_i' - \Psi|U^\dagger P_S U|2\Psi_i' - \Psi_i\rangle}_{\geq 0} \\ &\leq 4\langle\Psi_i'|U^\dagger P_S U|\Psi_i'\rangle = 4g_{i+1} \end{aligned}$$

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 at most q queries 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 $i \overset{\$}{\leftarrow} \{1,\dots,d\}$; 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

$$\Pr[\mathsf{Find} : A^{\mathcal{O}_S^{SC}}(z)] \le 4d \cdot \Pr[S \cap T \neq \varnothing : T \leftarrow B(z)] \tag{8}$$

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

D Proof of Theorem 3

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

Lemma 8 (One-way to hiding, pure states) *Fix G, H, S, z satisfying $\forall x \notin S.\ G(x) = H(x)$. ($G, H, S, z$ are not randomized in this lemma.) Let $A^H(z)$ be a unitary quantum oracle algorithm with query depth d. Let Q_A denote the register containing all of A's state.*

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

Let $|\Psi_{\mathrm{left}}\rangle$ be the final state of A after running $A^H(z)$. And let $|\Psi_{\mathrm{right}}\rangle$ be the final state of A after running $A^G(z)$.

Let

$$P_{\mathrm{guess}} := \Pr[S \cap T \neq \varnothing : T \leftarrow B^H(z)]$$

Then $\big\| |\Psi_{\mathrm{left}}\rangle - |\Psi_{\mathrm{right}}\rangle \big\| \le 2d\sqrt{P_{\mathrm{guess}}}$.

Proof. The state of A is composed of three quantum systems A, Q, R where Q,R are the query and the response register for oracle queries. (That is, Q consists of a number of registers $Q_1, \dots, Q_n$ where r is the maximum number of queries performed in parallel, and R consists of corresponding registers $R_1, \dots, R_n$.) Then an execution of $A^H(z)$ leads to the final state $(UO_H)^q|\Psi_0\rangle$ where $|\Psi_0\rangle$ is an initial state that depends on z (but not on G, H, or S), $O_H : |a, q_1, \dots, q_n, r_1, \dots, r_n\rangle \mapsto |a, q_1, \dots, q_n, r_1 \oplus H(q_1), \dots, r_n \oplus H(q_n)\rangle$ is an oracle query, and U is A's state transition operation. (And analogously for A^G.)

We define $|\Psi_H^i\rangle := (UO_H)^i|\Psi_0\rangle$ and similarly $|\Psi_G^i\rangle$. Then $|\Psi_{\text{left}}\rangle = |\Psi_H^d\rangle$ and $|\Psi_{\text{right}}\rangle = |\Psi_G^d\rangle$.

And in our notation, we can describe B as follows: $B^H(x)$ picks $i \overset{\$}{\leftarrow} \{1,\dots,d\}$ and $y \overset{\$}{\leftarrow} Y$, measures the quantum system Q of the state $|\Psi_H^{i-1}\rangle$ (this gives a list T of inputs), and outputs the result T. Thus

$$P_{\text{guess}} = \tfrac{1}{q}\big\|P_S|\Psi_H^{i-1}\rangle\big\|^2 = \sum_{i=1}^{q} \tfrac{1}{q}B_i \qquad \text{with} \qquad B_i := \big\|P_S|\Psi_H^{i-1}\rangle\big\|^2. \tag{30}$$

Here P_S is the orthogonal projector projecting Q onto states $|T\rangle$ with $S\cap T \neq \varnothing$, formally $P_S := \sum_{T\,\text{s.t.}\,S\cap T\neq\varnothing}|T\rangle\langle T|$. (I.e., $\|P_S|\Psi_H^{i-1}\rangle\|^2$ is the probability of measuring T with $S\cap T\neq\varnothing$ in register Q given the state $|\Psi_H^{i-1}\rangle$.)

