Classical vs Quantum Advice and Proofs Under Classically-Accessible Oracle

Xingjian Li
Tsinghua University, Beijing, China

Qipeng Liu
University of California at San Diego, La Jolla, CA, USA

Angelos Pelecanos
University of California at Berkeley, CA, USA

Takashi Yamakawa
NTT Social Informatics Laboratories, Tokyo, Japan

Abstract

It is a long-standing open question to construct a classical oracle relative to which $\mathsf{BQP/qpoly} \neq \mathsf{BQP/poly}$ or $\mathsf{QMA} \neq \mathsf{QCMA}$. In this paper, we construct *classically-accessible classical oracles* relative to which $\mathsf{BQP/qpoly} \neq \mathsf{BQP/poly}$ and $\mathsf{QMA} \neq \mathsf{QCMA}$. Here, classically-accessible classical oracles are oracles that can be accessed only classically even for quantum algorithms. Based on a similar technique, we also show an alternative proof for the separation of QMA and QCMA relative to a distributional quantumly-accessible classical oracle, which was recently shown by Natarajan and Nirkhe.

2012 ACM Subject Classification Theory of computation → Complexity classes; Theory of computation → Quantum computation theory

Keywords and phrases quantum computation, computational complexity

Digital Object Identifier 10.4230/LIPIcs.ITCS.2024.72

Related Version *Full Version*: https://arxiv.org/abs/2303.04298

Funding *Qipeng Liu*: supported in part by the Simons Institute for the Theory of Computing, through a Quantum Postdoctoral Fellowship, by the DARPA SIEVE-VESPA grant No.HR00112020023 and by the NSF QLCI program through grant number OMA-2016245.
Angelos Pelecanos: supported by DARPA under Agreement No. HR00112020023.

1 Introduction

Quantum information is often in possession of richer structures than classical information, at least intuitively. The first (but often false) thought is that phases and magnitudes are continuous, and a piece of quantum information may be able to store exponentially or infinitely more information than classical ones; which is always not true[1]. Since classical and quantum information present distinct and unique natures, the community studies their differences under different contexts and directions, including advice-aided quantum computation [23, 1, 2, 4, 22, 19, 12, 11, 17, 20], QMA vs QCMA (i.e., quantum NP with either quantum or classical witness) [6, 5, 15, 21], quantum vs classical communication complexity [28, 10, 24, 7, 8, 16] and many others.

One way to understand their differences is by studying one-way communication complexity: i.e., Alice and Bob want to jointly compute a function with their private inputs, but only one-time quantum/classical communication from Alice to Bob is allowed. Among many

[1] As storing and extracting information takes resources that scale with accuracy.

15th Innovations in Theoretical Computer Science Conference (ITCS 2024).
Editor: Venkatesan Guruswami; Article No. 72; pp. 72:1–72:19
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

works, Bar-Yossef, Jayram, and Kerenidis [8] showed an exponential separation between quantum and classical one-way communication complexity, for the so-called hidden matching problem.

The other approach is by looking at QMA vs QCMA. In 2007, Aaronson and Kuperberg [5] showed a black-box separation with respect to a black-box quantum unitary and left the same separation with respect to a classical oracle as an open question. More than a decade later, Fefferman and Kimmel [15] proved a second black-box separation using a distributional in-place oracle, which is a non-standard type of oracles. Recently, Natarajan and Nirkhe [21] moved a step closer to the goal by presenting a black-box separation with respect to a distributional oracle[2]. Therefore, we would like to further investigate the difference between quantum and classical proofs, i.e., the separation between QMA vs QCMA. In this work, we address the question by demonstrating a separation relative to a *classically accessible classical* oracle.

▶ **Definition 1** (QMA). *A language $\mathcal{L}$ is said to be in* QMA *if there exists a quantum polynomial-time machine $\mathcal{V}$ together with a polynomial $p(\cdot)$ such that,*

- *For all $x \in \mathcal{L}$, there exists a quantum state $|\psi_x\rangle$ of at most $p(|x|)$ qubits, such that $\mathcal{V}$ accepts on $|x\rangle, |\psi_x\rangle$ with a probability at least $2/3$.*
- *For all $x \notin \mathcal{L}$, for all quantum states $|\psi_x\rangle$ of at most $p(|x|)$ qubits, $\mathcal{V}$ accepts on $|x\rangle, |\psi_x\rangle$ with a probability at most $1/3$.*

One can similarly define the class QCMA except $|\psi_x\rangle$ are of $p(|x|)$ classical bits.

We also aim to understand the difference between quantum and classical information through advice-aided quantum computation. Classically, a piece of advice can significantly speed up classical computation, from speeding up exhaustive search [18] to deciding the unary Halting Problem. In a quantum world, advice can be either a piece of classical or quantum information. It is very natural to ask the question: does quantum advice "outperform" classical advice? Among many questions, one of the most fundamental is to understand the power of BQP/qpoly and BQP/poly: i.e., the class of languages that can be decided by bounded-error quantum machines with arbitrary bounded-length quantum/classical advice and polynomial time.

▶ **Definition 2** (BQP/qpoly). *A language $\mathcal{L}$ is said to be in* BQP/qpoly *if and only if there exists a quantum polynomial-time machine $\mathcal{A}$ together with a collection of polynomial-sized quantum states $\{|z_n\rangle\}_{n\in\mathbb{N}}$ such that,*

- *For all $x \in \mathcal{L}$,* $\Pr\left[\mathcal{A}(x, |z_{|x|}\rangle) = 1\right] \geq 2/3$.
- *For all $x \notin \mathcal{L}$,* $\Pr\left[\mathcal{A}(x, |z_{|x|}\rangle) = 0\right] \geq 2/3$.

One can similarly define the class BQP/poly except $|z_n\rangle$ are poly-sized classical strings.

Similar to the case of QMA vs QCMA, Aaronson and Kuperberg [5] in the same paper showed an oracle separation between these two classes, leaving the separation with respect to a classical oracle as an open question. Recently, Liu [20] showed the separation for its relational variants (i.e., FBQP/qpoly and FBQP/poly) under a special case, where the oracle is never given to the algorithms[3]. Despite all the efforts, the separation between BQP/poly and BQP/qpoly relative to a classical oracle remains obscure.

---

[2] In this case, the witness only depends on the distribution. The oracle is later picked from the distribution, but independent of the witness.

[3] As mentioned in Section 1.3, a concurrent work by Aaronson, Buhrman, and Kretschmer [3] proves their relational variants FBQP/qpoly $\neq$ FBQP/poly *unconditionally.*

In this work, we proceed with the question by showing a *full separation* relative to a *classically accessible classical* oracle. Along the way, we adapt our techniques and give an alternative proof for the separation between QMA vs QCMA relative to a *quantumly accessible classical* distributional oracle, which was recently established by Natarajan and Nirkhe [21].

1.1 Our Results

Our first result is a black-box separation between BQP/qpoly vs BQP/poly with respect to a classically-accessible classical oracle.

Classically-accessible classical oracles. A classical oracle $\mathcal{O}$ is said to be classically accessible if a quantum algorithm can only query the oracle classically; in other words, the only interface of $\mathcal{O}$ to quantum algorithms is classical: given an input x, it outputs $y = \mathcal{O}(x)$.

▶ **Theorem 3** (Informal). *There exists a language $\mathcal{L}$ in* BQP/qpoly *but not in* BQP/poly*, relative to a classically accessible classical oracle $\mathcal{O}$.*

Our work is based on the previous works on quantum advantages with unstructured oracles, by Yamakawa and Zhandry [26] and recent separation by Liu [20]. Our work improves [20] in two aspects: first, Liu only proved the separation of their relational variants, instead of the original decision classes; second, the separation by Liu does not allow algorithms to have access to the oracle (either classically, or quantumly), but only the advice can depend on the oracle.

▶ Remark 4. Although the long-standing open question is to understand the separation relative to a quantumly accessible classical oracle, our theorem is not weaker but incomparable. Since classical access is not stronger than quantum access, it limits both the computational power of quantum machines with quantum and classical advice: intuitively, classical access makes it easier to prove a language $\mathcal{L}$ is not in $\mathsf{BQP}^{\mathcal{O}}/\mathsf{poly}$ but harder to prove $\mathcal{L} \in \mathsf{BQP}^{\mathcal{O}}/\mathsf{qpoly}$. A similar observation is applicable to the theorem concerning the separation between QMA and QCMA.

Our second result is about the separation between QMA and QCMA.

▶ **Theorem 5** (Informal). *There exists a language $\mathcal{L}$ in* QMA *but not in* QCMA*, relative to a classically accessible classical oracle $\mathcal{O}$.*

Inspired by the techniques used in our Theorem 3 and Theorem 5, we give an alternative proof for the result by Natarajan and Nirkhe [21]. Note in the following result, the classical oracle is *quantumly accessible.*

▶ **Theorem 6** (Informal). *There exists a language $\mathcal{L}$ in* QMA *but not in* QCMA*, relative to a distributional quantumly-accessible classical oracle $\mathcal{O}$.*

A difference between the construction of [21] and ours is in the case when the oracle is fixed to some oracle from the distribution. In [21], they can neither prove nor disprove the separation between QMA and QCMA. However, in our construction, the separation will disappear if we fix our oracle.

