# Improved Bounds on Quantum Learning Algorithms


Alp Atıcı*
*Department of Mathematics*
*Columbia University*
*New York, NY 10027*

Rocco A. Servedio†
*Department of Computer Science*
*Columbia University*
*New York, NY 10027*



(Dated: November 18, 2004)



In this article we give several new results on the complexity of algorithms that learn Boolean functions from quantum queries and quantum examples.

- Hunziker *et al.* [17] conjectured that for any class $C$ of Boolean functions, the number of quantum black-box queries which are required to exactly identify an unknown function from $C$ is $O(\frac{\log |C|}{\sqrt{\hat{\gamma}^C}})$, where $\hat{\gamma}^C$ is a combinatorial parameter of the class $C$. We essentially resolve this conjecture in the affirmative by giving a quantum algorithm that, for any class $C$, identifies any unknown function from $C$ using $O(\frac{\log |C| \log \log |C|}{\sqrt{\hat{\gamma}^C}})$ quantum black-box queries.

- We consider a range of natural problems intermediate between the exact learning problem (in which the learner must obtain all bits of information about the black-box function) and the usual problem of computing a predicate (in which the learner must obtain only one bit of information about the black-box function). We give positive and negative results on when the quantum and classical query complexities of these intermediate problems are polynomially related to each other.

- Finally, we improve the known lower bounds on the number of quantum examples (as opposed to quantum black-box queries) required for $(\epsilon, \delta)$-PAC learning any concept class of Vapnik-Chervonenkis dimension $d$ over the domain $\{0,1\}^n$ from $\Omega(\frac{d}{n})$ to $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta} + d + \frac{\sqrt{d}}{\epsilon})$. This new lower bound comes closer to matching known upper bounds for classical PAC learning.



Keywords: quantum query algorithms, quantum computation, computational learning theory, PAC learning


## I. INTRODUCTION

### A. Motivation and Background

A major focus of study in quantum computation is the power of quantum algorithms to extract information from a "black-box" oracle for an unknown Boolean function. Many of the most powerful ideas for both algorithmic results and lower bounds in quantum computing have emerged from this framework, which has been studied for more than a decade.

The most frequently considered problem in this setting is to determine whether or not the black-box oracle (which is typically assumed to belong to some particular *a priori* fixed class $C$ of possible functions) has some specific property, such as being identically 0 [4, 14], being exactly balanced between outputs 0 and 1 [10], or being invariant under an XOR mask [22]. However, as described below researchers have also studied several other problems in which the goal is to obtain more than just one bit of information about the target black-box function:

**Quantum exact learning from membership queries:** Servedio and Gortler [20] initiated a systematic study of the quantum black-box query complexity required to *exactly learn* any unknown function $c$ from a class $C$ of Boolean functions. This is a natural quantum analogue of the standard classical model of *exact learning from membership queries* which was introduced in computational learning theory by Angluin [2]. This quantum exact learning model was also studied by Hunziker *et al.* [17] and by Ambainis *et al.* [1], who gave a general upper bound on the quantum query complexity of learning any class $C$.

---


*Electronic address: atici@math.columbia.edu
†Electronic address: rocco@cs.columbia.edu

**PAC learning from quantum examples:** In another line of related research, Bshouty and Jackson [9] introduced a natural quantum analogue of Valiant's well-known Probably Approximately Correct (PAC) model of Boolean function learning [23] which is widely studied in computational learning theory. [20] subsequently gave a $\Omega(d/n)$ lower bound on the number of quantum examples required for any PAC learning algorithm for any class $C$ of Boolean functions over $\{0,1\}^n$ which has Vapnik-Chervonenkis dimension $d$.

### B. Our Results

In this paper we study three natural problems of quantum learning: (i) exact learning from quantum membership queries as described above; (ii) learning a *partition* of a class of functions from quantum membership queries (this is an intermediate problem between the quantum exact learning problem and the well-studied problem of obtaining a single bit of information about the target function), and (iii) quantum PAC learning as described above. For each of these problems we give new bounds on the number of quantum queries or examples that are required for learning.

For the quantum exact learning model, Hunziker *et al.* [17] conjectured that for any class $C$ of Boolean functions, the number of quantum black-box queries that are required to exactly learn an unknown function from $C$ is $O(\frac{\log |C|}{\sqrt{\hat{\gamma}^C}})$, where $\hat{\gamma}^C$ (defined in Section III A) is a combinatorial parameter of the class $C$. We give a new quantum exact learning algorithm based on a multi-target Grover search on a prescribed subset of the inputs, and show that the query complexity for this algorithm is $O(\frac{\log |C| \log\log |C|}{\sqrt{\hat{\gamma}^C}})$; this resolves the conjecture of Hunziker *et al.* [17] up to a $\log\log |C|$ factor. Our new bound is incomparable with the upper bound of Ambainis *et al.* [1], but as we show it improves on this bound for a wide range of parameter settings. We also show that for every class $C$ of Boolean functions, the query complexity of our generic algorithm is guaranteed to be at most a (roughly) quadratic factor worse than the query complexity of the *best* quantum algorithm for learning $C$ (which may be tailored for the specific class $C$).

For our second problem, we study a more general problem which is intermediate between learning the black-box function exactly and computing a single Boolean predicate of the unknown black-box function. This problem is the following: given a partition of a class $C$ into disjoint subsets $P_1, \ldots, P_k$, determine which piece the unknown black-box function $c \in C$ belongs to. Ambainis *et al.* proposed the study of this problem as an interesting direction for future work in [1]. Note that the problem of computing a single Boolean predicate of an unknown function $c \in C$ corresponds to having a two-way partition, whereas the problem of exact learning corresponds to a partition of $C$ into $|C|$ disjoint pieces.

We show that for any concept class $C$ and any partition size $2 \leq k \leq |C|$, there is a partition of $C$ into $k$ pieces such that the classical and quantum query complexities are polynomially related. On the other hand, we also show that for a wide range of partition sizes $k$ it is possible for the quantum and classical query complexities of learning a $k$-way partition to have a superpolynomial separation. These results show that the structure of the partition plays a more important role than the size in determining the relationship between quantum and classical complexity of learning.

Finally, for the quantum PAC learning model, we improve the $\Omega(\frac{d}{n})$ lower bound of [20] on the number of quantum examples which are required to PAC learn any concept class of Vapnik-Chervonenkis dimension $d$ over $\{0,1\}^n$. Our new bound of $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta} + d + \frac{\sqrt{d}}{\epsilon})$ is not far from the known lower bound of Ehrenfeucht *et al.* [11] of $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta} + \frac{d}{\epsilon})$ for classical PAC learning. Since the lower bound of [11] is known to be nearly optimal for classical PAC learning algorithms (an upper bound of $O(\frac{1}{\epsilon}\log\frac{1}{\delta} + \frac{d}{\epsilon}\log\frac{1}{\epsilon})$ was given by [6]), our new quantum lower bound is not far from being the best possible.

### C. Organization

Section III gives our new quantum algorithm for exactly learning a black-box function. Section IV gives some simple examples and poses a question about the relation between query complexity of quantum and classical exact learning. Section V gives our results on the partition learning problem, and Section VI gives our new lower bound on the sample complexity of quantum PAC learning. Section VII concludes with some additional open questions for further work.

## II. PRELIMINARIES

### A. Learning Preliminaries

A *concept* $c$ over $\{0,1\}^n$ is a Boolean function $c : \{0,1\}^n \to \{0,1\}$. Equivalently we may view a concept as a subset of $\{0,1\}^n$ defined by $\{x \in \{0,1\}^n : c(x) = 1\}$. A *concept class* $\mathcal{C} = \cup_{n \geq 1} C_n$ is a set of concepts where $C_n$ consists of those concepts in $\mathcal{C}$ whose domain is $\{0,1\}^n$. For ease of notation throughout the paper we will omit the subscript in $C_n$ and simply write $C$ to denote a collection of concepts over $\{0,1\}^n$. It will often be useful to think of $C$ as a $|C| \times 2^n$-binary matrix where rows correspond to concepts $c \in C$, columns correspond to inputs $x \in \{0,1\}^n$, and the $(i,j)$ entry of the matrix is the value of the $i$-th concept on the $j$-th input.

We say that a concept class $C$ is *1-sensitive* if it has the property that for each input $x$, at least half of all concepts $c \in C$ have $c(x) = 0$ (i.e. each column of the matrix $C$ is at most half ones). Given any $C$ it is possible to convert it to an equivalent 1-sensitive concept class by flipping the value obtained from any input $x$ which has $|\{c : c(x) = 1\}| > |\{c : c(x) = 0\}|$. This condition on $x$ can simply be checked by enumerating all concepts $c$ in $C$ – without making any queries. In general, we refer to the process of flipping the matrix entries which reside in a particular subset of columns as performing a *column flip*. This notion of 1-sensitivity and a column flip was first introduced by [1].

It is important to note that achieving the effect of a column flip in our algorithms involves creating and using simulated oracles. In other words, a column flip affects not only the matrix corresponding to the set of candidate concepts $C$ but also the result of classical and quantum membership queries. Therefore, after a column flip on the subset of inputs $K$, a membership query access to the target oracle at one of the inputs in $K$ should be considered to be inverted before returned to the algorithm. As remarked in [1], in both the classical and quantum learning models this can be achieved via some additional circuitry which is not significant for our purposes, since we are only interested in the query complexity.

### B. Classical Learning Models

The classical model of *exact learning from membership queries* was introduced by Angluin [2] and has since been studied by several authors [8, 13, 15, 16]. In this framework, a learning algorithm for $C$ is given query access to a black-box oracle $MQ_c$ for the unknown target concept $c \in C$, i.e. when the learner provides $x \in \{0,1\}^n$ to $MQ_c$ she receives back the value $c(x)$. A learning algorithm is said to be an *exact learning algorithm for concept class* $C$ if the following holds: for any $c \in C$, with probability at least 2/3 the learning algorithm outputs a Boolean circuit $h$ which is logically equivalent to $c$. (Note that a learning algorithm for $C$ "knows" the class $C$ but does not know the identity of the target concept $c \in C$.) The *query complexity* of a learning algorithm is the number of queries that it makes to $MQ_c$ before outputting $h$. We will be chiefly concerned in this paper with a quantum version of the exact learning model, which we describe in Section III.

In the classical PAC (Probably Approximately Correct) learning model, which was introduced by Valiant [23] and subsequently studied by many authors, the learning algorithm has access to a *random example oracle* $EX(c, \mathcal{D})$ where $c \in C$ is the unknown target concept and $\mathcal{D}$ is an unknown probability distribution over $\{0,1\}^n$. At each invocation the oracle $EX(c, \mathcal{D})$ (which takes no inputs) outputs a labeled example $(x, c(x))$ where $x \in \{0,1\}^n$ is drawn from the distribution $\mathcal{D}$. An algorithm $A$ is a *PAC learning algorithm for concept class* $C$ if the following condition holds: given any $\epsilon, \delta > 0$, for all $c \in C$ and all distributions $\mathcal{D}$ over $\{0,1\}^n$, if $A$ is given $\epsilon, \delta$ and is given access to $EX(c, \mathcal{D})$ then with probability at least $1 - \delta$ the output of $A$ is a Boolean circuit $h : \{0,1\}^n \to \{0,1\}$ (called a hypothesis) which satisfies $\Pr_{x \in \mathcal{D}}[h(x) \neq c(x)] \leq \epsilon$. The *(classical) sample complexity* of $A$ is the maximum number of calls to $EX(c, \mathcal{D})$ which it makes for any $c \in C$ and any distribution $\mathcal{D}$. In Section VI we will study a quantum version of the PAC learning model.