Let $D_i := \big\||\Psi_H^i\rangle - |\Psi_G^i\rangle\big\|^2$. We have $D_0 = \big\||\Psi_0\rangle - |\Psi_0\rangle\big\|^2 = 0$, and for $i \geq 1$ we have:

$$\begin{aligned}
D_i &= \big\|UO_H|\Psi_H^{i-1}\rangle - UO_G|\Psi_G^{i-1}\rangle\big\|^2 \\
&\overset{(*)}{=} \big\|(O_H|\Psi_H^{i-1}\rangle - O_G|\Psi_H^{i-1}\rangle) + (O_G|\Psi_H^{i-1}\rangle - O_G|\Psi_G^{i-1}\rangle)\big\|^2 \\
&\overset{(**)}{\leq} \big\|(O_H - O_G)|\Psi_H^{i-1}\rangle\big\|^2 + \big\|O_G(|\Psi_H^{i-1}\rangle - |\Psi_G^{i-1}\rangle)\big\|^2 \\
&\qquad + 2\big\|(O_H - O_G)|\Psi_H^{i-1}\rangle\big\| \cdot \big\|O_G(|\Psi_H^{i-1}\rangle - |\Psi_G^{i-1}\rangle)\big\| \\
&\overset{(***)}{=} \big\|(O_H - O_G)P_S|\Psi_H^{i-1}\rangle\big\|^2 + \big\||\Psi_H^{i-1}\rangle - |\Psi_G^{i-1}\rangle\big\|^2 \\
&\qquad + 2\big\|(O_H - O_G)P_S|\Psi_H^{i-1}\rangle\big\| \cdot \big\||\Psi_H^{i-1}\rangle - |\Psi_G^{i-1}\rangle\big\| \\
&\overset{(****)}{\leq} 4\underbrace{\big\|P_S|\Psi_H^{i-1}\rangle\big\|^2}_{=B_i} + \underbrace{\big\||\Psi_H^{i-1}\rangle - |\Psi_G^{i-1}\rangle\big\|^2}_{=D_{i-1}} \\
&\qquad + 4\underbrace{\big\|P_S|\Psi_H^{i-1}\rangle\big\|}_{=\sqrt{B_i}} \cdot \underbrace{\big\||\Psi_H^{i-1}\rangle - |\Psi_G^{i-1}\rangle\big\|}_{=\sqrt{D_i}} \\
&= 4B_i + D_{i-1} + 4\sqrt{B_iD_{i-1}} = (\sqrt{D_{i-1}} + 2\sqrt{B_i})^2.
\end{aligned} \tag{31}$$

Here $(*)$ uses that U is unitary. And $(**)$ uses the inequality $\|a+b\|^2 \leq \|a\|^2 + \|b\|^2 + 2\|a\|\cdot\|b\|$. And $(***)$ uses that $(O_H - O_G)P_S = O_H - O_g$ since $G = H$ outside of S (this can be verified by checking on all basis states $|a, q_1, \dots, r_1, \dots\rangle$), and that O_G is unitary. And $(****)$ follows since $O_H - O_G$ has operator norm ≤ 2.

From (31), we get $\sqrt{D_i} \leq \sqrt{D_{i-1}} + 2\sqrt{B_i}$. This implies (with $D_0 = 0$) that

$$\sqrt{D_d} \leq 2\sum_{i=1}^{d}\sqrt{B_i} = 2d\sum_{i=1}^{d}\tfrac{1}{d}\sqrt{B_i} \overset{(*)}{\leq} 2d\sqrt{\sum_{i=1}^{d}\tfrac{1}{d}B_i} \overset{(30)}{=} 2d\sqrt{P_{\text{guess}}}$$

where $(*)$ follows from Jensen's inequality. By definition of D_q, this shows the lemma. □

Lemma 9 (One-way to hiding, mixed states) *Let G, H, S, z be random satisfying $\forall x \notin S.\ G(x) = H(x)$. (With some joint distribution.)*

Let A be a quantum oracle algorithm with query depth q (not necessarily unitary). Let B and P_{guess} be as in Theorem 3.