Finally, we observe that the problem considered in [26] gives a new superpolynomial separation between classical and quantum one-way communication complexity. Though such a separation has been known since 2004 [8], the new separation has two interesting features that Bob's input length is short and the classical lower bound holds even if Bob can classically access Alice's input as an oracle (see the full version for more detail).[4]

[4] As mentioned in Section 1.3, a concurrent work by Aaronson, Buhrman, and Kretschmer [3] observes

1.2 Overview

Quantum Advantages and Separation for a Special Case

We will introduce the recent work on quantum advantages from unstructured oracles by Yamakawa and Zhandry [26], as it will be used for both of our results in this work. They proved that there exists a code $C \subseteq \Sigma^n$ and an oracle-aided function f_C such that, relative to a random oracle from $H : [n] \times \Sigma \to \{0,1\}$, the function f_C^H is easy to invert on any image with quantum access but inversion is hard with only classical access. The function is defined as the following:

$$f_C^H(v_1, \cdots, v_n) = \begin{cases} H(1, v_1)||H(2, v_2)|| \cdots ||H(n, v_n) & \text{if } (v_1, \cdots, v_n) \in C \\ \perp & \text{if } (v_1, \cdots, v_n) \notin C \end{cases}.$$

Intuitively, although the function computes an entry-by-entry hash, the requirement that $(v_1, \cdots, v_n)$ must be a codeword enforces the hardness of inversion when only classical queries are allowed. More precisely, the underlying code C satisfies a property called list-recoverability; even if a classical algorithm learns hash values of a subset $E_i \subseteq \Sigma$ for each of $H(i, \cdot)$, only a polynomial number of codewords can be found in $E_1 \times E_2 \times \cdots \times E_n$, which does not help invert random images[5]. On the other hand, they showed a quantum algorithm that uses quantum queries to invert images.

Liu [20] observed that the inversion quantum algorithm only needs to make non-adaptive quantum queries that are independent of the image y. Therefore, the algorithm with quantum access can be easily cast into a quantum algorithm with quantum advice but no queries; on the other hand, he showed that, if an algorithm has no access to the random oracle, it can not invert even with a piece of exponentially large classical advice. Since given an image y there are multiple pre-images of y, the above two statements lead to the separation between $\mathsf{FBQP/qpoly}$ and $\mathsf{FBQP/poly}$ when an algorithm has no access to the oracle.

Allowing classical queries

Our first result is to extend the previous separation of $\mathsf{FBQP/qpoly}$ and $\mathsf{FBQP/poly}$ by Liu, by allowing quantum algorithms to make online *classical queries* to H. Since the algorithm with quantum advice makes no queries, it also works in this setting of classical access. We only need to prove the hardness with classical advice: i.e., no quantum algorithms can invert with classical queries and bounded classical advice.

Following the framework by Guo et al. [17], when only making classical queries (say, at most T), a piece of S bits of classical advice is equivalent to the so-called "ordinary advice": the advice consists of only ST coordinates, or ST pairs of inputs and outputs. More precisely, the information the algorithm can learn from S bits of advice and T classical queries is roughly the same as that from ST bits of ordinary advice and T classical queries. Thus, the first step is to replace classical advice with ordinary classical advice.

It now remains to show that a quantum algorithm with classical access to H and short ordinary advice can not invert random images. As ordinary advice only gives information on at most polynomially many pairs of inputs and outputs, let E_i denote the subset of inputs

that there is a variant of [8] that satisfies the former (but not latter).

[5] When all classical queries are non-adaptive, this is clearly true: as only polynomially many f_c^H are known for codewords in C. The idea can be adapted to adaptive queries as well; we do not elaborate on it here.

whose hash values under $H(i, \cdot)$ are in the ordinary advice; since the ordinary advice is of length polynomial, $|E_i|$ is polynomial for each $i \in [n]$. Now let us assume the algorithm makes non-adaptive queries that are independent of a challenge. In this case, we can further define E'_i that consists of all x inputs whose values $H(i, x)$ are known from these classical queries. The algorithm in total learns hashes of $H(i, \cdot)$ for the inputs in $E_i \cup E'_i$. Observing that $|E_i \cup E'_i|$ is polynomial for each i, by the list-recoverability of the underlying code C, the algorithm only learns values of f^H_C for a polynomial number of codewords, which almost always never hits a random challenge.

Lastly, we extend the proof to adaptive cases (for more details, please refer to Section 3).

Upgrading to BQP/qpoly v.s. BQP/poly

Next, we turn the above separation of relational classes into a separation of BQP/qpoly v.s. BQP/poly. Our idea is to define a language $\mathcal{L}$ through a family of random functions for each n: $G_n : \{0,1\}^n \to \{0,1\}$, i.e., for $x \in \{0,1\}^n$, $x \in \mathcal{L}$ if and only if $G_n(x) = 1$. We omit the subscript when it is clear from context in the introduction.

We use an oracle $\mathcal{O}$ to hide the evaluation of G on x, by requiring the algorithm to invert the function f^H_C on image x. More precisely, we define $\mathcal{O}$ as follows:

$$\mathcal{O}(\mathbf{v}, x) = \begin{cases} G(x) & \text{if } f^H_C(\mathbf{v}) = x, \\ \perp & \text{otherwise.} \end{cases}$$

Then an algorithm only gets oracle access to $\mathcal{O}$, but not H or G.

To see that $\mathcal{L}$ is decidable by a BQP/qpoly machine, we can just take the quantum advice as in [20]. On an input x, an algorithm generates $\mathbf{v}$ such that $f^H_C(\mathbf{v}) = x$ using the quantum advice. We can then evaluate $G(x)$ by querying $\mathcal{O}$ at $(x, \mathbf{v})$ and decide if $x \in \mathcal{L}$.

On proving that $\mathcal{L}$ cannot be decided by any BQP/poly machine, we leverage the statement proved above: given classical advice, by only querying a classical oracle H, it is hard for an efficient algorithm to invert f^H_C. The beyond result implies that the algorithm should only have negligible query weight on $\mathbf{v}$ under the classical oracle $\mathcal{O}$ such that $f^H_C(\mathbf{v}) = x$; otherwise the algorithm can be turned into another one that inverts f^H_C.

Therefore, we can reduce the problem to the case where an algorithm has only access to $G(y)$ for all $y \neq x$, but no access to $G(x)$ for the challenge x; the goal is still to learn $G(x)$. This is exactly the famous Yao's box problem [27]: a piece of advice is allowed to depend on the whole oracle G, but then an online algorithm uses the advice to find $G(x)$ for a random x, with no access to $G(x)$. By adapting the ideas in [27, 14] and combining all the previous ideas with a standard diagonalization argument, we prove the separation $\mathsf{BQP}^{\mathcal{O}}/\mathsf{qpoly} \neq \mathsf{BQP}^{\mathcal{O}}/\mathsf{poly}$ relative to a classically accessible classical oracle.

Separation between QMA and QCMA relative to a classically accessible classical oracle

We first construct a problem that has a short quantum proof for YES instances, no quantum proof for NO instances, and no classical proof that can distinguish between YES and NO instances. Given random functions $H : [n] \times \Sigma \to \{0,1\}$ and $G : \{0,1\}^n \to \{0,1\}^n$, as well as a subset $S \subseteq \{0,1\}^n$ with a size of at least $2/3 \cdot 2^n$, we create two pairs of oracles:

- For a YES instance, the oracle G and an oracle $\mathcal{O}[G, H, \emptyset]$ are provided. The latter takes an input $t \in \{0,1\}^n$ and a vector $\mathbf{v} \in C$ and outputs 1 if and only if $f^H(\mathbf{v}) = G(t)$.

- If it is a NO instance, the oracle G and an oracle $\mathcal{O}[G, H, S]$ is given. The latter takes as input a $t \in \{0,1\}^n$ and a vector $\mathbf{v} \in C$, it outputs 1 if and only if $f^H(\mathbf{v}) = G(t)$ and $t \notin S$.

A quantum algorithm $\mathcal{A}$, with the same advice as in [20], achieves the following:

- The algorithm $\mathcal{A}$ on oracles $G, \mathcal{O}$ (which will be either $\mathcal{O}[G, H, \emptyset]$ or $\mathcal{O}[G, H, S]$), uniformly at random samples $t \in \{0,1\}^n$. It then uses the quantum advice to compute a vector $\mathbf{v}$ such that $f^H(\mathbf{v}) = G(t)$ and outputs $\mathcal{O}(t, \mathbf{v})$.
- When $\mathcal{O}[G, H, \emptyset]$ is given, by the correctness of the YZ algorithm, $\mathcal{A}$ will output 1 with an overwhelming probability.
- When $\mathcal{O}[G, H, S]$ is given, by the definition and the condition $|S| \geq 2/3 \cdot 2^n$, $\mathcal{A}$ will output 1 with a probability at most 1/3.

On the other hand, for any quantum algorithm $\mathcal{B}$ with classical queries and bounded-sized classical proof, it can not distinguish between the case of having access to $G, \mathcal{O}[G, H, \emptyset]$ or $G, \mathcal{O}[G, H, S]$. On a high level, the only way to tell the difference is by finding an input $(t, \mathbf{v})$ such that $\mathcal{O}[G, H, \emptyset](t, \mathbf{v}) \neq \mathcal{O}[G, H, S](t, \mathbf{v})$. This will require $\mathcal{B}$ to query on an input $(t, \mathbf{v})$ such that $f^H(\mathbf{v}) = G(t)$, which is difficult for $\mathcal{B}$ with only classical advice.

Finally, we mount our new separation result on the diagonalization argument and construct a language that shows the separation between QMA and QCMA relative to a classically accessible classical oracle. Please refer to the full version for more details.