## III. EXACT LEARNING WITH QUANTUM MEMBERSHIP QUERIES

Given any concept $c : \{0,1\}^n \to \{0,1\}$, the *quantum membership oracle* $QMQ_c$ is the transformation which acts on the computational basis states by mapping $|x, b\rangle \mapsto |x, b \oplus c(x)\rangle$ where $x \in \{0,1\}^n$ and $b \in \{0,1\}$. A *quantum exact learning algorithm* for a concept class $C$ is a sequence of unitary transformations $U_0, QMQ_c, U_1, QMQ_c, \ldots, QMQ_c, U_T$ where each $U_i$ is a fixed unitary transformation without any dependence on $c$. The algorithm must satisfy the following property: for any target concept $c \in C$ which is used to instantiate the $QMQ$ queries, a measurement performed on the final state will with probability at least 2/3 yield a representation of a (classical) Boolean circuit $h : \{0,1\}^n \to \{0,1\}$ such that $h(x) = c(x)$ for all $x \in \{0,1\}^n$. The *quantum query complexity* of the algorithm is $T$, the number of invocations of $QMQ_c$.

Note that a quantum membership oracle $QMQ_c$ is identical to the notion of "a quantum black-box oracle for $c$" which has been widely studied in e.g. [3, 12, 14] and many other works. Most of this work, however, focuses on the quantum query complexity of computing a single bit of information about the unknown oracle, e.g. the OR of all its output values [14] or the parity of all its output values [12]. The quantum exact learning problem which we consider in this section was proposed in [20] and later studied in [1] (where it is called the "oracle identification problem") and in [17].

Throughout the paper we write $R(C)$ to denote the minimum query complexity of any classical (randomized) exact learning algorithm for concept class $C$. We write $Q(C)$ to denote the minimum query complexity of any quantum exact learning algorithm for $C$. We write $N$ to denote $2^n$, the number of elements in the domain of each $c \in C$.

In Section III A we briefly recap known bounds on the query complexity of quantum and classical exact learning algorithms. In Section III B we give our new quantum learning algorithm, prove correctness, and analyze its query complexity.

### A. Known bounds on query complexity for exact learning

We begin by defining a combinatorial parameter $\hat{\gamma}^C$ of a concept class $C$ which plays an important role in bounds on query complexity of exact learning algorithms.

**Definition III.1** *Let $C$ be a concept class over $\{0,1\}^n$. We define*

$$\gamma_a^{C'} = \min_{b \in \{0,1\}} |\{c \in C' : c(a) = b\}|/|C'|, \qquad \text{where } a \in \{0,1\}^n, C' \subseteq C$$

$$\gamma^{C'} = \max_{a \in \{0,1\}^n} \gamma_a^{C'}, \qquad \text{where } C' \subseteq C$$

$$\hat{\gamma}^C = \min_{C' \subseteq C, |C'| \geq 2} \gamma^{C'}.$$

If $C' \subseteq C$ is the set of possible remaining target concepts, then $\gamma^{C'}$ is the maximum fraction of $C'$ which a (classical) learning algorithm can be sure of eliminating with a single query. Thus, intuitively, the smaller $\hat{\gamma}^C$ is the more membership queries should be required to learn $C$.

The following lower and upper bounds on the query complexity of classical exact learning were established in [8]:

**Theorem III.2** *For any concept class $C$ we have $R(C) = \Omega(\frac{1}{\hat{\gamma}^C})$ and $R(C) = \Omega(\log |C|)$.*

**Theorem III.3** *There is a classical exact learning algorithm which learns any concept class $C$ using $O(\frac{\log |C|}{\hat{\gamma}^C})$ many queries, so consequently $R(C) = O(\frac{\log |C|}{\hat{\gamma}^C})$.*

A quantum analogue of this classical lower bound was obtained in [20]:

**Theorem III.4** *For any concept class $C$ over $\{0,1\}^n$ we have $Q(C) = \Omega(\frac{1}{\sqrt{\hat{\gamma}^C}})$ and $Q(C) = \Omega(\frac{\log |C|}{n})$.*

Given these results it is natural to seek a quantum analogue of the classical $O(\frac{\log |C|}{\hat{\gamma}^C})$ upper bound. Hunziker *et al.* [17] made the following conjecture:

**Conjecture III.5** *There is a quantum exact learning algorithm which learns any concept class $C$ using $O(\frac{\log |C|}{\sqrt{\hat{\gamma}^C}})$ quantum membership queries.*

In Section III B we prove this conjecture up to a $\log \log |C|$ factor.

Hunziker *et al.* [17] also conjectured that there is a quantum exact learning algorithm which learns any concept class $C$ using $O(\sqrt{|C|})$ queries. This was established by Ambainis *et al.* [1], who also proved the following result:

**Theorem III.6** *There is a quantum exact learning algorithm which learns any concept class $C$ with $|C| > N$ using $O(\sqrt{N \log |C| \log N} \log \log |C|)$ many queries.*

### B. A New Quantum Exact Learning Algorithm

We start with a simple yet useful observation:

**Lemma III.7** *For any concept class $C$, there exists an $x \in \{0,1\}^n$ for which at least a $\hat{\gamma}^C$ fraction of concepts $c \in C$ satisfy $c(x) = 1$. More generally for every subset $C' \subseteq C$ with $|C'| \geq 2$, there exists an input $x$ at which the fraction of concepts in $C'$ yielding $1$ is at least $\gamma^{C'}$ (which is at least as large as $\hat{\gamma}^C$).*

**Proof:** It is sufficient to prove the result in the latter general form. Consider any subset $C' \subseteq C$ with $|C'| \geq 2$. By Definition III.1 we know:

- $\gamma^{C'} \geq \hat{\gamma}^C$.
- At any input $z \in \{0,1\}^n$, the fraction of concepts in $C$ yielding 1 has to be at least $\gamma_z^{C'}$.

Now consider the input $a$ which satisfies $\gamma_a^{C'} = \gamma^{C'}$: the fraction of concepts in $C$ yielding 1 at input $a$ should therefore be at least $\gamma^{C'}$. Thus taking $x = a$ gives the intended result. ■

The quantity $\hat{\gamma}^C$ can be bounded as follows:

**Lemma III.8** *For any concept class $C$ with $|C| \geq 2$, $\frac{1}{N+1} \leq \gamma^C \leq \frac{1}{2}$. This also implies $\frac{1}{N+1} \leq \hat{\gamma}^C \leq \frac{1}{2}$ by Definition III.1.*

**Proof:** $\gamma^C \leq \frac{1}{2}$ is clear from the Definition III.1. To prove the other direction we may assume that $\gamma^C < \frac{1}{N}$, since otherwise the result is obviously true. Therefore $|C| > N$ must hold. Observe that at each input $x$ one of the following must hold:

- The fraction of concepts in $C$ yielding 0 at $x$ is at least $1 - \gamma^C$ and thus the fraction of concepts in $C$ yielding 1 at $x$ is at most $\gamma^C$.
- The fraction of concepts in $C$ yielding 1 at $x$ is at least $1 - \gamma^C$ and thus the fraction of concepts in $C$ yielding 0 at $x$ is at most $\gamma^C$.

Hence $C$ can contain at most $\gamma^C|C|N$ concepts which are not identically equal to the concept $c_{\mathsf{maj}}$ defined as follows:

$$c_{\mathsf{maj}}(x) = \begin{cases} 0, & \text{if at least half of the concepts in } C \text{ yield 0 at } x; \\ 1, & \text{otherwise.} \end{cases}$$

Therefore $C$ must be comprised of these $\gamma^C|C|N$ concepts and possibly $c_{\mathsf{maj}}$. Thus we obtain:

$$\gamma^C|C|N + 1 \geq |C| \Longrightarrow \gamma^C \geq \frac{|C|-1}{|C|N} \geq \frac{N}{(N+1)N} = \frac{1}{N+1}$$

■

**Definition III.9** *A subset of inputs $\mathcal{I} \subseteq \{0,1\}^n$ is said to satisfy* the semi-rich row condition for $C$ *if at least half the concepts in $C$ have the property that they yield $1$ for at least a $\hat{\gamma}^C$ fraction of the inputs $x$ in $\mathcal{I}$.*

The phrase "semi-rich row condition" is used because viewing $C$ as a matrix, at least half the rows of $C$ are "rich" in 1s (have at least a $\hat{\gamma}^C$ fraction of 1s) within the columns indexed by inputs in $\mathcal{I}$. A simple greedy approach can be used to construct a set of inputs which satisfies the semi-rich row condition for $C$:

**Lemma III.10** *Let $C$ be any concept class with $|C| \geq 2$. Then Algorithm 1 outputs a set of inputs $\mathcal{I}$ with $|\mathcal{I}| \leq \frac{1}{\hat{\gamma}^C}$ which satisfies the semi-rich row condition for $C$.*

**Proof:** Let $\tau_j|C|$ be the number of concepts in $C \setminus S$ after the $j$-th execution of the repeat loop in Algorithm 1, so $\tau_0 = 1$. Using the first result of Lemma III.7, we obtain $\tau_1 \leq 1 - \hat{\gamma}^C$. Now invoking Lemma III.7 once again but this time in its general form, we obtain $\tau_2 \leq (1 - \hat{\gamma}^C)^2$; note that after the second iteration of the loop, each concept in $S$ will yield 1 on at least half of the elements in $\mathcal{I}$. Let $j'$ equal $\lfloor \frac{1}{\hat{\gamma}^C} \rfloor$. If Algorithm 1 proceeds for $j'$ iterations through the loop, then it must be the case that $\tau_{j'} \leq (1 - \hat{\gamma}^C)^{j'}$ and that each concept in $S$ yields 1 on at least a $\hat{\gamma}^C$ fraction of the elements in $\mathcal{I}$. It is easy to verify that $(1-x)^{\lfloor 1/x \rfloor} < 1/2$ for $0 < x < \frac{1}{2}$. We thus have that $\tau_{j'}|C| < |C|/2$, Algorithm 1: Constructing a set of inputs which satisfies the semi-rich row condition.

$S \leftarrow \emptyset$, $\mathcal{I} \leftarrow \emptyset$.
**repeat**
  Perform a column flip on $C \setminus S$ to make $C \setminus S$ be 1-sensitive.
  $a_{max} \leftarrow$ the input in $\{0,1\}^n \setminus \mathcal{I}$ at which the highest fraction of concepts in $C \setminus S$ yield 1.
  $\mathcal{I} \leftarrow \mathcal{I} \cup \{a_{\max}\}$.
  $S \leftarrow S \cup$ the set of concepts in the original matrix $C$ that yield 1 at input $a_{max}$.
**until** $|S| \geq |C|/2$.
Output $\leftarrow \mathcal{I}$.

and consequently $|S| > |C|/2$, so $\mathcal{I}$ satisfies the semi-rich row condition for $C$ and the algorithm will terminate before starting the $(\lfloor \frac{1}{\hat{\gamma}^C} \rfloor + 1)$-th iteration.

In the case $\hat{\gamma}^C \leq \frac{1}{N}$, then the set of all $N$ inputs will satisfy the semi-rich row condition for $C$: since any concept which does not yield 0 for all inputs actually yields 1 for at least a $\hat{\gamma}^C$ fraction of all inputs. Therefore in this case the algorithm will terminate successfully with an output $|\mathcal{I}| \leq N \leq \frac{1}{\hat{\gamma}^C}$. Otherwise, since $\hat{\gamma}^C > \frac{1}{N}$, we have that $j' < N$. This means the algorithm never runs out of inputs to add (i.e. $\{0,1\}^N \setminus \mathcal{I}$ is nonempty at every iteration). ■

Our quantum learning algorithm is given in Algorithm 2. Throughout the algorithm the set $S \subseteq C$ should be viewed as the set of possible target concepts that have not yet been eliminated; the algorithm halts when $|S| = 1$. The high-level idea of the algorithm is that in every repetition of the outer loop the size of $S$ is multiplied by a factor which is at most $\frac{1}{2}$, so at most $\log |C|$ repetitions of the outer loop are performed.