Let ρ_{left} be the final state of $A^H(z)$ and let ρ_{right} be the final state of $A^G(z)$

Then $F(\rho_{\text{left}}, \rho_{\text{right}}) \geq 1 - 2d^2 P_{\text{guess}}$ and $B(\rho_{\text{left}}, \rho_{\text{right}}) \leq 2d\sqrt{P_{\text{guess}}}$.

Proof. Without loss of generality, we can assume that A is unitary during the execution, and applies a quantum operation $\mathcal{E}$ to its state in the last step. (Note that transforming an adversary A into a unitary adversary A' may change the internal state during the execution because additional auxiliary qubits are used to simulate measurements. However, this does not affect the probability P_{guess} because B does not measure those auxiliary qubits of A'.)

For fixed G, H, S, z, let $|\Psi^{HSz}_{\text{left}}\rangle, |\Psi^{GSz}_{\text{right}}\rangle, P^{HSz}_{\text{guess}}$ refer to the values $|\Psi_{\text{left}}\rangle, |\Psi_{\text{right}}\rangle, P_{\text{guess}}$ from Lemma 8 for those fixed G, H, S, z.

Let $\hat\rho_{\text{left}}$ and $\hat\rho_{\text{right}}$ refer to the state of A before applying $\mathcal{E}$ in the games defining $\hat\rho_{\text{left}}$ and $\hat\rho_{\text{right}}$, respectively.

Then

$$\hat\rho_{\text{left}} = \operatorname*{Exp}_{GHSz}\bigl[|\Psi^{HSz}_{\text{left}}\rangle\langle\Psi^{HSz}_{\text{left}}|\bigr] \qquad \text{and}$$
$$\hat\rho_{\text{right}} = \operatorname*{Exp}_{GHSz}\bigl[|\Psi^{GSz}_{\text{right}}\rangle\langle\Psi^{GSz}_{\text{right}}|\bigr].$$

Thus we have

$$\begin{aligned}
F(\rho_{\text{left}}, \rho_{\text{right}}) &= F(\mathcal{E}(\hat\rho_{\text{left}}), \mathcal{E}(\hat\rho_{\text{right}})) \overset{(*)}{\geq} F(\hat\rho_{\text{left}}, \hat\rho_{\text{right}}) \\
&= F\Bigl(\operatorname*{Exp}_{HGSz}[|\Psi^{HSz}_{\text{left}}\rangle\langle\Psi^{HSz}_{\text{left}}|], \operatorname*{Exp}_{HGSz}[|\Psi^{GSz}_{\text{right}}\rangle\langle\Psi^{GSz}_{\text{right}}|]\Bigr) \\
&\overset{(**)}{\geq} \operatorname*{Exp}_{HGSz}[F\bigl(|\Psi^{HSz}_{\text{left}}\rangle\langle\Psi^{HSz}_{\text{left}}|, |\Psi^{GSz}_{\text{right}}\rangle\langle\Psi^{GSz}_{\text{right}}|\bigr)] \\
&\overset{\text{Lemma 3}}{\geq} \operatorname*{Exp}_{HGSz}\bigl[1 - \tfrac{1}{2}\bigl\||\Psi^{HSz}_{\text{left}}\rangle - |\Psi^{GSz}_{\text{right}}\rangle\bigr\|^2\bigr] \\
&\overset{\text{Lemma 8}}{\geq} \operatorname*{Exp}_{HGSz}\bigl[1 - \tfrac{1}{2}(4dP^{HSz}_{\text{guess}})\bigr] \overset{(***)}{=} 1 - 2d^2 P_{\text{guess}}.
\end{aligned}$$

Here $(*)$ follows from the monotonicity of the fidelity [NC00, Thm. 9.6], and $(**)$ follows from the joint concavity of the fidelity [NC00, (9.95)]. And $(*{*}*)$ follows since $P_{\text{guess}} = \operatorname{Exp}_{HGSz}[P^{HSz}_{\text{guess}}]$.