Separation between QMA and QCMA relative to a distributional oracle

To separate QMA from QCMA, we would try to separate two distributions of oracles, namely the YES and NO distributions, defined below: For a random function $H : [n] \times \Sigma \to \{0,1\}$,

- If it is a YES instance, an oracle distribution $\{\mathcal{O}[r]\}_r$ with the index being drawn uniformly at random, such that: $\mathcal{O}[r]$ takes $\mathbf{v} \in \Sigma^n$ and evaluates $f_C^H(\mathbf{v})$, and outputs 1 iff $f_C^H(\mathbf{v}) = r$;
- If it is a NO instance, the oracle always outputs 0.

To see that the two distributions can be distinguished using a QMA machine, notice that we can take the quantum advice as in [20], and generate $\mathbf{v}$ such that $f_C^H(\mathbf{v}) = r$. By querying $\mathcal{O}$ at $\mathbf{v}$, we can distinguish whether the oracle belongs to the YES distribution or the NO distribution.

To prove the two distributions cannot be distinguished by any QCMA machine, we notice that the difference between YES and NO oracles is only on inputs $\mathbf{v}$ that $f_C^H(\mathbf{v}) = r$. Therefore, we reduce it to the hardness of finding $\mathbf{v}$ for random r such that $f_C^H(\mathbf{v}) = r$, even when given classical advice.

In the full version, we define unary languages $\mathcal{L}_i$ and their related oracle distributions for each n. By a standard diagonalization argument, we can argue that there exists some language $\mathcal{L}$ that is in $\mathsf{QMA}^{\mathcal{O}}$ but not in $\mathsf{QCMA}^{\mathcal{O}}$.

1.3 Concurrent Work

A concurrent work by Aaronson, Buhrman, and Kretschmer [3], among many results, proves FBQP/poly $\neq$ FBQP/qpoly *unconditionally* where FBQP/poly and FBQP/qpoly are the relational variants of BQP/poly and BQP/qpoly, respectively. The key insight behind the result is an observation that a variant of hidden matching problem [8] gives an exponential separation between classical and quantum one-way communication complexity with short input length for Bob. They essentially prove that any such a separation with efficient Bob can be used to prove $\mathsf{FBQP/poly} \neq \mathsf{FBQP/qpoly}$. We independently had an observation that [26] gives such a separation of classical and quantum one-way communication complexity with short input length for Bob (see the full version). By relying on their proof, which is fairly easy in hindsight, it seems possible to prove $\mathsf{FBQP/poly} \neq \mathsf{FBQP/qpoly}$ by using [26] instead of the hidden matching problem.

A crucial difference between the one-way communication variant of [26] and the hidden matching problem is that the hardness of the former with classical communication holds even if Bob can classically query Alice's input. Due to this difference, one cannot reprove $\mathsf{BQP/poly} \neq \mathsf{BQP/qpoly}$ and $\mathsf{QMA} \neq \mathsf{QCMA}$ relative to a classically-accessible classical oracle by using the hidden matching problem instead of [26]. On the other hand, it seems possible to prove $\mathsf{QMA} \neq \mathsf{QCMA}$ relative to a distributional quantumly-accessible classical oracle by using (a parallel repetition variant of) their variant of the hidden matching problem instead of [26].

2 Preliminaries

Basic notations

For a set X, $|X|$ denotes the cardinality of X. We write $x \leftarrow X$ to mean that x is uniformly taken from X. For a distribution $\mathcal{D}$ over classical strings, we write $x \leftarrow \mathcal{D}$ to mean that x is sampled from the distribution $\mathcal{D}$. For sets X and Y, $\mathsf{Func}(X, Y)$ denotes the set of all functions from X to Y. For a positive integer n, $[n]$ denotes the set $\{1, 2, ..., n\}$. QPT stands for "Quantum Polynomial-Time". We use poly to mean a polynomial and negl to mean a negligible function.

Oracle variations

In the literature on quantum computation, when we say that a quantum algorithm has oracle access to $f : X \to Y$, it usually means that it is given oracle access to an oracle that applies a unitary $|x\rangle |y\rangle \mapsto |x\rangle |y \oplus f(x)\rangle$. We refer to such a standard oracle as **quantumly-accessible classical oracles**. In this paper, we consider the following two types of non-standard oracles.

The first is **classically-accessible classical oracles**. A classically-accessible classical oracle for a classical function $f : X \to Y$ takes a *classical string* $x \in X$ as input and outputs $f(x)$. In other words, when an algorithm sends $\sum_{x,y} \alpha_{x,y} |x\rangle |y\rangle$ to the oracle, the oracle first *measures* the first register and then applies the unitary $|x\rangle |y\rangle \mapsto |x\rangle |y \oplus f(x)\rangle$. Note that classically-accessible classical oracles apply non-unitary operations.

The second is **distributional quantumly-accessible classical oracles**. They are specified by a distribution $\mathcal{F}$ over classical functions f rather than by a single function f. When we consider an algorithm that is given oracle access to a distributional quantumly-accessible classical oracle, it works as follows: At the beginning of an execution of the algorithm, a function f is chosen according to the distribution $\mathcal{F}$, and then the algorithm has access to a quantumly-accessible classical oracle that computes f. Note that f is sampled at the beginning and then the same f is used throughout the execution.

Complexity classes

We define the complexity classes which we consider in this paper. Specifically, we define $\mathsf{BQP/qpoly}$, $\mathsf{BQP/poly}$, QMA, and QCMA relative to classically-accessible classical oracles and QMA and QCMA relative to distributional quantumly-accessible classical oracles.

▶ **Definition 7** (BQP/qpoly and BQP/poly languages relative to classically-accessible classical oracles.). *Let $\mathcal{O}$ be a classically-accessible classical oracle. A language $\mathcal{L} \subseteq \{0,1\}^*$ belongs to* BQP/qpoly *relative to $\mathcal{O}$ if there is a QPT machine $\mathcal{A}$ and a polynomial-size family $\{|z_n\rangle\}_{n\in\mathbb{N}}$ of quantum advice such that for any $x \in \{0,1\}^*$,*

$$\Pr[\mathcal{A}^{\mathcal{O}}(x, |z_{|x|}\rangle) = \mathcal{L}(x)] \geq 2/3$$

where $\mathcal{L}(x) := 1$ if $x \in \mathcal{L}$ and otherwise $\mathcal{L}(x) := 0$.

BQP/poly *is defined similarly except that the advice is required to be classical.*

▶ **Definition 8** (QMA and QCMA languages relative to classically-accessible classical oracles.). *Let $\mathcal{O}$ be a classically-accessible classical oracle. A language $\mathcal{L} \subseteq \{0,1\}^*$ belongs to* QMA *relative to $\mathcal{O}$ if there is a QPT machine V with classical access to its oracle and a polynomial p such that the following hold:*

Completeness *For any $x \in \mathcal{L}$, there is a $p(|x|)$-qubit witness $|w\rangle$ such that*

$$\Pr[V^{\mathcal{O}}(x, |w\rangle) = 1] \geq 2/3.$$

Soundness *For any $x \notin \mathcal{L}$ and $p(|x|)$-qubit witness $|w\rangle$,*

$$\Pr[V^{\mathcal{O}}(x, |w\rangle) = 1] \leq 1/3.$$

QCMA *is defined similarly except that the witness is required to be classical.*

▶ **Definition 9** (QMA and QCMA languages relative to distributional quantumly-accessible classical oracles.). *Let $\mathcal{F}$ be a distributional quantumly-accessible classical oracle, i.e., it specifies a distribution over classical functions f. A language $\mathcal{L} \subseteq \{0,1\}^*$ belongs to* QMA *relative to $\mathcal{F}$ if there is a QPT machine V with quantum access to its oracle and a polynomial p such that the following hold:*

Completeness *For any $x \in \mathcal{L}$, there is a $p(|x|)$-qubit witness $|w\rangle$ such that*

$$\Pr_{f \leftarrow \mathcal{F}}[V^{f}(x, |w\rangle) = 1] \geq 2/3.$$

Soundness *For any $x \notin \mathcal{L}$ and $p(|x|)$-qubit witness $|w\rangle$,*

$$\Pr_{f \leftarrow \mathcal{F}}[V^{f}(x, |w\rangle) = 1] \leq 1/3.$$

QCMA *is defined similarly except that the witness is required to be classical.*

▶ Remark 10. Notice that the quantum/classical witness $|w\rangle / w$ can only depend on the distribution $\mathcal{F}$ rather than a specific oracle f.

Non-Uniformity in the ROM

Prior work has developed a number of tools to characterize the power of a non-uniform adversary in the random oracle model (ROM). Note that we are considering the ROM where we only allow classical access to random oracles unlike the quantum ROM [9]. This is sufficient for our purpose because we use these tools only for proving BQP/qpoly $\neq$ BQP/poly and QMA $\neq$ QCMA relative to *classically-accessible* classical oracles.

Similar to [20], we will be using the presampling technique, introduced by [25] and further developed by [13, 17]. First, we define games in the ROM.