Algorithm 2: A Quantum Exact Learning Algorithm.

$S \leftarrow C$.
**repeat**
  Perform a column flip on $S$ to make $S$ 1-sensitive. Let $\mathcal{K}$ be the set of inputs at which the output is flipped during this procedure.
  $\mathcal{I} \leftarrow$ The output of Algorithm 1 invoked on the set of concepts $S$.
  Counter $\leftarrow 0$, Success$\leftarrow$`False`.
  **repeat**
    Perform the multi-target subset Grover search on $\mathcal{I}$ using $\frac{9}{2}\lceil\sqrt{|\mathcal{I}|}\rceil$ queries [7].
    $a \leftarrow$ Result of the Grover search.
    **if** a classical query of the oracle at $a$ yields 1 **then**
      $S \leftarrow$ {the concepts in $S$ that yield 1 at $a$}, Success$\leftarrow$`True`.
    **end if**
    Counter$\leftarrow$Counter+1.
  **until** Success Or (Counter>= $\log(3 \log |C|)$)
  **if** Not Success **then**
    $S \leftarrow$ the set of concepts that yield 0 for all of the inputs in $\mathcal{I}$.
  **end if**
  Flip the outputs of concepts in $S$ for all elements in $\mathcal{K}$ to reverse the earlier column flip (thus restoring all concepts in $S$ to their original behavior on all inputs).
**until** $|S| = 1$.
Output $\leftarrow$ A representation of a circuit which computes the sole concept $c$ in $S$.

**Theorem III.11** *Let $C$ be any concept class with $|C| \geq 2$. Algorithm 2 is a quantum exact learning algorithm for $C$ which performs $O(\frac{\log |C| \log\log |C|}{\sqrt{\hat{\gamma}^C}})$ quantum membership queries.*

**Proof:** Consider a particular iteration of the outer Repeat-Until loop. The set $S$ is 1-sensitive by virtue of the first step (the column flip). By Lemma III.10, in the second step of this iteration, $\mathcal{I}$ becomes a set of at most $\frac{1}{\hat{\gamma}^S}$ many inputs which satisfies the semi-rich row condition for $S$. Consequently, each execution of the Grover search within the inner Repeat-Until loop uses $O(\sqrt{\frac{1}{\hat{\gamma}^S}})$ (which is also $O(\sqrt{\frac{1}{\hat{\gamma}^C}}))$ many queries. Since the inner loop repeats at most $\log(3 \log |C|)$ many times, if we can show that each iteration of the outer loop does indeed with high probability (i) cause the size of $S$ to be multiplied by a factor which is at most $\frac{1}{2}$, and (ii) maintain the property that the target concept is contained in $S$, then the theorem will be proved.

As shown in [7], the multi-target Grover search algorithm over a space of $|\mathcal{I}|$ many inputs using $\frac{9}{2}\lceil\sqrt{|\mathcal{I}|}\rceil$ many queries has the property that if there is any input $a \in \mathcal{I}$ on which the target function yields 1, then the search will output such an $a$ with probability at least $\frac{1}{2}$. Since the inner loop repeats $\log(3\log|C|)$ many times, we thus have that if there is any input $a \in \mathcal{I}$ on which the target concept yields 1, then with probability at least $1 - \frac{1}{3\log|C|}$ one of the $\log(3\log|C|)$ many iterations of the inner loop will yield such an $a$ and the "Success" variable will be set to `True`. Since the set $S$ is 1-sensitive, when we eliminate from $S$ all the concepts which yield 0 at $a$ we will multiply the size of $S$ by at most $\frac{1}{2}$ as desired in this case (and clearly we will not eliminate the target concept from $S$). On the other hand, if the set $\mathcal{I}$ contains no input $a$ on which the target concept yields 1, then after $\log(3\log|C|)$ iterations of the inner loop we will exit with Success set to `False`, and the concepts that yield 1 on any input in $\mathcal{I}$ will be removed from $S$. This will clearly not cause the target to be removed from $S$. Moreover because $|\mathcal{I}| \leq \frac{1}{\hat{\gamma}^S}$, any concept for which even a single input in $\mathcal{I}$ yields 1 has the property that at least a $\hat{\gamma}^S$ fraction of inputs in $\mathcal{I}$ yield 1. Since $\mathcal{I}$ satisfies the semi-rich row condition for $S$, this means that we have eliminated at least half the concepts in $S$. Thus, the algorithm will succeed with probability at least $(1 - \frac{1}{3\log|C|})^{\log C}$ which is larger than 2/3, and the theorem is proved. ■

Recall that $Q(C)$ denotes the optimal query complexity over all quantum exact learning algorithms for concept class $C$. We can show that the query complexity of Algorithm 2 is never much worse than the optimal query complexity $Q(C)$:

**Corollary III.12** *For any concept class $C$, Algorithm 2 uses $O(nQ(C)^2 \log\log|C|)$ queries.*

**Proof:** This follows directly from Theorem III.11 and the bound $Q(C) = \Omega(\frac{\log|C|}{n} + \frac{1}{\sqrt{\hat{\gamma}^C}})$ of Theorem III.4. ■

Since $|C| \leq 2^{2^n}$, the bound $O(nQ(C)^2 \log\log|C|)$ is always $O(n^2 Q(C)^2)$, and thus the query complexity of Algorithm 2 is always polynomially related to the query complexity of the optimal algorithm for any concept class $C$.

### C. Discussion

Algorithm 2 can be viewed as a variant of the algorithm of [1] which learns any concept class $C$ from $O(\sqrt{N \log|C| \log N} \log\log|C|)$ quantum membership queries. This algorithm repeatedly performs Grover search over the set of all inputs, with the goal each time of eliminating at least half of the remaining target concepts. Instead, our approach is to perform each Grover search only over sets which satisfy the semi-rich row condition for the remaining set of possible target concepts. By doing this, we are able to obtain an upper bound on query complexity in terms of $\hat{\gamma}^C$ for every such iteration.

We observe that our new bound of $O(\frac{\log|C| \log\log|C|}{\sqrt{\hat{\gamma}^C}})$ is stronger than the previously obtained upper bound of $O(\sqrt{N \log|C| \log N} \log\log|C|)$ from [1] as long as $\frac{\log|C|}{\hat{\gamma}^C} = o(N \log N)$. Thus, for any concept class $C$ for which the $O(\frac{\log|C|}{\hat{\gamma}^C})$ upper bound of Theorem III.3 on classical membership query algorithms is nontrivial (i.e. is less than $N$), our results give an improvement.

We note that independently Iwama *et al.* [18] have recently given a new algorithm for quantum exact learning that uses ideas similar to the construction of Algorithm 1; however the analysis is different and their results are incomparable to ours (their bounds depend only on the number of concepts in $C$ and not on the combinatorial parameter $\hat{\gamma}^C$). The main focus of [18] is on obtaining robust learning algorithms that can learn successfully using noisy oracle queries.

## IV. RELATIONS BETWEEN QUERY COMPLEXITY OF QUANTUM AND CLASSICAL EXACT LEARNING

As noted in [20], combining Theorems III.3 and III.4 yields the following:

**Corollary IV.1** *For any concept class $C$, we have $Q(C) \leq R(C) = O(nQ(C)^3)$.*

Can tighter bounds relating $R(C)$ and $Q(C)$ be given which hold for all concept classes $C$? While we have not been able to answer this question, here we make some simple observations and pose a question which we hope will stimulate further work.

We first observe that the factor $n$ is required in the bound $R(C) = O(nQ(C)^3)$:

**Lemma IV.2** *For any positive integer $d$ there exists a concept class $C$ over $\{0,1\}^n$ with $R(C) = \omega(1)$ which has $R(C) = \Omega(Q(C)^d)$.*

**Proof:** We assume $d > 1$. Recall that in the Bernstein-Vazirani problem, the target concept is an unknown parity function over some subset of the $n$ Boolean variables $x_1, \dots, x_n$; Bernstein and Vazirani showed [5] that for this concept class we have $R(C) = n$ whereas $Q(C) = 1$. We thus consider a concept class in which each concept $c$ contains $n^{1/d}$ copies of the Bernstein-Vazirani problem (each instance of the problem is over $n^{(d-1)/d}$ variables) as follows: we view $n$-bit strings $a, x$ as

$$a = (a_{1,1}, a_{1,2}, \dots, a_{1,n^{(d-1)/d}}, a_{2,1}, a_{2,2}, \dots, a_{2,n^{(d-1)/d}}, \dots a_{n^{1/d},1}, \dots, a_{n^{1/d},n^{(d-1)/d}})$$
$$x = (x_{1,1}, x_{1,2}, \dots, x_{1,n^{(d-1)/d}}, x_{2,1}, x_{2,2}, \dots, x_{2,n^{(d-1)/d}}, \dots x_{n^{(1/d)},1}, \dots, x_{n^{1/d},n^{(d-1)/d}})$$

The class $C$ consists of the set of all $2^n$ concepts:

$$f_a(x) = \bigvee_{i=1}^{n^{1/d}} ((a_{i,1}, a_{i,2}, \dots, a_{i,n^{(d-1)/d}}) \cdot (x_{i,1}, x_{i,2}, \dots, x_{i,n^{(d-1)/d}}) \bmod 2)$$

i.e. $f_a(x)$ equals 1 if any of the $n^{1/d}$ parities corresponding to the substrings $a_{i,\cdot}$ take value 1 on the corresponding substring of $x$.

It is easy to see that $n$ queries suffice for a classical algorithm, and by Theorem III.2 we have $R(C) = \Omega(\log |C|)$, so $R(C) = \Theta(n)$. On the other hand, it is also easy to see that $Q = O(n^{\frac{1}{d}})$ since a quantum algorithm can learn by making $n^{1/d}$ successive runs of the Bernstein-Vazirani algorithm.

Finally, if $d = 1$ then as shown in [24] the set $C$ of all $2^{2^n}$ concepts over $\{0,1\}^n$ has $Q(C) = \Theta(2^n)$ and $R(C) = \Theta(2^n)$. ■

The bound $R(C) = O(nQ(C)^3)$ implies that the gap of Lemma IV.2 can only be achieved for concept classes $C$ which have $R(C)$ small. However, it is easy to exhibit concept classes which have a factor $n$ difference between $R(C)$ and $Q(C)$ for a wide range of values of $R(C)$:

**Lemma IV.3** *For any $k$ such that $n - k = \Theta(n)$, there is a concept class $C$ with $R(C) = \Theta(n2^k)$ and $Q(C) = \Theta(2^k)$.*

**Proof:** The concept class $C$ is defined as follows. A concept $c \in C$ corresponds to $(a^0, \dots, a^{2^k-1})$, where each $a^i$ is a $(n-k)$-bit string. The concept $c$ maps input $x \in \{0,1\}^n$ to $(a^i \cdot y) \bmod 2$, where $i$ is the number between 0 and $2^k - 1$ whose binary representation is the first $k$ bits of $x$ and $y$ is the $(n-k)$-bit suffix of $x$. Since each concept in $C$ is defined uniquely by $2^k$ many $(n-k)$-bit strings $a^0, \dots, a^{2^k-1}$, there are $2^{2^k(n-k)}$ concepts in $C$.

Theorem III.2 yields $R(C) = \Omega(2^k(n-k))$. It is easy to see that in fact $R(C) = \Theta(2^k(n-k))$: For each of the $2^k$ parities which one must learn (corresponding to the $2^k$ possible prefixes of an input), one can learn the $(n-k)$-bit parity with $n-k$ classical queries.