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

$$B(\rho_{\text{left}},\rho_{\text{right}})^2 = 2(1-F(\rho_{\text{left}},\rho_{\text{right}})) \le 2(1-(1-2d^2P_{\text{guess}})) = 4d^2P_{\text{guess}},$$

hence $B(\rho_{\text{left}},\rho_{\text{right}}) \le 2d\sqrt{P_{\text{guess}}}$, as claimed. □

Theorem 3 (One-way to hiding, probabilities – 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 quantum oracle algorithm with query depth d (not necessarily unitary).

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

Let

$$\begin{aligned} P_{\text{left}} &:= \Pr[b=1 : b \leftarrow A^H(z)] \\ P_{\text{right}} &:= \Pr[b=1 : b \leftarrow A^G(z)] \\ P_{\text{guess}} &:= \Pr[S \cap T \neq \varnothing : T \leftarrow B^H(z)] \end{aligned}$$

Then

$$|P_{\text{left}} - P_{\text{right}}| \le 2d\sqrt{P_{\text{guess}}} \qquad \textit{and} \qquad \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \le 2d\sqrt{P_{\text{guess}}}$$

The same result holds with B^G instead of B^H in the definition of P_{guess}.

Proof. The output bit b of A is the result of a measurement M applied to its final state. Thus, with $\rho_{A,1},\rho_{A,2}$ as in Lemma 9, $P_{\text{left}},P_{\text{right}}$ is the probability that the measurement M returns 1 when measuring $\rho_{\text{left}},\rho_{\text{right}}$, respectively. By Lemma 4,

$$\left|P_{\text{left}} - P_{\text{right}}\right| \le B(\rho_{\text{left}},\rho_{\text{right}}) \qquad \text{and} \qquad \left|\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}\right| \le B(\rho_{\text{left}},\rho_{\text{right}})$$

By Lemma 9, $B(\rho_{\text{left}},\rho_{\text{right}}) \le 2d\sqrt{P_B}$. The corollary follows. □

E Optimality of Corollary 1

Lemma 10 *If $S = \{x\}$ where $x \overset{\$}{\leftarrow} \{1,\dots,N\}$, then there is a q-query algorithm $A^{\mathcal{O}_S^{SC}}$ such that*

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

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. □

Symbol index

$\mathcal{O}_I^{SC}$	Semi-classical oracle for set I	
Find	Semi-classical $\mathcal{O}_S^{SC}$ returns 1	
$\Delta(X, Y)$	Statistical distance between distributions/random variables X and Y	
$\mathrm{flip}_i(l)$	Flips i-th bit of l	
$F(\rho_1, \rho_2)$	Fidelity between ρ_1 and ρ_2	12
$\mathrm{TD}(\rho_1, \rho_2)$	Trace distance between ρ_1 and ρ_2.	12
$\operatorname{tr} \rho$	Trace of ρ	
$B(\rho_1, \rho_2)$	Bures distance between ρ_1 and ρ_2	12
$\lvert\Psi\rangle$	Refers to a quantum state (or, for $x \in M$, $\lvert x\rangle$ refers to a basis vector of $\mathbb{C}^M$)	
$\operatorname{tr} A\rho$	Partial trace of ρ, removing register A	
$\langle\Psi\rvert$	Adjoint of $\lvert\Psi\rangle$, i.e., $\langle\Psi\rvert^\dagger$	
$\mathbb{C}$	Complex numbers	
$\mathcal{E}$	A quantum operation (superoperator)	
$\mathcal{D}$	A distribution	
$x \xleftarrow{\$} M$	x picked uniformly from the set M	
$H \setminus I$	Oracle H, punctured at I	
$\lvert x\rvert$	Absolute value of x / cardinality of set x	
$x \leftarrow A$	x assigned output of algorithm A / picked according to distribution A	
Guess	Query to fully-quantum oracle is in S	
$\lVert x\rVert$	Norm of x	
$\mathrm{Exp}_z[y]$	Expectation of y, taken over the randomness of z	