▶ **Definition 11** (Games in the ROM). *A game $\mathcal{G}$ in the ROM is specified by three classical algorithms* $\mathsf{Samp}^H, \mathsf{Query}^H$, *and* Verify^H *where $H \leftarrow \mathsf{Func}(X, Y)$ for some sets X, Y:*

- $\mathsf{Samp}^H(r)$*: it is a deterministic algorithm that takes uniformly random coins* $r \in \mathcal{R}$ *as input, and outputs a challenge* CH*.* [6]
- $\mathsf{Query}^H(r,\cdot)$*: it is a deterministic classical algorithm that hardcodes the randomness* r *used to construct the challenge and provides the adversary's online queries.* [7]
- $\mathsf{Verify}^H(r,\mathsf{ans})$*: it is a deterministic algorithm that takes the same random coins for generating a challenge and an alleged answer* ans*, and outputs* b *indicating whether the game is won (*$b = 1$ *for winning).*

Let T_{Samp} *be the number of queries made by* Samp *and* T_{Verify} *be the number of queries made by* Verify*.*

For a fixed $H \in \mathsf{Func}(X,Y)$ *and a quantum algorithm* $\mathcal{A}$*, the game* $\mathcal{G}^H_{\mathcal{A}}$ *is executed as follows:*

- *A challenger* $\mathcal{C}$ *samples* $\mathsf{CH} \leftarrow \mathsf{Samp}^H(r)$ *using uniformly random coins* r*.*
- *A (uniform or non-uniform) quantum algorithm* $\mathcal{A}$*, that has classical oracle access to* $\mathsf{Query}^H(r,\cdot)$*, takes* CH *as input and outputs* ans*. We call* $\mathcal{A}$ *an online adversary/algorithm.*
- $b \leftarrow \mathsf{Verify}^H(r,\mathsf{ans})$ *is the game's outcome.*

▶ **Definition 12.** *We say that a game* $\mathcal{G}$ *in the ROM has security* $\delta(Z,Q) := \delta$ *if*

$$\max_{\mathcal{A}} \Pr_H[\mathcal{G}^H_{\mathcal{A}} = 1] \le \delta$$

where $H \leftarrow \mathsf{Func}(X,Y)$ *and* $\max$ *is taken over all* $\mathcal{A}$ *with* Z*-bit classical advice that makes* Q *classical queries.*

The presampling technique relates the probability of success of a non-uniform algorithm with classical queries to a random oracle with the success probability of a uniform algorithm in the P bit-fixing game, as defined below.

▶ **Definition 13** (Games in the P-BF-ROM)**.** *A game* $\mathcal{G}$ *in the* P*-BF-ROM is specified by two classical algorithms* Samp^H *and* Verify^H *that work similarly to those in Definition 11.*

For a fixed $H \in \mathsf{Func}(X,Y)$ *and a pair of algorithms* $(f,\mathcal{A})$*, the game* $\mathcal{G}^H_{f,\mathcal{A}}$ *is executed as follows:*

- **Offline Stage:** *Before a game starts, an offline algorithm* f *(having no input) generates a list* $\mathcal{L} = \{(x_i,y_i)\}_{i\in[P]} \in (X \times Y)^P$ *containing at most* P *input-output pairs (all* x_i*'s are distinct).*
- **Online Stage:** *The game is then executed with* $\mathcal{A}$ *that knows* $\mathcal{L}$ *and has oracle access to* H *similarly to Definition 11.* H *is a function drawn at random such that it satisfies* $\mathcal{L}$*.*

▶ **Definition 14.** *We say that a game* $\mathcal{G}$ *in the* P*-BF-ROM has security* $\nu(P,Q) := \nu$ *if*

$$\max_{f,\mathcal{A}} \Pr_H[\mathcal{G}^H_{f,\mathcal{A}} = 1] \le \nu$$

where $H \leftarrow \mathsf{Func}(X,Y)$ *and* $\max$ *is taken over all* f *that outputs* P *input-output pairs and* $\mathcal{A}$ *that makes* Q *classical queries.*

▶ **Theorem 15** ([20, Theorem A.1][8])**.** *Let* $\mathcal{G}$ *be any game with* $Q_{\mathsf{Samp}}, Q_{\mathsf{Verify}}$ *being the number of queries made by* Samp *and* Verify*. For any classical advice length* Z*, and number of online queries* Q*:*

---

[6] We note that the randomness r is separate from the randomness of H. r is used to sample the challenge.

[7] As an example, for most applications, $\mathsf{Query}^H(r,\cdot) = H(\cdot)$.

[8] Similar theorems with slightly worse bounds are presented in [13, Theorem 5 and 6] and [17, Theorem 3].

1. *For $P = Z(Q + Q_{\mathsf{Samp}} + Q_{\mathsf{Verify}})$, if $\mathcal{G}$ has security $\nu(P, Q)$ in the P-BF-ROM, then it has security $\delta(Z, Q) \leq 2 \cdot \nu(P, Q)$ against non-uniform unbounded-time algorithms with Z bits of classical advice and Q classical queries.*
2. *For any $P > 0$, if $\mathcal{G}$ has security $1/2 + \nu(P, Q)$ in the P-BF-ROM, then it has security $\delta(Z, Q) \leq 1/2 + \nu(P, Q) + Z(Q + Q_{\mathsf{Verify}} + Q_{\mathsf{Samp}})/P$ against non-uniform unbounded-time algorithms with Z bits of classical advice and Q classical queries.*

We can use the above correspondence to bound the success probability of a non-uniform quantum algorithm with classical queries to the Yao's Box game [27].

▶ **Lemma 16** (Yao's Box with Classical Queries [27, 14]). *Let $G : [N] \to \{0, 1\}$ be a random function. Let $\mathcal{A}$ be an unbounded-time algorithm, with Z bits of (classical) advice z_G and Q classical queries to G. The probability that $\mathcal{A}$ computes $G(x)$ without querying G at random index x is at most*

$$\Pr_x[\mathcal{A}^G(z_G, x) = G(x)] \leq \frac{1}{2} + 2\sqrt{\frac{Z(Q+1)}{N}}.$$

The above lemma was essentially proved in [14], but we offer here an alternative proof using the presampling technique of Theorem 15.

Proof of Lemma 16. We consider the bit-fixing game in the P-BF-ROM, where we fix the value of G at P positions in the offline phase. Since $\mathcal{A}$ is not allowed to query G at position x even in the P-BF-ROM (because it queries G via the original Query^G), the only way for $\mathcal{A}$ to successfully compute $G(x)$ is if x is included in the set of fixed positions during the offline phase. This happens with probability $\frac{P}{N}$ and thus

$$\nu(P, T) \leq \frac{P}{N}.$$

In this game $Q_{\mathsf{Samp}} = 0$, since the Sampler outputs a challenge $x \in [N]$ uniformly at random, without the need to perform any queries. The Verifier only needs to query G at position x, thus $Q_{\mathsf{Verify}} = 1$. The statement of Item 2 of Theorem 15 with $P = \sqrt{Z(Q+1)N}$ implies that the advantage of a non-uniform algorithm with advice z_G and Q queries is at most

$$\delta(Z, Q) \leq \frac{1}{2} + 2\sqrt{\frac{Z(Q+1)}{N}}.$$ ◀

3 Quantum vs Classical Advice for [YZ22]

We review the result of [26], which gives an NP-search problem that is easy for BQP machines but hard for BPP machines relative to a random oracle. Then we show that the problem is easy for BQP machines with quantum advice and no online query but hard for unbounded-time machine with polynomial-size classical advice and polynomially-many classical online queries relying on an observation of [20].

▶ **Definition 17** ([26]). *Let $C \subseteq \Sigma^n$ be a code over an alphabet Σ and $H : [n] \times \Sigma \to \{0, 1\}$ be a function. The following function is called a YZ function with respect to C and H:*

$$f_C^H : C \to \{0,1\}^n$$
$$f_C^H(v_1, v_2, ..., v_n) = H(1, v_1)||H(2, v_2)||...||H(n, v_n).$$

▶ Remark 18. When we refer to a code $C \subseteq \Sigma^n$, it actually means a family $\{C_n\}_{n\in\mathbb{N}}$ of codes $C_n \subseteq \Sigma_n^n$ over an alphabet Σ_n. We often omit the dependence of n for notational simplicity.

[26] shows that there is an error correcting code C, which satisfies a property called *list-recoverability*, such that f_C^H is easy to invert with quantum queries to H but hard to invert with classical queries to H where H is modeled as a random oracle. Liu [20] observed that the quantum inversion algorithm of [26] does not need to make adaptive queries, and having a polynomial-size quantum advice on H that is independent of the target r suffices. In addition, he proved that classical advice does not suffice for the task if no adaptive query is allowed. Combining the above, we have the following theorem.

▶ **Theorem 19** ([26, 20]). *There is a code $C \subseteq \Sigma^n$ satisfying the following:*

1. **(Easiness with Quantum Advice)** *There is a QPT algorithm $\mathcal{A}$ and a family of $\mathsf{poly}(n)$-qubit quantum advice $\{|z_H\rangle\}_H$ such that for any $r \in \{0,1\}^n$,*

$$\Pr_H[f_C^H(\mathcal{A}(|z_H\rangle, r)) = r)] \geq 1 - \mathsf{negl}(n)$$

where $H \leftarrow \mathsf{Func}([n] \times \Sigma, \{0,1\})$.

2. **(Hardness with Classical Advice)** *For any unbounded-time algorithm $\mathcal{B}$ and polynomial s, there is a negligible function μ such that for any family of $s(n)$-bit classical advice $\{z_H\}_H$,*

$$\Pr_{H,r}[f_C^H(\mathcal{B}(z_H, r)) = r] \leq \mu(n)$$

where $H \leftarrow \mathsf{Func}([n] \times \Sigma, \{0,1\})$ and $r \leftarrow \{0,1\}^n$.

Moreover, C satisfies (ζ, ℓ, L)-list-recoverability, i.e., for any subset $T_i \subseteq \Sigma$ such that $|T_i| \leq \ell$ for every $i \in [n]$,

$$|\{(v_1, ..., v_n) \in C : |\{i \in [n] : x_i \in T_i\}| \geq (1-\zeta)n\}| \leq L$$

where $\zeta = \Omega(1)$, $\ell = 2^{n^c}$, and $L = 2^{\tilde{O}(n^{c'})}$ for some constants $0 < c < c' < 1$.