It is also easy to see that by running the Bernstein-Vazirani algorithm $2^k$ times (once for each different $k$-bit prefix), a quantum algorithm can learn an unknown concept from $C$ exactly using $2^k$ queries, and thus $Q(C) = O(2^k)$. The $Q(C) = \Omega(\frac{\log |C|}{n})$ lower bound of Theorem III.2 gives us $Q(C) = \Omega(\frac{n-k}{n} \cdot 2^k) = \Omega(2^k)$, and the lemma is proved. ■

Based on these observations, we pose the following question:

**Question IV.4** *Does every concept class $C$ satisfy $R(C) = O(nQ(C) + Q(C)^2)$?*

Note that the example in Lemma IV.3 and the concept class of Grover search [14]: $C = \{f_i, 0 \le i < N : f_i(x) = \delta_{i,x}\}$ saturate this upper bound.

## V. ON LEARNING A PARTITION OF A CONCEPT CLASS

**Definition V.1** *Let $C$ be a concept class over $\{0,1\}^n$. A* partition $\mathcal{P}$ of $C$ *is a collection of nonempty disjoint subsets $P_1, \dots, P_k$ whose union yields $C$.*

In this section we study a different problem, mentioned by Ambainis *et al.* [1], that is more relaxed than exact learning: given a partition $\mathcal{P}$ of $C$ and a black-box (quantum or classical) oracle for an unknown target concept $c$ in $C$, what is the query complexity of identifying the set $P_i$ in $\mathcal{P}$ which contains $c$? It is easy to see that both the exact learning problem (in which $|\mathcal{P}| = |C|$) and the problem of computing some binary property of $c$ (for which $|\mathcal{P}| = 2$)

are special cases of this more general problem. One can view these problems in the following way: for the exact learning problem the algorithm must obtain all $\log |C|$ bits of information about the target concept, whereas for the problem of computing a property of $c$ the algorithm must obtain a single bit of information. In a general instance of the partition problem, the algorithm must obtain $\log |\mathcal{P}|$ bits of information about the target concept.

Given a concept class $C$ and a partition $\mathcal{P}$ of $C$, we will write $R_{\mathcal{P}}(C)$ to denote the optimal query complexity of any classical (randomized) algorithm for the partition problem which outputs the correct answer with probability at least $2/3$ for any target concept $c$. We similarly write $Q_{\mathcal{P}}(C)$ to denote the optimal complexity of any quantum query algorithm with the same success criterion.

As noted earlier, for the case $|\mathcal{P}| = |C|$ we know from Corollary IV.1 that the quantities $R_{\mathcal{P}}(C)$ and $Q_{\mathcal{P}}(C)$ are polynomially related for any concept class $C$, since $R_{\mathcal{P}}(C) = O(nQ_{\mathcal{P}}(C)^3)$. On the other extreme, if $|\mathcal{P}| = 2$ then concept classes are known for which $R_{\mathcal{P}}(C)$ and $Q_{\mathcal{P}}(C)$ are polynomially related (see e.g. [3]), and concept classes are also known for which there is an exponential gap [22]. It is thus natural to investigate the relationship between the size of $|\mathcal{P}|$ and the existence of a polynomial relationship between $R_{\mathcal{P}}(C)$ and $Q_{\mathcal{P}}(C)$.

In this section, we show that the number of sets in $|\mathcal{P}|$ alone (viewed as a function of $|C|$) often does not provide sufficient information to determine whether $R_{\mathcal{P}}(C)$ and $Q_{\mathcal{P}}(C)$ are polynomially related. More precisely, in Section V A we show that for *any* concept class $C$ over $\{0,1\}^n$ and any value $2 \le k \le |C|$, there is a partition $\mathcal{P}$ of $C$ with $|\mathcal{P}| = k$ for which we have $R_{\mathcal{P}}(C) = O(nQ_{\mathcal{P}}(C)^3)$. On the other hand, in Section V B we show that for a wide range of values of $|\mathcal{P}|$ (again as a function of $|C|$), there are concept classes which have a superpolynomial separation between $R_{\mathcal{P}}(C)$ and $Q_{\mathcal{P}}(C)$. Thus, our results concretely illustrate that the structure of the partition (rather than the number of the sets in the partition) plays an important role in determining whether the quantum and classical query complexities are polynomially related.

### A. Partition Problems for which Quantum and Classical Complexity are Polynomially Related

The following simple lemma extends the cardinality-based lower bounds of Theorem III.2 and Theorem III.4 for exact learning to the problem of learning a partition:

**Lemma V.2** *For any partition $\mathcal{P}$ of any concept class $C$ over $\{0,1\}^n$, we have $R_{\mathcal{P}}(C) = \Omega(\log |\mathcal{P}|)$ and $Q_{\mathcal{P}}(C) = \Omega(\frac{\log |\mathcal{P}|}{n})$.*

**Proof:** Let $C' \subseteq C$ be a concept class formed by taking any single element from each subset in the partition $\mathcal{P}$. Learning $\mathcal{P}$ requires at least as many queries as exact learning the concept class $C'$, and so the result follows from Theorem III.2 and Theorem III.4. ■

To obtain a partition analogue of the other lower bounds of Theorems III.2 and III.4, we define the following combinatorial parameter which is an analogue of $\hat{\gamma}^C$:

**Definition V.3** *Let $\mathcal{S}$ be the set of all subsets $C' \subseteq C$, $|C'| \ge 2$ which have the property that any subset $C'' \subseteq C'$ with $|C''| \ge \frac{3}{4}|C'|$ must intersect at least two subsets in $\mathcal{P}$. We define $\hat{\gamma}^C_{\mathcal{P}}$ to be $\hat{\gamma}^C_{\mathcal{P}} := \min_{C' \subseteq \mathcal{S}} \gamma^{C'}$.*

Thus each subset $C'$ in $\mathcal{S}$ has the property that the partition induced on $C'$ by $\mathcal{P}$ contains no subset of size as large as $\frac{3}{4}|C'|$.

The next lemma shows that for each $C' \in \mathcal{S}$, the lower bounds for exact learning $R(C') = \Omega(\frac{1}{\gamma^{C'}})$ and $Q(C') = \Omega(\frac{1}{\sqrt{\gamma^{C'}}})$ which are implied by Theorems III.2 and III.4 extend to the problem of learning a partition to yield $R_{\mathcal{P}}(C') = \Omega(\frac{1}{\gamma^{C'}})$ and $Q_{\mathcal{P}}(C') = \Omega(\frac{1}{\sqrt{\gamma^{C'}}})$. By considering the $C' \in \mathcal{S}$ which minimizes $\gamma^{C'}$, we obtain the strongest lower bound (this is the motivation behind Definition V.3).

**Lemma V.4** *For any partition $\mathcal{P} = P_1, \dots, P_k$ of the concept class $C$, we have $R_{\mathcal{P}}(C) = \Omega(\frac{1}{\hat{\gamma}^C_{\mathcal{P}}})$ and $Q_{\mathcal{P}}(C) = \Omega(\frac{1}{\sqrt{\hat{\gamma}^C_{\mathcal{P}}}})$.*

**Proof:** Let $C' \in \mathcal{S}$ be such that $\hat{\gamma}^C_{\mathcal{P}} = \gamma^{C'}$. We consider the problem of learning the partition induced by $\mathcal{P}$ over $C'$, and shall prove the lower bound for this easier problem. We may assume without loss of generality that $C'$ is 1-sensitive.

We first consider the classical case. We claim that there is a partition $\{S_1, S_2\}$ of $C'$ with the property that each subset $(P_j \cap C')$ is contained entirely in exactly one of $S_1, S_2$ (i.e. $S_1, S_2$ is a "coarsening" of the partition induced by $\mathcal{P}$ over $C'$) which satisfies $\min_{i=1,2} |S_i| > \frac{1}{4}|C'|$. To see this, we may start with $S_2 = \emptyset$, $S_1 = \cup_{i=1}^k (P_i \cap C') = C'$

and consider a process of "growing" $S_2$ by successively removing the smallest piece $P_j \cap C'$ from $S_1$ and adding it to $S_2$. W.l.o.g. we may suppose that $|P_j \cap C'| \leq |P_{j+1} \cap C'|$ for all $j$, so the pieces $P_j \cap C'$ are added to $S_2$ in order of increasing $j = 1, 2, \ldots$. Let $t$ be the index such that adding $P_t \cap C'$ to $S_2$ causes $|S_2|$ to exceed $\frac{1}{4}|C'|$ for the first time. By Definition V.3 it cannot be the case that adding $P_t \cap C'$ causes $S_2$ to become all of $C'$ (since this would mean that $P_t \cap C'$ is a subset of size at least $\frac{3}{4}|C'|$ which intersects only $P_t$); thus it must be the case that after this $t$-th step $S_1$ is still nonempty. However, it also cannot be the case that after this $t$-th step we have $|S_1| < \frac{3}{8}|C'|$; for if this were the case, then after the $t$-th step we would have $|P_t \cap C'| > \frac{3}{8}|C'| > |S_1| = \cup_{j=t+1}^{k}(P_j \cap C')$ and this would violate the assumption that sets are added to $S_2$ in order of increasing size.

Since $C'$ is 1-sensitive, the "worst case" for a learning algorithm is that each classical query to the target concept (some $c \in C'$) yields 0. By definition of $\gamma^{C'}$, each such query eliminates at most $\gamma^{C'} \cdot |C'|$ many possible target concepts from $C'$. Consequently, after $\lfloor \frac{1/4}{\gamma^{C'}} \rfloor - 1$ classical queries, the set of possible target concepts in $C'$ is of size at least $\frac{3}{4}|C'|$, and so it must intersect both $S_1$ and $S_2$. It is thus impossible to determine with probability greater than $1/2$ whether $c$ belongs to $S_1$ or $S_2$, and thus which piece $P_i$ of $\mathcal{P}$ contains $c$. This gives the classical lower bound.

Our analysis for the quantum case requires some basic definitions and facts about quantum computing:

**Definition V.5** *If $|\phi\rangle = \sum_z \alpha_z|z\rangle$ and $|\psi\rangle = \sum_z \beta_z|z\rangle$ are two superpositions of basis states, then the* Euclidean distance *between $|\phi\rangle$ and $|\psi\rangle$ is $\||\phi\rangle - |\psi\rangle\| = (\sum_z |\alpha_z - \beta_z|^2)^{1/2}$. The* total variation distance *between two distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ is defined to be $\sum_x |\mathcal{D}_1(x) - \mathcal{D}_2(x)|$.*

**Fact V.6 (See [5])** *Let $|\phi\rangle$ and $|\psi\rangle$ be two unit length superpositions which represent possible states of a quantum register. If the Euclidian distance $\||\phi\rangle - |\psi\rangle\|$ is at most $\epsilon$, then performing the same observation on $|\phi\rangle$ and $|\psi\rangle$ induces distributions $\mathcal{D}_\phi$ and $\mathcal{D}_\psi$ which have total variation distance at most $4\epsilon$.*

For the quantum lower bound, suppose we have a quantum learning algorithm which makes at most $T = \lfloor \frac{1/4}{32\sqrt{\gamma^{C'}}} \rfloor - 1$ quantum membership queries. We will use the following result which combines Theorem 6 and Lemma 7 from [20] (those results are in turn based on Theorem 6.6 of [4]):

**Lemma V.7 (See [20])** *Consider any quantum exact learning algorithm $\mathcal{N}$ which makes $T$ quantum membership queries. Let $|\phi_T^c\rangle$ denote the state of the quantum register after all $T$ membership queries are performed in the algorithm, if the target concept is $c$. Then for any 1-sensitive set $C'$ of concepts with $|C'| \geq 2$ and any $\epsilon > 0$, there is a set $S \subseteq C'$ of cardinality at most $T^2|C'|\gamma^{C'}/\epsilon^2$ such that for all $c \in C' \setminus S$, we have $\||\phi_T^{\mathbf{0}}\rangle - |\phi_T^c\rangle\| \leq \epsilon$ (where $\mathbf{0}$ denotes the identically 0 concept).*

If we take $\epsilon = \frac{1}{32}$, then Lemma V.7 implies that there exists a set $S \subseteq C'$ of cardinality less than $\frac{1}{4} \cdot |C'|$ such that for all $c \in C' \setminus S$ one has $\||\phi_T^{\mathbf{0}}\rangle - |\phi_T^c\rangle\| \leq \frac{1}{32}$. Consequently by Definition V.3 there must exist two concepts $c_1, c_2 \in C' \setminus S$ with $\||\phi_T^{c_1}\rangle - |\phi_T^{c_2}\rangle\| \leq \frac{1}{16}$ which belong to different subsets $P_i$ and $P_j$ of $\mathcal{P}$. By Fact V.6, the probability that our quantum learning algorithm outputs "$i$" can differ by at most $\frac{1}{4}$ when the target concept is $c_1$ versus $c_2$; but this contradicts the assumption that the algorithm is correct with probability at least 2/3 on all target concepts. This proves the quantum lower bound. ■

Before proving the main result of this section, we establish the following result which gives a sufficient condition for the quantum and classical complexities $Q_{\mathcal{P}}(C)$ and $R_{\mathcal{P}}(C)$ of learning a partition to be polynomially related. This result is a generalization of Corollary IV.1.

**Corollary V.8** *For a partition $\mathcal{P}$ over the concept class $C$, if the size of the largest subset $P_i$ in $\mathcal{P}$ is less than $\frac{3/4}{\hat{\gamma}^C}$, then we have $R_{\mathcal{P}}(C) = O(nQ_{\mathcal{P}}(C)^3)$.*

**Proof:** Let $C'$ be a subset of $C$ for which $\gamma^{C'}$ equals $\hat{\gamma}^C$. We have that $|C'| \geq \frac{1}{\hat{\gamma}^C}$ by Definition III.1. Thus any $\frac{3}{4}$ fraction of $C'$ must intersect at least two subsets in $\mathcal{P}$, so $C'$ must belong to $\mathcal{S}$. This forces $\hat{\gamma}^C_{\mathcal{P}} = \gamma^{C'} = \hat{\gamma}^C$. Moreover, we have that $|\mathcal{P}| \geq \frac{4}{3}\hat{\gamma}^C \cdot |C|$, and we know that $\frac{1}{N+1} \leq \hat{\gamma}^C \leq \frac{1}{2}$ by Lemma III.8. Thus we have $\log|\mathcal{P}| \geq \log\frac{4}{3} - n + \log|C|$, and consequently $\frac{\log|\mathcal{P}|}{n} > \frac{\log|C|}{n} - 1$. Lemmas V.2 and V.4 yield $Q_{\mathcal{P}}(C) = \Omega(\frac{\log|\mathcal{P}|}{n} + \frac{1}{\sqrt{\hat{\gamma}^C_{\mathcal{P}}}}) = \Omega(\frac{\log|C|}{n} + \frac{1}{\sqrt{\hat{\gamma}^C}})$. Combining this with the bound $R_{\mathcal{P}}(C) = O(\frac{\log|C|}{\hat{\gamma}^C})$ (which clearly follows from Theorem III.3 since the partition learning problem is no harder than the exact learning problem), we have that $R_{\mathcal{P}}(C)$ must be $O(nQ_{\mathcal{P}}(C)^3)$. ■

We note here that we could have used any constant $\lambda$ satisfying $\frac{2}{3} < \lambda < 1$ in Definition V.3 in place of 3/4, and obtained corresponding versions of Lemma V.4 and the above corollary with $\lambda$ in place of 3/4.

Algorithm 3: A slightly modified version of Algorithm 1 to be used in generating a partition.

**Require:** $C$ is 1-sensitive.
$R \leftarrow \emptyset$, $\mathcal{I} \leftarrow \emptyset$, $\mathcal{J} \leftarrow \emptyset$.
**repeat**
  Perform a column flip on $C \setminus R$ to make it 1-sensitive; call the resulting 1-sensitive matrix $M$.
  $a_{max} \leftarrow$ the input in $\{0,1\}^n \setminus \mathcal{I}$ at which the highest fraction of concepts in $C \setminus R$ yield 1.
  $\mathcal{I} \leftarrow \mathcal{I} \cup \{a_{\max}\}$.
  $R \leftarrow R \cup$ the set of concepts in $M$ that yield 1 at input $a_{max}$.
  **if** the column corresponding to $a_{\max}$ in $M$ was flipped relative to $C$ **then**
    $\mathcal{J} \leftarrow \mathcal{J} \cup \{a_{\max}\}$.
  **end if**
**until** $|R| \geq |C|/2$.
Output $\leftarrow (\mathcal{I}, \mathcal{J})$.

Now we prove our main result of this subsection, showing that for *any* concept class $C$ and any partition size bound $2 \leq k \leq |C|$ there is a partition of $C$ into $k$ pieces such that the classical and quantum query complexities are polynomially related:

**Theorem V.9** *Let $C$ be any concept class and $k$ any integer satisfying $2 \leq k \leq |C|$. Then there is a partition $\mathcal{P}$ of $C$ with $|\mathcal{P}| = k$ for which we have $R_{\mathcal{P}}(C) = O(nQ_{\mathcal{P}}(C)^3)$.*

**Proof:** We will show that Algorithm 4 constructs a partition $\mathcal{P}$ with the desired properties. Algorithm 4 uses a slightly modified version of Algorithm 1, which we call Algorithm 3. Algorithm 3 differs from Algorithm 1 in that if the input $a_{\max}$ corresponds to a column which is flipped in the column flip on $C \setminus R$, then Algorithm 3 augments $R$ by adding those concepts in the flipped version of $C \setminus R$ which yield 1 on $a_{\max}$ (note that by 1-sensitivity this is fewer than half of the concepts in $C \setminus R$), whereas Algorithm 1 adds those concepts which yield 1 on $a_{\max}$ in the unflipped (original) version of $C \setminus R$. Thus at each stage Algorithm 3 grows the set $R$ by adding at most half of the remaining concepts in $C \setminus R$; we will need this property later. The analysis of Algorithm 1 carries over to show that the set $\mathcal{I}$ of inputs which Algorithm 3 constructs is of size at most $|\mathcal{I}| \leq \frac{1}{\hat{\gamma}^C}$.

At each iteration of the outer repeat loop, Algorithm 4 successively refines the partition $\mathcal{Q}$ until $|\mathcal{Q}| = k$. Let $C' \subseteq C$ be such that $\gamma^{C'} = \hat{\gamma}^C$. The first time Algorithm 4 passes through the inner repeat loop we will have $|C| = |S|$ and thus Algorithm 3 will be invoked on $C'$. We will write $C^\circ, C^\star$ to denote these sets $S^\circ$, $S^\star$ of concepts that are formed out of $C$ in this first iteration. The final partition $\mathcal{P}$ will ultimately be a refinement of the partition $\{C^\circ, C^\star\}$ obtained in this step; we will see later that this will force $\hat{\gamma}^C_{\mathcal{P}} = \hat{\gamma}^C$ (this is why the first iteration is treated differently than later iterations).

In addition to constructing the partition $\mathcal{P}$, the execution of Algorithm 4 should also be viewed as a "memoization" process in which various sets of inputs $\mathcal{I}(S), \mathcal{J}(S)$ and $\mathcal{K}(S)$ are defined to correspond to different sets of concepts $S$. These sets will be used during the execution of Algorithm 5 later. Roughly speaking, the division of $S$ in each iteration depends only on the values on inputs in $\mathcal{I}(S)$, the set $\mathcal{J}(S)$ is used to keep track of the column flips Algorithm 3 performs, and the set $\mathcal{K}(S)$ keeps track of those inputs which need to be flipped to achieve 1-sensitivity.

We now explain the outer loop of Algorithm 4 in more detail. The algorithm works in a breadth-first fashion to successively refine the partition $\mathcal{Q}$, which is initially just $\{C\}$, into the final partition $\mathcal{P}$. After the first iteration of the outer loop, $C$ has been partitioned into $\{C^\circ, C^\star\}$. Similarly, in the second iteration each of these sets is divided in two to give a four-way partition. The algorithm continues in this manner until the desired number of elements in the partition is reached. The main idea of the construction is that each division of a set $S$ (after the first iteration) creates two pieces $S^\circ$ and $S^\star$ of almost equal size as we shall describe below. Because degenerate divisions do not occur, we will see the algorithm will terminate after at most $O(\log k)$ iterations of the outer loop.

Recall from above that each invocation of Algorithm 3 in Algorithm 4 on a set $S$ of concepts yields a set $\mathcal{I}(S)$ of at most $\frac{1}{\hat{\gamma}^S}$ many inputs. By flipping the output of concepts in $S$ at inputs in $\mathcal{J}(S)$ in Step 14 of Algorithm 4, we ensure that the sets $S^\circ$ and $S^\star$ defined in steps 15 and 16 correspond precisely to the sets $C \setminus R$ and $R$ of Algorithm 3 when it terminates. It thus follows from the termination condition of Algorithm 3 that $|S^\star|/|S| \geq \frac{1}{2}$. Recall also from the discussion in the first paragraph of this proof that the last iteration of the loop of Algorithm 3 adds at most half of the remaining concepts into the set $R$. Therefore we have that the set $S^\circ$ in Algorithm 4 must satisfy $|S^\circ|/|S| > \frac{1}{4}$. It follows from these bounds on $|S^\circ|/|S|$ and $|S^\star|/|S|$ that Algorithm 4 makes at most $O(\log k)$ many iterations through the outer loop.

It is therefore clear that at each iteration of the main loop of Algorithm 4 for which $|S| \geq 2$, each of the sets $S^\circ$ and $S^\star$ formed from $S$ will be nonempty. This ensures that the algorithm will keep producing new elements in the partition

Algorithm 4: Constructing a partition for which $R_{\mathcal{P}}(C)$ and $Q_{\mathcal{P}}(C)$ are polynomially related.

$\mathcal{Q} \leftarrow \{C\}$<br>
**repeat**<br>
  $\mathcal{R} \leftarrow \emptyset$.<br>
  **repeat**<br>
    $S \leftarrow$ an element in $\mathcal{Q}$<br>
    **if** $|S| \geq 2$ **then**<br>
      **if** $|C| = |S|$ **then**<br>
        Let $\mathcal{K}(S)$ denote the inputs which, if flipped, would make $C'$ be 1-sensitive ($C'$ is defined in the 2nd paragraph of the proof of Theorem V.9). Flip the values of concepts in $S$ at inputs in $\mathcal{K}(S)$.<br>
        $(\mathcal{I}(S), \mathcal{J}(S)) \leftarrow$ The output of Algorithm 3 invoked with $C'$.<br>
      **else**<br>
        Let $\mathcal{K}(S)$ denote the inputs which, if flipped, would make $S$ be 1-sensitive. Flip the values of concepts in $S$ at inputs in $\mathcal{K}(S)$.<br>
        $(\mathcal{I}(S), \mathcal{J}(S)) \leftarrow$ The output of Algorithm 3 invoked with input $S$.<br>
      **end if**<br>
      Flip the values of concepts in $S$ at those inputs in $\mathcal{J}(S)$.<br>
      $S^{\circ} \leftarrow \{$the concepts in $S$ that yield 0 for each $x \in \mathcal{I}(S)\}$.<br>
      $S^{\star} \leftarrow S \setminus S^{\circ}$.<br>
      Flip the values of concepts in $S^{\circ}, S^{\star}$ for all elements in $\mathcal{J}(S)$.<br>
      Flip the values of concepts in $S^{\circ}, S^{\star}$ for all elements in $\mathcal{K}(S)$.<br>
      $\mathcal{R} \leftarrow \mathcal{R} \cup \{S^{\circ}, S^{\star}\}$.<br>
    **else**<br>
      $\mathcal{R} \leftarrow \mathcal{R} \cup \{S\}$.<br>
    **end if**<br>
    $\mathcal{Q} \leftarrow \mathcal{Q} \setminus \{S\}$.<br>
  **until** $\mathcal{Q} = \emptyset$ OR $|\mathcal{Q}| + |\mathcal{R}| = k$.<br>
  $\mathcal{Q} \leftarrow \mathcal{Q} \cup \mathcal{R}$.<br>
**until** $|\mathcal{Q}| = k$.<br>
$\mathcal{P} \leftarrow \mathcal{Q}$.

until $|\mathcal{P}| = k$ is reached. The same argument shows that $C^{\circ}, C^{\star}$ are each nonempty and satisfy $|C^{\circ} \cap C'|/|C'| > \frac{1}{4}$ and $|C^{\star} \cap C'|/|C'| \geq \frac{1}{2}$. This implies that $C'$ is an element of the set $\mathcal{S}$ of Definition V.3: any subset $C'' \subseteq C'$ with $|C''| \geq \frac{3}{4}|C'|$ must intersect both $C^{\circ}$ and $C^{\star}$, and thus must intersect at least two subsets of $\mathcal{P}$ since $\mathcal{P}$ is a refinement of $\{C^{\circ}, C^{\star}\}$. Consequently we have $\hat{\gamma}^C_{\mathcal{P}} = \hat{\gamma}^C$.