We extend Theorem 19 to require that the hardness with classical advice holds even if $\mathcal{A}$ is given a classical access to H. This can be seen as a unification of [26] and [20].

▶ **Theorem 20.** *Let $C \subseteq \Sigma^n$ be the code in Theorem 19, the following holds:*

Hardness with Classical Advice and Classical Queries *For any unbounded-time algorithm $\mathcal{B}$ that makes polynomially many classical queries to H and polynomial s, there is a negligible function μ such that for any family of $s(n)$-bit classical advice $\{z_H\}_H$,*

$$\Pr_{H,r}[f_C^H(\mathcal{B}^H(z_H, r)) = r] \leq \mu(n)$$

where $H \leftarrow \mathsf{Func}([n] \times \Sigma, \{0,1\})$ and $r \leftarrow \{0,1\}^n$.

Proof. The proof is obtained by combining the proofs of [26] and [20]. We give a full proof for completeness. By Item 1 of Theorem 15, we only have to prove that the winning probability of the following game in the P-BF-ROM is $\mathsf{negl}(n)$ for any $P = \mathsf{poly}(n)$ and an unbounded-time algorithm $\mathcal{B}$ that makes $Q = \mathsf{poly}(n)$ online classical queries.

Offline Stage. $\mathcal{B}$ chooses list $L = \{x_k, y_k\}_{k\in[P]}$ of P input-output pairs where $x_k \in [n] \times \Sigma$ and $y_k \in \{0,1\}$ for each $k \in [P]$ and $x_k \neq x_{k'}$ for all $k \neq k'$. Then the random oracle $H \leftarrow \mathsf{Func}([n] \times \Sigma, \{0,1\})$ is uniformly chosen under the constraint that $H(x_k) = y_k$ for all $i \in [P]$.

Online Stage. $\mathcal{B}$ takes $r \leftarrow \{0,1\}^n$ as input, makes Q classical queries to H, and outputs $\mathbf{v}^* = (v_1^*, ..., v_n^*) \in \Sigma^n$.
Decision. $\mathcal{B}$ wins if $\mathbf{v}^* \in C$ and $f_C^H(\mathbf{v}^*) = r$.
For $i \in [n]$, let $T_i := \{v_i \in \Sigma : x_k = (i, v_i)$ for some $k \in [P]\}$. Let

$$\mathsf{Good} := \{(v_1, ..., v_n) \in C : |\{i \in [n] : x_i \in T_i\}| \geq (1-\zeta)n\}.$$

By the (ζ, ℓ, L)-list-recoverability and $|T_i| \leq P \leq 2^{n^c} \leq \ell$ (for sufficiently large n), we have $|\mathsf{Good}| \leq L$. We consider the following two cases:

Case 1: $\mathbf{v}^* \in$ Good. For each element $\mathbf{v} \in \mathsf{Good}$, $\Pr_{r \leftarrow \{0,1\}^n}[f_C^H(\mathbf{v}) = r] = 2^{-n}$. Thus, by the union bound, the probability that $\mathcal{B}$ wins is at most $|\mathsf{Good}| \cdot 2^{-n} = 2^{-\Omega(n)}$ by $|\mathsf{Good}| \leq L = 2^{\tilde{O}(n^{c'})}$ and $c' < 1$.

Case 2: $\mathbf{v}^* \notin$ Good. The analysis of this case is very similar to the proof of soundness in [26] and the following proof is partially taken verbatim from theirs. For each $i \in [n]$ and $j \in [Q]$, let $S_i^j \subseteq \Sigma$ be the set of elements v_i such that $\mathcal{B}$ ever queried (i, v_i) by the point when it has just made the j-th query. Let $\hat{S}_i^j := S_i^j \cup T_i$ for each $i \in [n]$ and $j \in \{0, 1, ..., Q\}$. Without loss of generality, we assume that $v_i^* \in \hat{S}_i^Q$ for all $i \in [n]$.[9] After the j-th query, we say that a codeword $\mathbf{v} = (v_1, ..., v_n) \in C$ is K-*queried* if there is a subset $I \in [n]$ such that $|I| = K$, $v_i \in \hat{S}_i^j$ for all $i \in I$, and $v_i \notin \hat{S}_i^j$ for all $i \notin I$. By the assumption that $\mathbf{v}^* \notin \mathsf{Good}$, $\mathbf{v}^*$ is K_0-queried for some $K_0 \leq \lceil (1-\zeta)n \rceil$ right after the offline stage. By the assumption that $v_i^* \in \hat{S}_i^Q$ for all $i \in [n]$, $\mathbf{v}^*$ is n-queried after the Q-th query. Since a K-queried codeword either becomes $(K+1)$-queried or remains K-queried by a single query, $\mathbf{v}^*$ must be K-queried at some point of the execution of $\mathcal{B}$ for all $K = K_0, K_0+1, ..., n$.

We consider the number of codewords that ever become K-queried for $K = \lceil (1-\zeta)n \rceil$. If $\mathbf{v} = (v_1, ..., v_n) \in C$ is $\lceil (1-\zeta)n \rceil$-queried at some point, the number of i such that $v_i \in \hat{S}_i^Q$ is at least $\lceil (1-\zeta)n \rceil$ since $\hat{S}_i^j \subseteq \hat{S}_i^Q$ for all i, j. We have $|\hat{S}_i^Q| \leq Q + P = \mathsf{poly}(n) < 2^{n^c}$ for sufficiently large n. On the other hand, C is (ζ, ℓ, L)-list recoverable for $\ell = 2^{n^c}$ and $L = 2^{\tilde{O}(n^{c'})}$. Thus, the number of codewords that ever become $\lceil (1-\zeta)n \rceil$-queried is at most $L = 2^{\tilde{O}(n^{c'})}$.

Suppose that we simulate the oracle H for $\mathcal{B}$ via lazy sampling, that is, instead of uniformly choosing random functions at first, we sample function values whenever they are specified in the list sent in the offline stage or queried in the online stage. Suppose that a codeword $\mathbf{v}$ becomes $\lceil (1-\zeta)n \rceil$-queried at some point of the execution of the experiment. Since the function values on the unqueried $\lfloor \zeta n \rfloor$ positions are not sampled yet, $\mathbf{v}$ can become a valid proof only if all those values happen to be consistent to r, which occurs with probability $\left(\frac{1}{2}\right)^{\lfloor \zeta n \rfloor} = 2^{-\Omega(n)}$ by $\zeta = \Omega(1)$. Since one of them is the final output $\mathbf{v}^*$, by the union bound, the probability that $\mathbf{v}^*$ is a valid proof is at most $L \cdot \left(\frac{1}{2}\right)^{\lfloor \zeta n \rfloor} = 2^{-\Omega(n)}$ by $L = 2^{\tilde{O}(n^{c'})}$ and $c' < 1$. ◀

We have the following corollary. The motivation of showing this corollary is to ensure that "a large fraction of H works for all r" rather than that "for all r, a large fraction of H works". Looking ahead, this is needed for proving an oracle separation for $\mathsf{BQP/poly}$ and $\mathsf{BQP/qpoly}$ (but not for QMA and QCMA).

[9] This might increase Q by at most n, but T is still $\mathsf{poly}(n)$ anyway.

▶ **Corollary 21.** *Let C be the code in Theorem 19. Then the following hold:*

1. **(Easiness with Quantum Advice)** *There is a QPT algorithm $\mathcal{A}$ and a family of* $\mathsf{poly}(n)$*-qubit quantum advice $\{|z_{\mathcal{H}}\rangle\}_{\mathcal{H}}$ such that*

$$\Pr_{\mathcal{H}}[\forall r \in \{0,1\}^n \ \Pr[\exists j \in [n] \ s.t.\ f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = r : (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \leftarrow \mathcal{A}(|z_{\mathcal{H}}\rangle, r)] \geq 1 - \mathsf{negl}(n)]$$
$$\geq 1 - \mathsf{negl}(n)$$

where $\mathcal{H} = (H^{(1)}, ..., H^{(n)}) \leftarrow (\mathsf{Func}([n] \times \Sigma, \{0,1\}))^n$.

2. **(Hardness with Classical Advice and Classical Queries)** *For any unbounded-time algorithm $\mathcal{B}$ that makes* $\mathsf{poly}(n)$ *classical queries to $\mathcal{H} = (H^{(1)}, ..., H^{(n)})$ and polynomial s, there is a negligible function μ such that for any family of $s(n)$-bit classical advice $\{z_{\mathcal{H}}\}_{\mathcal{H}}$,*

$$\Pr_{\mathcal{H},r}[\exists j \in [n] \ s.t.\ f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = r : (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \leftarrow \mathcal{B}^{\mathcal{H}}(z_{\mathcal{H}}, r)] \leq \mu(n)$$

where $\mathcal{H} = (H^{(1)}, ..., H^{(n)}) \leftarrow (\mathsf{Func}([n] \times \Sigma, \{0,1\}))^n$ and $r \leftarrow \{0,1\}^n$.