To show that the partition $\mathcal{P}$ satisfies $R_{\mathcal{P}}(C) = O(nQ_{\mathcal{P}}(C)^3)$, we now give an analysis of the query complexity of learning $\mathcal{P}$ with both classical and quantum resources. As we will see, we need to give a classical upper bound and a quantum lower bound to obtain our goal.

In the classical case, we will show that Algorithm 5 makes $O(\frac{\log|\mathcal{P}|}{\hat{\gamma}^C})$ queries and successfully learns the partition $\mathcal{P}$. Using the sets $\mathcal{I}(S)$ which were defined by the execution of Algorithm 4, Algorithm 5 makes its way down the correct branch of the binary tree implicit in the successive refinements of Algorithm 4 to find the correct piece of the partition which contains $c$. More precisely, at the end of the $t$-th iteration of the outer loop of Algorithm 5, the set $S$ which Algorithm 5 has just obtained will be identical to the piece $c$ resides in of the partition constructed by Algorithm 4 at the end of the $t$-th iteration of its outer loop. As shown above, it takes $O(\log k) = O(\log|\mathcal{P}|)$ iterations until the subset which the target concept $c$ lies in is reached. Moreover, by the same argument in Lemma III.10, Algorithm 3 always outputs a set of inputs $\mathcal{I}(S)$ with size at most $\frac{1}{\hat{\gamma}^S} \leq \frac{1}{\hat{\gamma}^C}$ when invoked on a set of concepts $S$. Therefore at each of these $O(\log|\mathcal{P}|)$ iterations Algorithm 5 makes at most $\frac{1}{\hat{\gamma}^C}$ many queries. Thus Algorithm 5 is a classical algorithm that learns $\mathcal{P}$ using $O(\frac{\log|\mathcal{P}|}{\hat{\gamma}^C})$ queries, so we have $R_{\mathcal{P}}(C) = O(\frac{\log|\mathcal{P}|}{\hat{\gamma}^C})$.

In the quantum case: since we have $\hat{\gamma}^C_{\mathcal{P}} = \hat{\gamma}^C$, by Lemma V.4 any quantum algorithm learning $\mathcal{P}$ should perform $\Omega(\frac{1}{\sqrt{\hat{\gamma}^C}})$ quantum membership queries. Combining this result with that of Lemma V.2, we have that $Q_{\mathcal{P}}(C) = \Omega(\frac{\log|\mathcal{P}|}{n} + \frac{1}{\sqrt{\hat{\gamma}^C}})$. Combining this inequality with the classical upper bound $R_{\mathcal{P}} = O(\frac{\log|\mathcal{P}|}{\hat{\gamma}^C})$ from Algorithm 5, we have that $R_{\mathcal{P}}(C) = O(nQ_{\mathcal{P}}(C)^3)$ for this partition $\mathcal{P}$, and we are done. ∎

Algorithm 5: A classical algorithm learning $\mathcal{P}$.

$S \leftarrow C$
**repeat**
Flip the values of concepts in $S$ at those inputs in $\mathcal{K}(S)$.
Flip the values of concepts in $S$ at those inputs in $\mathcal{J}(S)$.
Classically query the given oracle implementing $c$ at all elements in $\mathcal{I}(S)$.
$Z \leftarrow$`True` if $c$ yields 0 for all elements in $\mathcal{I}(S)$. $Z \leftarrow$`False` otherwise.
**if** $Z$ **then**
$S \leftarrow \{$the concepts in $S$ that yield 0 for each $x \in \mathcal{I}(S)\}$.
**else**
$S \leftarrow \{$the concepts in $S$ that yield 1 for at least one $x \in \mathcal{I}(S)\}$.
**end if**
Flip the values of concepts in $S$ at those inputs in $\mathcal{J}(S)$.
Flip the values of concepts in $S$ at those inputs in $\mathcal{K}(S)$.
**until** $S \in \mathcal{P}$.
Output$\leftarrow S$.

## B. A Partition Problem with a Large Quantum-Classical Gap

In the previous subsection we showed that for any concept class and any partition size bound, there is a partition problem for which the classical and quantum query complexities are polynomially related. In this section, by adapting a result of Simon [22] we show that for a wide range of values of the partition size bound, it is possible for the classical query complexity to be superpolynomially larger than the quantum query complexity:

**Theorem V.10** *Let $n = m + \log m$. For any value $1 \le \ell < m$, there is a concept class $C$ over $\{0,1\}^n$ with $|C| < 2^{m\ell-\ell^2+\ell+m2^{m-\ell}}$ and a partition $\mathcal{P}$ of $C$ with $|\mathcal{P}| > 2^{m\ell-\ell^2-\ell}$ such that $R_{\mathcal{P}}(C) = \Omega(2^{m-\ell})$ and $Q_{\mathcal{P}}(C) =$poly$(m)$).*

Taking $\ell = m - \alpha(m)$ where $\alpha(m)$ is any function which is $\omega(\log m)$, we obtain $R_{\mathcal{P}}(C) = m^{\omega(1)}$ whereas $Q_{\mathcal{P}}(C) =$poly$(m)$, for a superpolynomial separation between classical and quantum query complexity. Such a choice of $\ell$ gives $|\mathcal{P}| = 2^{\Omega(m)}$ whereas $|C|$ is roughly $2^{m\cdot 2^{\alpha(m)}}$. Note that the size of $|C|$ can be made to be $2^{\beta(m)}$ for any slightly superpolynomial function $\beta(m)$ via a suitable choice of $\alpha(m) = \omega(\log m)$. This means that viewed as a function of $|C|$, it is possible for $|\mathcal{P}|$ to be as large as $2^{(\log|C|)^{\varepsilon(|C|)}}$ for any function $\varepsilon(\cdot) = o(1)$ and still have the classical query complexity be superpolynomially larger than the quantum query complexity.

**Proof of Theorem V.10:** We will use a result of Simon [22] who considers functions $f : \{0,1\}^m \to \{0,1\}^m$. Any such function $f : \{0,1\}^m \to \{0,1\}^m$ can equivalently be viewed as a function $\tilde{f} : (\{0,1\}^m \times \{1,\dots,m\}) \to \{0,1\}$ where $\tilde{f}(x,j)$ equals $f(x)_j$, the $j$-th bit of $f(x)$. It is easy to see that we can simulate a call to an oracle for $f : \{0,1\}^m \to \{0,1\}^m$ by making $m$ membership queries to the oracle for $\tilde{f}$, in both the classical and quantum case. This extra factor of $m$ is immaterial for our bounds, so we will henceforth discuss functions $f$ which map $\{0,1\}^m$ to $\{0,1\}^m$.

We view the input space $\{0,1\}^m$ as the vector space $\mathbb{F}_2^m$. Given a $\ell$-dimensional vector subspace $V \subset \mathbb{F}_2^m$, we say that a function $f : \{0,1\}^m \to \{0,1\}^m$ is *$V$-invariant* if the following condition holds: $f(x) = f(y)$ if and only if $x = y \oplus v$ for some $v \in V$. Thus a $V$-invariant function $f$ is defined by the $2^{m-\ell}$ distinct values it takes on representatives of the $2^{m-\ell}$ cosets of $V$. The concept class $C$ is the class of all functions $f$ which are $V$-invariant for some $\ell$-dimensional vector subspace $V$ of $\mathbb{F}_2^m$.

A simple counting argument shows that there are

$$N_{m,\ell} = \frac{(2^m-1)(2^m-2)(2^m-4)\cdots(2^m-2^{\ell-1})}{(2^\ell-1)(2^\ell-2)(2^\ell-4)\cdots(2^\ell-2^{\ell-1})}$$

many $\ell$-dimensional subspaces of $\mathbb{F}_2^m$. This is because there are $(2^m-1)(2^m-2)(2^m-4)\cdots(2^m-2^{\ell-1})$ ways to choose an ordered list of $\ell$ linearly independent vectors to form a basis for $V$, and given any $V$ there are $(2^\ell-1)(2^\ell-2)(2^\ell-4)\cdots(2^\ell-2^{\ell-1})$ ordered lists of vectors from $V$ which could serve as a basis.

We define the partition $\mathcal{P}$ to divide $C$ into $N_{m,\ell}$ equal-size subsets, one for each $\ell$-dimensional vector subspace $V$; the subset of concepts corresponding to a given $V$ is precisely those functions which are $V$-invariant. For any given $\ell$-dimensional subspace $V$, the number of functions that are $V$-invariant is

$$I_{m,\ell} = 2^m(2^m-1)(2^m-2)\cdots(2^m-2^{m-\ell}+1)$$

since one can uniquely define such a function by specifying distinct values to be attained on each of the $2^{m-\ell}$ coset representatives.

Therefore we have $|C| = N_{m,\ell} \cdot I_{m,\ell}$, and it is easy to check that $2^{m\ell-\ell^2-\ell} \leq N_{m,\ell} \leq 2^{m\ell-\ell^2+\ell}$ and $I_{m,\ell} \leq 2^{m2^{m-\ell}}$. It remains only to analyze the quantum and classical query complexities.

For the quantum case, it follows easily from Simon's analysis of his algorithm in [22] that for any $V$-invariant $f$, each iteration of Simon's algorithm (which requires a single quantum query to $f$) will yield a vector that is independently and uniformly drawn from the $(m-\ell)$-dimensional subspace $V^{\perp}$. A standard analysis shows that after $O(m)$ iterations, with very high probability we will have obtained a set of vectors that span $V^{\perp}$; from these it is easy to identify $V$ and thus the piece of the partition to which $f$ belongs.