Proof. For proving Item 1, we can set the advice as $|z_{\mathcal{H}}\rangle = |z_{H^{(1)}}\rangle \otimes |z_{H^{(2)}}\rangle \otimes \cdots \otimes |z_{H^{(n)}}\rangle$, and the algorithm just parallel runs the algorithm in Theorem 19 for different $H^{(i)}$. Assume the algorithm in Theorem 19 satisfies:

$$\Pr_{H}[f_C^H(\mathcal{A}(|z_H\rangle, r)) = r)] \geq 1 - \eta(n),$$

for some negligible function $\eta(n)$. For any $r \in \{0,1\}^n$,

$$\Pr_{\mathcal{H}}[\forall j \in [n], f_C^{H^{(j)}}(\mathbf{v}^{(j)}) \neq r : (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \leftarrow \mathcal{A}(|z_{\mathcal{H}}\rangle, r)] \leq \eta(n)^n.$$

By an averaging argument, at most $\eta(n)^{n/2}$ fraction of $\mathcal{H}$ will satisfy

$$\Pr[\forall j \in [n], f_C^{H^{(j)}}(\mathbf{v}^{(j)}) \neq r : (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \leftarrow \mathcal{A}(|z_{\mathcal{H}}\rangle, r)] \geq \eta(n)^{n/2}$$

By applying a union bound over all $r \in \{0,1\}^n$, we obtain that

$$\Pr_{\mathcal{H}}[\exists r \in \{0,1\}^n \Pr[\forall j \in [n], f_C^{H^{(j)}}(\mathbf{v}^{(j)}) \neq r : (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \leftarrow \mathcal{A}(|z_{\mathcal{H}}\rangle, r)] \geq \eta(n)] \leq (4\eta(n))^{\frac{n}{2}}$$

Applying negation, we obtain the bound above.

For proving Item 2, we will show how an adversary $\mathcal{B}$ that breaks hardness with classical advice and classical queries, can be used to construct an adversary $\mathcal{B}'$ that breaks hardness with classical advice and classical queries of Theorem 20. To show this, we go through the following steps:

1. For each fixed $j \in [n], \mathcal{H}_{\bar{j}} = (H^{(1)}, \ldots, H^{(j-1)}, H^{(j+1)}, \ldots, H^{(n)})$, we define a pair $(\mathcal{B}'[j, \mathcal{H}_{\bar{j}}], \{z'_H[j, \mathcal{H}_{\bar{j}}]\}_H)$ of an adversary and advice in which j and $\mathcal{H}_{\bar{j}}$ is hardwired.
2. Show that $(\mathcal{B}'[j, \mathcal{H}_{\bar{j}}], \{z'_H[j, \mathcal{H}_{\bar{j}}]\}_H)$ breaks Theorem 20 on average over the choice of j.
3. Fix the "best" j and $\mathcal{H}_{\bar{j}}$ (w.r.t. random H) to get a fixed pair of algorithm and advice that breaks hardness of Theorem 20.

Specifically, it works as follows. Suppose there exists some adversary $\mathcal{B}$ and some polynomial $q(n)$ such that for sufficiently large n,

$$\Pr_{\mathcal{H},r}[\exists j \in [n] \text{ s.t. } f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = r : (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \leftarrow \mathcal{B}^{\mathcal{H}}(z_{\mathcal{H}}, r)] \geq q(n)$$

For each j and $\mathcal{H}_{\bar{j}}$, we define $(\mathcal{B}'[j, \mathcal{H}_{\bar{j}}], \{z'_H[j, \mathcal{H}_{\bar{j}}]\}_H)$ as follows:

$z'_H[j, \mathcal{H}_{\bar{j}}]$: Given $\mathcal{H}_{\bar{j}}$, set

$$\mathcal{H} = (H^{(1)}, \ldots, H^{(j-1)}, H, H^{(j+1)}, \ldots, H^{(n)}).$$

Set $z'_H[j, \mathcal{H}_{\bar{j}}]$ to the advice of $\mathcal{B}$ for $\mathcal{H}$, i.e. $z'_H[j, \mathcal{H}_{\bar{j}}] := z_{\mathcal{H}}$.

$\mathcal{B}'[j,\mathcal{H}_{\bar{j}}]^H(z'_H[j,\mathcal{H}_{\bar{j}}],r)$: It runs $\mathcal{B}^{\mathcal{H}}(z'_H[j,\mathcal{H}_{\bar{j}}],r)$ where $\mathcal{B}'$ hardwired $\mathcal{H}_{\bar{j}}$ into its algorithm. It simulates the oracle $\mathcal{H}$ for $\mathcal{B}$ by querying its own oracle H as $H^{(j)}$ and simulates other $H^{(i)}(i \neq j)$ by querying the hardwired $\mathcal{H}_{\bar{j}}$.

By the above argument, we have

$$\begin{aligned}
&\Pr_{H,r,j,\mathcal{H}_{\bar{j}}}[f_C^H(\mathcal{B}'[j,\mathcal{H}_{\bar{j}}]^H(z'_H[j,\mathcal{H}_{\bar{j}}],r)) = r] \\
&= \Pr_{\mathcal{H},r,j}[f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = x : (\mathbf{v}^{(1)},...,\mathbf{v}^{(n)}) \leftarrow \mathcal{B}^{\mathcal{H}}(z_{\mathcal{H}},r)] \\
&\geq \frac{q(n)}{n}.
\end{aligned}$$

Thus, by taking $j = j^*$ and $\mathcal{H}^*_{\bar{j}^*}$ that makes the above probability the largest for random $\mathcal{H}$, $(\mathcal{B}',\{z'_H\}_H) := (\mathcal{B}'[j^*,\mathcal{H}^*_{\bar{j}^*}],\{z'_H[j^*,\mathcal{H}^*_{\bar{j}^*}]\}_H)$ breaks hardness of Theorem 20. ◀

4 BQP/poly vs BQP/qpoly under Classically-Accessible Oracle

In this section, we demonstrate a $\mathsf{BQP/qpoly}$ and $\mathsf{BQP/poly}$ separation relative to a classically-accessible classical oracle.

The main technical lemma needed for proving the separation is the following.

▶ **Lemma 22.** *There is a family of distributions $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, where $\mathcal{D}_n$ is supported on tuples $(G,\mathcal{O})$ of functions $G : \{0,1\}^n \to \{0,1\}$ and $\mathcal{O} : \{0,1\}^{p(n)} \to \{0,1\}^{q(n)}$ for some polynomials p and q, satisfying the following:*

1. **(Easiness with Quantum Advice)** *There is a QPT algorithm $\mathcal{A}$ with classical access to $\mathcal{O}$ and a family of $\mathsf{poly}(n)$-qubit quantum advice $\{|z_{\mathcal{O}}\rangle\}_{\mathcal{O}}$ such that*

$$\Pr_{(G,\mathcal{O})\leftarrow\mathcal{D}_n}[\forall x \in \{0,1\}^n \;\; \Pr[\mathcal{A}^{\mathcal{O}}(|z_{\mathcal{O}}\rangle, x) = G(x)] \geq 1 - \mathsf{negl}(n)] \geq 1 - \mathsf{negl}(n).$$

2. **(Hardness with Classical Advice)** *For any unbounded-time algorithm $\mathcal{B}$ that makes $\mathsf{poly}(n)$ classical queries to $\mathcal{O}$ and a family of $\mathsf{poly}(n)$-bit classical advice $\{z_{\mathcal{O}}\}_{\mathcal{O}}$,*

$$\Pr_{\substack{(G,\mathcal{O})\leftarrow\mathcal{D}_n \\ x\leftarrow\{0,1\}^n}}[\mathcal{B}^{\mathcal{O}}(z_{\mathcal{O}},x) = G(x)] \leq \frac{3}{5}$$

for all sufficiently large n.

For proving Lemma 22, we prepare the following lemma.

▶ **Lemma 23.** *Let $G : \{0,1\}^n \to \{0,1\}$ be a uniformly random function. For an unbounded-time algorithm $\mathcal{A}$ that makes $\mathsf{poly}(n)$ classical queries to G and a family of $\mathsf{poly}(n)$-bit classical advice $\{z_G\}_G$, suppose that the following holds:*

$$\Pr_{G,x\leftarrow\{0,1\}^n}[\mathcal{A}^G(z_G,x) = G(x)] > \frac{3}{5}.$$

Then the probability that x is contained in the query list is at least $\frac{1}{20}$ for $\frac{1}{30}$ fraction of $x \in \{0,1\}^n$ for sufficiently large n.

Proof. For each $x \in \{0,1\}^n$, we define G_x as the random function G with its input on x removed, i.e. $G_x(x') = G(x')$ for $x' \neq x$ and $G_x(x) = 0$. Since $\mathcal{A}$ only makes classical queries to G, the only way for it to distinguish G from G_x is to query the oracle at x. Denote by δ_x the probability that x is in the query list of $\mathcal{A}^{G_x}$, where the probability is over the randomness of $\mathcal{A}$ and G_x. We obtain that,

$$|\Pr_G[\mathcal{A}^G(z_G, x) = G(x)] - \Pr_G[\mathcal{A}^{G_x}(z_G, x) = G(x)]| \leq \delta_x.$$

Now we consider the case when we uniform randomly choose $x \leftarrow \{0,1\}^n$, and require $\mathcal{A}^{G_x}(z_G, x)$ to output $G(x)$. This is exactly Yao's box problem, where the adversary is required to output $G(x)$ without querying x. By Lemma 16, we have the following bound for Yao's box with classical queries and classical advice:

$$\Pr_{G,x}[\mathcal{A}^{G_x}(z_G, x) = G(x)] \leq \frac{1}{2} + 2\sqrt{\frac{|z_G|(Q+1)}{2^n}} = \frac{1}{2} + \mathsf{negl}(n)$$

where we assume that $\mathcal{A}$ makes Q queries. Thus we have that

$$\Pr_{G,x}[\mathcal{A}^G(z_G, x) = G(x)] - \Pr_{G,x}[\mathcal{A}^{G_x}(z_G, x) = G(x)] \geq \frac{1}{10} - \mathsf{negl}(n),$$

Therefore, we have

$$\mathop{\mathbb{E}}_x[\delta_x] \geq \mathop{\mathbb{E}}_x\left[\left|\Pr_G[\mathcal{A}^G(z_G, x) = G(x)] - \Pr_G[\mathcal{A}^{G_x}(z_G, x) = G(x)]\right|\right] \geq \frac{1}{10} - \mathsf{negl}(n)$$

We now show that δ_x is at least $\frac{1}{20}$ with probability $\frac{1}{30}$ for sufficiently large n.

$$\Pr_x\left[\delta_x \geq \frac{1}{20}\right] + \left(1 - \Pr_x\left[\delta_x \geq \frac{1}{20}\right]\right) \cdot \frac{1}{20} \geq \mathbb{E}_x[\delta_x] \geq \frac{1}{10} - \mathsf{negl}(n)$$

$$\implies \Pr_x\left[\delta_x \geq \frac{1}{20}\right] \geq \frac{1}{20} - \mathsf{negl}(n)$$

Thus for sufficiently large n, for a $\frac{1}{20} - \mathsf{negl}(n) \geq \frac{1}{30}$ fraction of $x \in \{0,1\}^n$, $\mathcal{A}$ will query x with probability at least $\frac{1}{20}$. ◀

Then we prove Lemma 22.

Proof of Lemma 22. We define $\mathcal{D}_n$ to be the distribution that samples G and $\mathcal{O}$ as follows:

$\mathcal{D}_n$: Let $C \subseteq \Sigma^n$ be the code in Corollary 21. It samples random functions $G : \{0,1\}^n \to \{0,1\}$ and $H^{(j)} : \{0,1\}^{\log n} \times \Sigma \to \{0,1\}$ for $j \in [n]$ and defines $\mathcal{O}$ as follows: $\mathcal{O}$ takes $x \in \{0,1\}^n$ and $(\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \in C^n$ as input, and outputs $G(x)$ if there is $j \in [n]$ such that $f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = x$ and outputs $\perp$ otherwise.[10] For simplicity we will denote by $\mathcal{H} = (H^{(1)}, \ldots, H^{(n)})$.

First, we show the easiness with quantum advice (Item 1). Let $(\mathcal{A}', \{|z'_{\mathcal{H}}\rangle\}_{\mathcal{H}})$ be the tuple of an algorithm and family of quantum advice in Item 1 of Corollary 21. We construct $(\mathcal{A}, \{|z_{\mathcal{O}}\rangle\}_{\mathcal{O}})$ that satisfies Item 1 of Lemma 22.

In fact, we allow the advice $|z_{\mathcal{O}}\rangle$ to be a mixed state and write it by $\rho_{\mathcal{O}}$. Note that this does not weaken the statement since any mixed state can be considered as a distribution over pure states and thus there must exist a pure state advice $|z_{\mathcal{O}}\rangle$ that is at least as good as $\rho_{\mathcal{O}}$. The algorithm $\mathcal{A}$ and quantum advice $\rho_{\mathcal{O}}$ is described as follows:

[10] Note that x here plays the role of r in Corollary 21.

$\rho_{\mathcal{O}}$: We describe a randomized procedure to set $\rho_{\mathcal{O}}$ given an oracle $\mathcal{O}$. This should be understood as setting $\rho_{\mathcal{O}}$ to be the mixed state corresponding to the output of the procedure. Sample $(G, \mathcal{H})$ from the conditional distribution of $\mathcal{D}_n$ conditioned on the given $\mathcal{O}$. Note that then the joint distribution of $(G, \mathcal{H}, \mathcal{O})$ is identical to the real one. Then $\rho_{\mathcal{O}}$ is set to be $\rho_{\mathcal{H}}$.

$\mathcal{A}^{\mathcal{O}}(\rho_{\mathcal{O}}, x)$: It runs $\mathbf{v} \leftarrow \mathcal{A}'(\rho_{\mathcal{O}}, x)$, queries $(x, \mathbf{v})$ to $\mathcal{O}$, and outputs whatever $\mathcal{O}$ returns.

Then Item 1 of Corollary 21 implies

$$\Pr_{(G,\mathcal{O})\leftarrow\mathcal{D}_n}[\forall x \in \{0,1\}^n \ \Pr[\mathcal{A}^{\mathcal{O}}(\rho_{\mathcal{O}}, x) = G(x)] \geq 1 - \mathsf{negl}(n)] \geq 1 - \mathsf{negl}(n).$$

Thus, Item 1 of Lemma 22 holds.

Next, we show the hardness with classical advice (Item 2). Suppose that there is $(\mathcal{B}, \{z_{\mathcal{O}}\}_{\mathcal{O}})$ that breaks it. Then we have

$$\Pr_{\substack{(G,\mathcal{O})\leftarrow\mathcal{D}_n \\ x\leftarrow\{0,1\}^n}}[\mathcal{B}^{\mathcal{O}}(z_{\mathcal{O}}, x) = G(x)] > \frac{3}{5}$$

for infinitely many $n \in \mathbb{N}$. Recall that $\mathcal{O}$ returns $G(x)$ only if the query $(x, (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}))$ satisfies $f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = x$ for some $j \in [n]$. Thus, by a direct reduction to Lemma 23, for a $\frac{1}{30}$ fraction of $x \in \{0,1\}^n$, the query list of $\mathcal{B}$ to a randomly chosen $\mathcal{O}$ according to $\mathcal{D}_n$ will contain a query of the form $(x, (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}))$ such that there is $j \in [n]$ such that $f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = x$ with probability at least $\frac{1}{20}$.

Also note that classical access to $\mathcal{O}$ can be simulated by classical access to G and $\mathcal{H} = (H^{(1)}, \ldots, H^{(n)})$. Thus, the above directly gives an algorithm that violates the hardness with classical advice and classical access to $\mathcal{H}$ (Item 2 of Corollary 21). To show this, we go through the following steps:

1. For each fixed G, we define a pair $(\mathcal{B}'[G], \{z'_{\mathcal{H}}[G]\}_{\mathcal{H}})$ of an adversary and advice in which G is hardwired.
2. Show that $(\mathcal{B}'[G], \{z'_{\mathcal{H}}[G]\}_{\mathcal{H}})$ breaks Item 2 of Corollary 21 on average over the choice of G.
3. Fix the "best" G (w.r.t. random $\mathcal{H}$) to get a fixed pair of algorithm and advice that breaks Item 2 of Corollary 21.

Specifically, it works as follows. For each G, we define $(\mathcal{B}'[G], \{z'_{\mathcal{H}}[G]\}_{\mathcal{H}})$ as follows:

$z'_{\mathcal{H}}[G]$: Construct $\mathcal{O}$ from $(G, \mathcal{H})$. Set $z'_{\mathcal{H}}[G] := z_{\mathcal{O}}$.

$\mathcal{B}'[G]^{\mathcal{H}}(z'_{\mathcal{H}}[G], x)$: It runs $\mathcal{B}^{\mathcal{O}}(z'_{\mathcal{H}}[G], x)$ where $\mathcal{B}'$ simulates the oracle $\mathcal{O}$ for $\mathcal{B}$ by using its own oracle $\mathcal{H}$ and the hardwired oracle G and outputs a uniformly chosen query by $\mathcal{B}$.

By the above argument, we have

$$\Pr_{G,\mathcal{H},x}[\exists j \in [n] \text{ s.t. } f_C^{H^{(j)}}(\mathbf{v}^{(j)}) = x : (\mathbf{v}^{(1)}, ..., \mathbf{v}^{(n)}) \leftarrow \mathcal{B}'[G]^{\mathcal{H}}(z'_{\mathcal{H}}[G], x)] \geq \frac{1}{600Q}$$

where Q is the number of queries by $\mathcal{B}$. Thus, by taking $G = G^*$ that makes the above probability the largest, $(\mathcal{B}', \{z'_{\mathcal{H}}\}_{\mathcal{H}}) := (\mathcal{B}'[G^*], \{z'_{\mathcal{H}}[G^*]\}_{\mathcal{H}})$ breaks Item 2 of Corollary 21.[11] ◀

[11] At first glance, this argument seems to allow $\mathcal{B}'$ to be a non-uniform machine that takes G as advice. However, this is not needed since $\mathcal{B}'$ can find the best G by itself by using its unbounded-time computational power.

Given Lemma 22, it is straightforward to prove a separation between $\mathsf{BQP}/\mathsf{qpoly}$ and $\mathsf{BQP}/\mathsf{poly}$ relative to a classically-accessible classical oracle by the standard diagonalization argument.

▶ **Theorem 24.** *There is a classically-accessible classical oracle $\mathcal{O}$ relative to which* $\mathsf{BQP}/\mathsf{poly} \neq \mathsf{BQP}/\mathsf{qpoly}$.

Proof. Suppose that we generate $(G, \mathcal{O}) \leftarrow \mathcal{D}_n$ and define a language $\mathcal{L} := \bigsqcup_{n\in\mathbb{N}} G_n^{-1}(1)$ and an oracle $\mathcal{O}$ that returns $\mathcal{O}_{|x|}(x)$ on a query $x \in \{0,1\}^*$. We claim that $\mathcal{L} \in \mathsf{BQP}^{\mathcal{O}}/\mathsf{qpoly}$ and $\mathcal{L} \notin \mathsf{BQP}^{\mathcal{O}}/\mathsf{poly}$ with probability 1.