For the classical case, an analysis much like that of ([22], Section 3.2) can be used to show that any classical algorithm which correctly identifies the vector subspace $V$ with high probability must make $2^{\Omega(m-\ell)}$ many queries; since the proof is similar to [22] we only sketch the main ideas here. We say that a sequence of queries is *good* if it contains two distinct queries which yield the same output (i.e. two queries $x, y$ which have $(x \oplus y) \in V$), and otherwise the sequence is *bad*. The argument of [22] applied to our setting shows that if the target vector subspace is chosen uniformly at random, then any classical algorithm making $M = 2^{(m-\ell)/3}$ queries makes a good sequence of queries with very small probability. On the other hand, if a sequence of $M = 2^{(m-\ell)/3}$ queries is bad, then this restricts the possibilities for $V$ by establishing a set $S$ of $\binom{M}{2} < 2^{2(m-\ell)/3}$ many "forbidden" vectors from $\mathbb{F}_2^m$ which must not belong to the target vector space $V$ (since for each pair of elements $x, y$ in $M$ we know that $(x \oplus y) \notin V$). Given a fixed nonzero vector $z \in \mathbb{F}_2^m$, we have that a random $\ell$-dimensional vector space $W$ contains $z$ with probability $\frac{2^\ell-1}{2^m-1} < \frac{2^\ell}{2^m}$, and consequently the probability that a random $W$ contains any element of $S$ is at most $2^{2(m-\ell)/3} \cdot \frac{2^\ell}{2^m} = 2^{(\ell-m)/3}$, which is less than $1/2$ if $\ell < m-6$ (and if $\ell \geq m-6$ the bound of the theorem is trivially true). Thus at least half of the $N_{m,\ell}$ possible $\ell$-dimensional vector subspaces are compatible with any given bad sequence of $2^{(m-\ell)/3}$ queries, so the classical algorithm cannot have identified the right subspace with nonnegligible probability. ■

## VI. QUANTUM VERSUS CLASSICAL PAC LEARNING

### A. The Quantum PAC Learning Model

The quantum PAC learning model was introduced by Bshouty and Jackson in [9]. A quantum PAC learning algorithm is defined analogously to a classical PAC algorithm, but instead of having access to a random example oracle $EX(c, \mathcal{D})$ it can (repeatedly) access a *quantum superposition* of labeled examples. The goal of constructing a classical Boolean circuit $h$ which is an $\epsilon$-approximator for $c$ with probability $1-\delta$ is unchanged. More precisely, for $\mathcal{D}$ a distribution over $\{0,1\}^n$ we say that the *quantum example oracle* $QEX(c, \mathcal{D})$ is a gate which transforms the computational basis state $|0^n, 0\rangle$ as follows:

$$|0^n, 0\rangle \mapsto \sum_{x \in \{0,1\}^n} \sqrt{\mathcal{D}(x)}|x, c(x)\rangle.$$

We leave the action of a $QEX(c, \mathcal{D})$ gate undefined on other basis states, and we require that a quantum PAC learning algorithm may only invoke a $QEX(c, \mathcal{D})$ oracle on the basis state $|0^n, 0\rangle$. It is easy to verify (see [9]) that a $QEX(c, \mathcal{D})$ oracle can simulate a classical $EX(c, \mathcal{D})$ oracle.

As noted in Lemma 6 of [9], we may assume without loss of generality (by renumbering qubits) that all $QEX(c, \mathcal{D})$ calls of a quantum PAC learning algorithm occur sequentially at the beginning of the algorithm's execution and that the $t$-th invocation of $QEX(c, \mathcal{D})$ affects the qubits $(t-1)(n+1)+1, (t-1)(n+1)+2, \ldots, t(n+1)$. After all $T$ $QEX(c, \mathcal{D})$ queries have been performed, the algorithm performs a fixed unitary transformation and then a measurement takes place. (See [9, 20] for more details on the quantum PAC learning model.) The *quantum sample complexity* is the number of invocations $T$ of $QEX$ which the quantum PAC learning algorithm performs, i.e. the number of $QEX$ gates in the quantum circuit corresponding to the quantum PAC learning algorithm.

The following definition plays an important role in the sample complexity of both classical and quantum PAC learning:

**Definition VI.1** *If $C$ is a concept class over some domain $X$ and $W \subseteq X$, we say that $W$ is* shattered *by $C$ if for every $W' \subseteq W$, there exists a $c \in C$ such that $W' = c \cap W$. The* Vapnik-Chervonenkis dimension *of $C$, $VC-DIM(C)$, is the cardinality of the largest $W \subseteq X$ such that $W$ is shattered by $C$.*

### B. Known Results on Quantum versus Classical PAC Learning

The classical sample complexity of PAC learning has been intensively studied and nearly matching upper and lower bounds are known:

**Theorem VI.2** *(i) [11] Any classical $(\epsilon, \delta)$-PAC learning algorithm for a non-trivial concept class $C$ of VC dimension $d$ must have classical sample complexity $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta} + \frac{d}{\epsilon})$. (ii) [6] Any concept class $C$ of VC dimension $d$ can be $(\epsilon, \delta)$-PAC learned by a classical algorithm with sample complexity $O(\frac{1}{\epsilon}\log\frac{1}{\delta} + \frac{d}{\epsilon}\log\frac{1}{\epsilon})$.*

Servedio and Gortler [20] gave a lower bound on the quantum sample complexity of PAC learning. They showed that for any concept class $C$ of VC dimension $d$ over $\{0,1\}^n$, if the distribution $\mathcal{D}$ is uniform over the $d$ examples in some shattered set $S$, then even if the learning algorithm is allowed to make quantum membership queries on any superposition of inputs in the domain $S$, any algorithm which with high probability outputs a high-accuracy hypothesis with respect to $\mathcal{D}$ must make at least $\Omega(\frac{d}{n})$ many queries. Such membership queries can simulate $QEX(c, \mathcal{D})$ queries since the support of $\mathcal{D}$ is $S$, and thus this gives a lower bound on the sample complexity of quantum PAC learning with a $QEX$ oracle.

### C. Improved Lower Bounds on Quantum Sample Complexity of PAC Learning

In this section we give improved lower bounds on the sample complexity of quantum $(\epsilon, \delta)$-PAC learning algorithms for concept classes $C$ of VC dimension $d$. These new bounds nearly match the classical lower bounds of [11].

We first note that the $\Omega(\frac{d}{n})$ lower bound of [20] can be easily strengthened to $\Omega(d)$:

**Observation VI.3** *Let $C$ be any concept class of VC dimension $d$ and let $\mathcal{D}$ be the uniform distribution over a shattered set $S$ of size $d$. Then any quantum learning algorithm which (i) can make quantum membership queries on any superposition of inputs in the domain $S$, and (ii) with high probability outputs a hypothesis with error rate at most $\epsilon = \frac{1}{10}$, must make at least $\frac{d}{100}$ queries (and consequently the sample complexity of PAC learning $C$ with a $QEX$ oracle is $\Omega(d)$).*

Recall that in the exact learning model, the concept class $C$ of all parity functions over $n$ Boolean variables has VC dimension $d = n$ yet can be exactly learned with one call to a quantum membership oracle using the Bernstein-Vazirani algorithm [5]. In light of this, we feel that this improvement from $\Omega(\frac{d}{n})$ to $\Omega(d)$ is somewhat unexpected, and may even at first appear contradictory. The key to the apparent contradiction is that the Bernstein-Vazirani algorithm makes its membership query on a superposition of all $2^n$ inputs in $\{0,1\}^n$, not just the $n$ inputs in a fixed shattered set $S$.

**Proof:** It suffices to slightly sharpen the proof of Theorem 4.2 from [20]. The key observation is that since queries always have zero amplitude on computational basis states outside of the shattered set $S$, the effective value of the domain size $N$ is $|S| = d$ rather than $|\{0,1\}^n| = 2^n$. With this modification, at the end of the proof of Theorem 4.2 we obtain the inequality $N_0 = \sum_{i=0}^{2T}\binom{d}{i} \geq 2^{d/6}$ where $T$ is the quantum query complexity of the algorithm (instead of the inequality $\sum_{i=0}^{2T}\binom{2^n}{i} \geq 2^{d/6}$ which appears in [20]). Now standard tail bounds on binomial coefficients (see e.g. Appendix 9 of [19]) show that $T > \frac{d}{100}$. ■

We now give lower bounds on the quantum query complexity of $(\epsilon, \delta)$-PAC learning which depend on $\epsilon$ and $\delta$. We require the following definition and fact:

**Definition VI.4** *A concept class $C$ is said to be* trivial *if either $C$ contains only one concept, or $C$ contains exactly two concepts $c_0, c_1$ with each $x \in \{0,1\}^n$ belonging to exactly one of $c_0, c_1$.*

**Fact VI.5 (See [21])** *Let $|\psi^{(0)}\rangle, |\psi^{(1)}\rangle$ represent states of a quantum system such that for some measurement $\Pi$ we have $\langle\psi^{(0)}|\Pi|\psi^{(0)}\rangle \geq 1 - \delta$ and $\langle\psi^{(1)}|\Pi|\psi^{(1)}\rangle \leq \delta$ for some $\delta > 0$. Then we have $|\langle\psi^{(0)}|\psi^{(1)}\rangle| \leq 2\sqrt{\delta(1-\delta)}$.*

It is clear that a trivial concept class can be learned exactly from any single (classical) example. For nontrivial concept classes [6] gave a classical sample complexity lower bound of $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta})$. We now extend this bound to the quantum setting:

**Lemma VI.6** *Any quantum algorithm with a $QEX(c, \mathcal{D})$ oracle which $(\epsilon, \delta)$-learns a non-trivial concept class must have quantum sample complexity $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta})$.*

**Proof:** Since $C$ is non-trivial, without loss of generality we may assume that there are two inputs $x^0, x^1$ and two concepts $c_0, c_1 \in C$ such that $c_0(x_0) = c_1(x_0) = 0$ while $c_0(x_1) = 0, c_1(x_1) = 1$.

Let $\mathcal{D}$ be the distribution where $\mathcal{D}(x_0) = 1 - 3\epsilon$ and $\mathcal{D}(x_1) = 3\epsilon$. Under this distribution, no hypothesis which is $\epsilon$-accurate for $c_0$ can be $\epsilon$-accurate for $c_1$ and vice versa.

Let $|\psi_T^{(i)}\rangle$ be the state of the system immediately after the $T$ queries of $QEX(c_i, \mathcal{D})$ are performed. Then we have

$$\langle\psi_T^{(i)}|\underbrace{x_0, 0, x_0, 0, \ldots, x_0, 0}_{\text{repeated } T \text{ times}}, 0 \ldots, 0\rangle = (1 - 3\epsilon)^{T/2}, \quad \text{for } i = 0, 1.$$

It is easy to see that any other computational basis state $|\ldots, x_1, b, \ldots\rangle$ which has nonzero amplitude in $|\psi_T^{(b)}\rangle$ must have zero amplitude in the other possible state $|\psi_T^{(1-b)}\rangle$, because $c_0$ and $c_1$ disagree on $x_1$. Consequently we have $\langle\psi^{(0)}|\psi^{(1)}\rangle = (1 - 3\epsilon)^T$. If $(1 - 3\epsilon)^T > 2\sqrt{\delta(1-\delta)}$ then Fact VI.5 dictates there is some output hypothesis which occurs with probability greater than $\delta$ whether the target is $c_0$ or $c_1$; but this cannot be the case for an $(\epsilon, \delta)$-PAC learning algorithm. Thus we must have $(1 - 3\epsilon)^{2T} \leq 4\delta$ yielding $T = \Omega(\frac{1}{\epsilon}\log\frac{1}{\delta})$. ■

Ehrenfeucht *et al.* [11] obtained a $\Omega(\frac{d}{\epsilon})$ lower bound for classical PAC learning by considering a distribution $\mathcal{D}$ which distributes $\Theta(\epsilon)$ weight evenly over all but one of the elements in a shattered set. In other words under $\mathcal{D}$ one element in the shattered set has weight $1 - \Theta(\epsilon)$ and all the remaining $d - 1$ elements has equal weight $\frac{\Theta(\epsilon)}{d-1}$. We use such a distribution to obtain the following quantum lower bound (no attempt has been made to optimize constants):

**Theorem VI.7** *Let $C$ be any concept class of VC dimension $d + 1$. Let $\delta = 1/5$. Then we have that for sufficiently large $d$ (i.e. $d \geq 625$ suffices) and any $0 < \epsilon < \frac{1}{32}$, any quantum algorithm with a $QEX(c, \mathcal{D})$ oracle which $(\epsilon, \delta)$-learns $C$ must have quantum sample complexity at least $\frac{\sqrt{d}}{10000\epsilon}$.*

**Proof:** Let $\{x_0, x_1, \ldots, x_d\}$ be a set of inputs which is shattered by $C$. We consider the distribution $\mathcal{D}$, first introduced by [11], which has $\mathcal{D}(x_0) = 1 - 8\epsilon$ and $\mathcal{D}(x_i) = \frac{8\epsilon}{d}$ for $i = 1, \ldots, d$. Let $H(x) = -x\log x - (1 - x)\log(1 - x)$ denote the binary entropy function. As is noted in [20], there exists a set $s^1, \ldots, s^A$ of $d$-bit strings such that for all $i \neq j$ the strings $s^i$ and $s^j$ differ in at least $d/4$ positions where $A \geq 2^{d(1-H(1/4))} > 2^{d/6}$. For each $i = 1, \ldots, A$ let $c_i$ be a concept such that (i) $c_i(x_0) = 0$, and (ii) the $d$-bit string $(c_i(x_1), \ldots, c_i(x_d))$ is $s^i$. The existence of such concepts follows from Definition VI.1. Since we have $\epsilon < \frac{1}{32}$, our quantum PAC learning algorithm should successfully distinguish between any two target concepts $c_i$ and $c_j$ with confidence at least $1 - \delta = \frac{4}{5}$. Moreover, without loss of generality we may suppose that $\epsilon < \frac{1}{100\sqrt{d}}$ since otherwise Observation VI.3 yields the required lower bound.