To see $\mathcal{L} \in \mathsf{BQP}^{\mathcal{O}}/\mathsf{qpoly}$ with probability 1, Item 1 of Lemma 22 implies that there is a BQP machine $\mathcal{A}$ with polynomial-size quantum advice that decides $\mathcal{L}$ on all x of length n with probability at least $1 - \frac{1}{n^2}$ for all sufficiently large n. Since $\sum_{n=1}^{\infty} \frac{1}{n^2} = \pi^2/6$ is finite, the Borel–Cantelli lemma implies that the probability that $\mathcal{A}$ fails on infinitely many n is 0. In other words, the probability that there is N such that $\mathcal{A}$ succeeds in deciding $\mathcal{L}$ on all x such that $|x| \geq N$ is 1. By augmenting $\mathcal{A}$ to decide $\mathcal{L}$ by brute-force when the instance has length smaller than N, we can conclude that there is a BQP machine with polynomial-size quantum advice that decides $\mathcal{L}$ on all $x \in \{0,1\}^*$ with probability 1 over the random choice of $(G, \mathcal{O})$ for $n \in \mathbb{N}$.

Next, we prove $\mathcal{L} \notin \mathsf{BQP}^{\mathcal{O}}/\mathsf{poly}$ with probability 1. For a BQP machine $\mathcal{B}$ that takes $\ell(n)$-bit classical advice for a polynomial ℓ, we define $S_{\mathcal{B}}(n)$ to be the event over the choice of $(G, \mathcal{O})$ that there is a $\ell(n)$-bit classical advice $z_{\mathcal{O}}$ such that

$$\Pr[\forall x \in \{0,1\}^n \ \mathcal{B}^{\mathcal{O}}(z_{\mathcal{O}}, x) = G(x)] \geq \frac{2}{3}.$$

Item 2 of Lemma 22 implies that there is an integer N such that for any BQP machine $\mathcal{B}$ with classical access to $\mathcal{O}$, we have $\Pr_{G,\mathcal{O}}[S_{\mathcal{B}}(n)] \leq c$ for all $n \geq N$ where $c := 9/10$. We now show that

$$\Pr_{G,\mathcal{O}}[S_{\mathcal{B}}(1) \wedge S_{\mathcal{B}}(2) \wedge \dots] = 0$$

- We will consider a sequence of input lengths $n_1, n_2, \dots$ defined by $n_1 := N$ and $n_i := T(n_{i-1}) + 1$, where $T(n)$ is the running time of $\mathcal{B}$ on input of length n. This means that when $\mathcal{B}$'s input length is n_{i-1}, it cannot touch the oracle on input length $\geq n_i$. This guarantees that $\Pr[S_{\mathcal{B}}(n_i) \mid S_{\mathcal{B}}(n_j)] = \Pr[S_{\mathcal{B}}(n_i)]$ for all $i > j$.
- We can now show that the probability that $\mathcal{B}$ succeeds on all inputs is equal to 0 over the choices of $G, \mathcal{O}$.

$$\begin{aligned}
&\Pr[S_{\mathcal{B}}(1) \wedge S_{\mathcal{B}}(2) \wedge \dots] \\
&\leq \Pr\left[\bigwedge_i S_{\mathcal{B}}(n_i)\right] \\
&= \Pr[S_{\mathcal{B}}(n_1)] \cdot \Pr[S_{\mathcal{B}}(n_2) \mid S_{\mathcal{B}}(n_1)] \cdot \dots \\
&\leq c \cdot c \cdot \dots \\
&= 0
\end{aligned}$$

Since there are countably many QPT machines,

$$\Pr_{G,\mathcal{O}}[\exists \mathcal{B} \ S_{\mathcal{B}}(1) \wedge S_{\mathcal{B}}(2) \wedge \dots] = 0.$$

This means that $\mathcal{L} \notin \mathsf{BQP}^{\mathcal{O}}/\mathsf{poly}$ with probability 1 over the choice of $(G, \mathcal{O})$. ◀

References

1 Scott Aaronson. Limitations of quantum advice and one-way communication. *Theory Comput.*, 1(1):1–28, 2005. doi:10.4086/toc.2005.v001a001.

2 Scott Aaronson. The learnability of quantum states. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 463(2088):3089–3114, September 2007. doi:10.1098/rspa.2007.0113.

3 Scott Aaronson, Harry Buhrman, and William Kretschmer. A qubit, a coin, and an advice string walk into a relational problem, 2023. doi:10.48550/arXiv.2302.10332.

4 Scott Aaronson and Andrew Drucker. A full characterization of quantum advice. *SIAM J. Comput.*, 43(3):1131–1183, 2014. doi:10.1137/110856939.

5 Scott Aaronson and Greg Kuperberg. Quantum versus classical proofs and advice. In *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC'07)*, pages 115–128. IEEE, 2007.

6 Dorit Aharonov and Tomer Naveh. Quantum np - a survey. *CoRR*, 2002. doi:10.48550/arXiv.quant-ph/0210077.

7 Andris Ambainis, Leonard J. Schulman, Amnon Ta-Shma, Umesh V. Vazirani, and Avi Wigderson. The quantum communication complexity of sampling. *SIAM J. Comput.*, 32(6):1570–1585, 2003. doi:10.1137/S009753979935476.

8 Ziv Bar-Yossef, T. S. Jayram, and Iordanis Kerenidis. Exponential separation of quantum and classical one-way communication complexity. In László Babai, editor, *36th ACM STOC*, pages 128–137. ACM Press, June 2004. doi:10.1145/1007352.1007379.

9 Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Heidelberg, December 2011. doi:10.1007/978-3-642-25385-0_3.

10 Harry Buhrman, Richard Cleve, and Avi Wigderson. Quantum vs. classical communication and computation. In Jeffrey Scott Vitter, editor, *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*, pages 63–68. ACM, 1998. doi:10.1145/276698.276713.

11 Kai-Min Chung, Siyao Guo, Qipeng Liu, and Luowen Qian. Tight quantum time-space tradeoffs for function inversion. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 673–684. IEEE, 2020.

12 Kai-Min Chung, Tai-Ning Liao, and Luowen Qian. Lower bounds for function inversion with quantum advice. *arXiv preprint*, 2019. arXiv:1911.09176.

13 Sandro Coretti, Yevgeniy Dodis, Siyao Guo, and John P. Steinberger. Random oracles and non-uniformity. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 227–258. Springer, Heidelberg, April / May 2018. doi:10.1007/978-3-319-78381-9_9.

14 Anindya De, Luca Trevisan, and Madhur Tulsiani. Time space tradeoffs for attacks against one-way functions and PRGs. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 649–665. Springer, Heidelberg, August 2010. doi:10.1007/978-3-642-14623-7_35.

15 Bill Fefferman and Shelby Kimmel. Quantum vs. classical proofs and subset verification. In Igor Potapov, Paul G. Spirakis, and James Worrell, editors, *43rd International Symposium on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK*, volume 117 of *LIPIcs*, pages 22:1–22:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.MFCS.2018.22.

16 Dmitry Gavinsky. Classical interaction cannot replace a quantum message. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 95–102, 2008.

17 Siyao Guo, Qian Li, Qipeng Liu, and Jiapeng Zhang. Unifying presampling via concentration bounds. In Kobbi Nissim and Brent Waters, editors, *TCC 2021, Part I*, volume 13042 of *LNCS*, pages 177–208. Springer, Heidelberg, November 2021. doi:10.1007/978-3-030-90459-3_7.

18 Martin Hellman. A cryptanalytic time-memory trade-off. *IEEE transactions on Information Theory*, 26(4):401–406, 1980.

19 Minki Hhan, Keita Xagawa, and Takashi Yamakawa. Quantum random oracle model with auxiliary input. In *Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8–12, 2019, Proceedings, Part I*, pages 584–614. Springer, 2019.

20 Qipeng Liu. Non-uniformity and quantum advice in the quantum random oracle model. *CoRR*, 2022. `doi:10.48550/arXiv.2210.06693`.

21 Anand Natarajan and Chinmay Nirkhe. A classical oracle separation between qma and qcma. *CoRR*, 2022. `doi:10.48550/arXiv.2210.15380`.

22 Aran Nayebi, Scott Aaronson, Aleksandrs Belovs, and Luca Trevisan. Quantum lower bound for inverting a permutation with advice. *arXiv preprint*, 2014. `arXiv:1408.3193`.

23 Harumichi Nishimura and Tomoyuki Yamakami. Polynomial time quantum computation with advice. *Inf. Process. Lett.*, 90(4):195–204, 2004. `doi:10.1016/j.ipl.2004.02.005`.

24 Ran Raz. Exponential separation of quantum and classical communication complexity. In *31st ACM STOC*, pages 358–367. ACM Press, May 1999. `doi:10.1145/301250.301343`.

25 Dominique Unruh. Random oracles and auxiliary input. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 205–223. Springer, Heidelberg, August 2007. `doi:10.1007/978-3-540-74143-5_12`.

26 Takashi Yamakawa and Mark Zhandry. Verifiable quantum advantage without structure. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 - November 3, 2022*, pages 69–74. IEEE, 2022. `doi:10.1109/FOCS54457.2022.00014`.

27 Andrew Chi-Chih Yao. Coherent functions and program checkers (extended abstract). In *22nd ACM STOC*, pages 84–94. ACM Press, May 1990. `doi:10.1145/100216.100226`.

28 Andrew Chi-Chih Yao. Quantum circuit complexity. In *34th FOCS*, pages 352–361. IEEE Computer Society Press, November 1993. `doi:10.1109/SFCS.1993.366852`.