We shall make use of the following standard inequality:

$$(1 - x)^\ell \geq 1 - x\ell, \text{if } x\ell < 1 \text{ for } \ell \in \mathbb{Z}^+, x \in \mathbb{R}^+. \tag{1}$$

Given a target concept $c$, we write $|\xi_{i_1,i_2,\ldots,i_t}\rangle$ to denote the basis state

$$|\xi_{i_1,i_2,\ldots,i_t}\rangle = |x_{i_1}, c(x_{i_1}), x_{i_2}, c(x_{i_2}), \ldots x_{i_t}, c(x_{i_t})\rangle.$$

We define the state $|\phi_t\rangle$ to be

$$|\phi_t\rangle = (1 - 8\epsilon)^{t/2}|\xi_{0,0,\ldots,0}\rangle + (1 - 8\epsilon)^{\frac{t-1}{2}}\sqrt{\frac{8\epsilon}{d}}\sum_{i=1}^{d}(|\xi_{i,0,0,\ldots,0}\rangle + |\xi_{0,i,0,\ldots,0}\rangle + \ldots + |\xi_{0,0,\ldots,i}\rangle) + \alpha|z\rangle.$$

Here $|z\rangle$ is some canonical basis state which is distinct from, and hence orthogonal to, all states of the form $|\xi_{i_1,i_2,\ldots,i_t}\rangle$, e.g. we could take $z = |x_1, c(x_1), x_1, 1 - c(x_1), 0, 0, \ldots, 0\rangle$. The scalar $\alpha$ is a suitable normalizing coefficient so that the Euclidean norm of $|\phi_t\rangle$ is 1.

Let $|\psi_t\rangle$ denote the state of the quantum register after $t$ invocations of $QEX(c, \mathcal{D})$ have occured. It is easy to see from the definition of the $QEX(c, \mathcal{D})$ oracle that the amplitude of $|\psi_t\rangle$ on the basis state $|\xi_{0,\ldots,0}\rangle$ will be $(1-8\epsilon)^{t/2}$, and that for each of the $td$ many basis states $|\xi_{0,0,\ldots,0,i,0,\ldots,0}\rangle$ (where $i$ ranges over all $t$ positions and ranges in value from 1 to $d$) the amplitude of $|\psi_t\rangle$ on this basis state will be $(1-8\epsilon)^{\frac{t-1}{2}}\sqrt{\frac{8\epsilon}{d}}$. We thus have that $\langle\psi_t|\phi_t\rangle = (1-8\epsilon)^t\left(1 + \frac{8t\epsilon}{1-8\epsilon}\right)$.

If we let $t = \frac{1}{100\epsilon\sqrt{d}}$ (note that $t \geq 1$ by our assumption that $\epsilon < \frac{1}{100\sqrt{d}}$), we then have that $(1 - 8\epsilon)^t > (1 - \frac{1}{12\sqrt{d}})$ by (1), and it is easy to check that $(1 + \frac{8t\epsilon}{1-8\epsilon}) > 1 + \frac{1}{12\sqrt{d}}$ from the bounds on $\epsilon$ and $d$ in the theorem statement. We thus have that

$$\langle\psi_t|\phi_t\rangle > 1 - \frac{1}{144d}. \tag{2}$$

Now let us consider what happens if we replace each successive block of $t = \frac{1}{100\epsilon\sqrt{d}}$ invocations of the $QEX(c, \mathcal{D})$ oracle in our PAC learning algorithm with the transformation

$$Q : |0^{t(n+1)}\rangle \mapsto |\phi_t\rangle.$$

If the learning algorithm makes a total of $T$ calls to $QEX(c, \mathcal{D})$ then we perform $T/t$ replacements. After all $T/t$ calls to $Q$ in the modified algorithm, the initial state $|0 \dots 0\rangle$ evolves into the following state $|\varphi\rangle$:

$$|\varphi\rangle = \underbrace{|\phi_t\rangle \dots |\phi_t\rangle}_{T/t \text{ times}} |0 \dots 0\rangle.$$

By Equation (2), we have that $\langle \phi_T | \varphi \rangle > (1 - \frac{1}{144d})^{T/t}$. If $T/t \leq d/100$ (i.e. if $T \leq \frac{\sqrt{d}}{10000\epsilon}$), then by (1) this lower bound is at least $\frac{143}{144}$, and this implies (since the original algorithm with $T$ many $QEX(c, \mathcal{D})$ calls was successful on each target $c^i$ with probability at least 4/5) that the modified algorithm which makes at most $d/100$ many calls to $Q$ is successful with probability at least 2/3. However, the exact same polynomial-based argument which underlies the $\Omega(d/n)$ lower bound for PAC learning proved in [20] (and the improved $\frac{d}{100}$ lower bound of Observation VI.3) implies that it is impossible for our modified algorithm, which makes at most $d/100$ many calls to $Q$, to succeed on each target $c_i$ with probability at least 2/3. (The crux of that proof is that each invocation of a black-box oracle for $c$ increases the degree of the polynomial associated to the coefficient of each basis state by at most one. This property is easily seen to hold for $Q$ as well – after $r$ queries to the $Q$ oracle, the coefficient of each basis state can be expressed as a degree-$r$ polynomial in the indeterminates $c(x_1), \dots, c(x_d)$.) This proves that we must have $T/t > d/100$, which gives the conclusion of the theorem. ■

Combining our results, we obtain the following quantum version of the classical $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta} + \frac{d}{\epsilon})$ bound:

**Theorem VI.8** *Any quantum $(\epsilon, \delta)$-PAC learning algorithm for a concept class of VC dimension $d$ must make at least $\Omega(\frac{1}{\epsilon}\log\frac{1}{\delta} + d + \frac{\sqrt{d}}{\epsilon})$ calls to the $QEX$ oracle.*

## VII. FUTURE WORK

Several natural questions for future work suggest themselves. For the quantum exact learning model, is it possible to get rid of the $\log\log|C|$ factor in our algorithm's upper bound and thus prove the conjecture of Hunziker *et al.* [17] exactly? For the partitions problem, can we extend the range of partition sizes (as a function of $|C|$) for which there can be a superpolynomial separation between the quantum and classical query complexity of learning the partition? Finally, for the PAC learning model, a natural goal is to strengthen our $\Omega(\frac{\sqrt{d}}{\epsilon})$ lower bound on sample complexity to $\Omega(\frac{d}{\epsilon})$ and thus match the lower bound of [11] for classical PAC learning.

---


[1] A. Ambainis, K. Iwama, A. Kawachi, H. Masuda, R. H. Putra, S. Yamashita, *Quantum Identification of Boolean Oracles*, Proceedings of STACS 2004, pp. 93-104.

[2] D. Angluin, *Queries and Concept Learning*, Machine Learning, Vol. **2**, pp. 319–342 (1988).

[3] R. Beals, H. Buhrman, R. Cleve, M. Mosca and R. de Wolf, *Quantum Lower Bounds by Polynomials* Proceedings of the 39th IEEE Symposium on Foundation of Computer Science, pp. 352–361 (1998).

[4] C. Bennett, E. Bernstein, G. Brassard and U. Vazirani, *Strengths and weaknesses of quantum computing*, SIAM J. Comput. Vol. **26**, No 5m pp.1510–1523 (1997).

[5] E. Bernstein, U. Vazirani, *Quantum Complexity Theory*, SIAM J. Comput. Vol. **26**, No. 5, pp. 1411–1473 (1997).

[6] A. Blumer, A. Ehrenfeucht, D. Haussler and M. Warmuth, *Learnability and the Vapnik-Chervonenkis Dimension*, J. Assoc. Comput. Mach., Vol. **36**, No. 4, pp. 929–965 (1989).

[7] M. Boyer, G. Brassard, P. Høyer and A. Tapp, *Tight Bounds on Quantum Searching*, Fortschritte der Physik, Vol. **46**(4-5), pp. 493–505 (1998).

[8] N. Bshouty, R. Cleve, R. Gavaldà, S. Kannan and C. Tamon. *Oracles and queries that are sufficient for exact learning*, J. Comput. Syst. Sci., Vol **52**, No. 3 , pp. 421-433 (1996).

[9] N. H. Bshouty, J. C. Jackson, *Learning DNF over the Uniform Distribution Using a Quantum Example Oracle*, SIAM J. Comput. Vol. **28**, No. 3, pp. 1136–1153 (1999).

[10] D. Deutsch and R. Josza, *Rapid solution of problems by quantum computation*, Proc. Royal Soc. London A, Vol **439**, pp. 553–558 (1992).

[11] A. Ehrenfeucht, D. Haussler, M. Kearns, L. Valiant, *A General Lower Bound on the Number of Examples Needed for Learning*, Information and Computation **82**, pp. 247–261 (1989).
[12] E. Farhi, J. Goldstone, S. Gutmann and M. Sipser, *A Limit on the Speed of Quantum Computation in Determining Parity*, Phys. Rev. Lett., Vol. **81**, pp. 5442–5444 (1998).
[13] R. Gavaldà. *The complexity of learning with queries*, Proc. Ninth Structure in Complexity Theory Conference, pp. 324-337 (1994).
[14] L. K. Grover, *A Fast Quantum Mechanical Algorithm for Database Search*, Proceedings of the 28th ACM Symposium on Theory of Computing, pp. 212–219 (1996).
[15] T. Hegedűs. *Generalized teaching dimensions and the query complexity of learning*, Proc. Eighth Conf. on Computational Learning Theory, pp. 108-117 (1995).
[16] L. Hellerstein, K. Pillaipakkamnatt, V. Raghavan and D. Wilkins. *How many queries are needed to learn?* J. ACM, Vol. **43**, No. 5, pp. 840-862 (1996).
[17] M. Hunziker, D. A. Meyer, J. Park, J. Pommersheim and M. Rothstein, *The Geometry of Quantum Learning*, arXiv:quant-ph/0309059; to appear in Quantum Information Processing.
[18] K. Iwama, A. Kawachi, R. Raymond and S. Yamashita, *Robust Quantum Algorithms for Oracle Identification*, arXiv:quant-ph/0411204 (2005).
[19] M. Kearns and U. Vazirani, *An Introduction to Computational Learning Theory*, MIT Press, 1994.
[20] R. A. Servedio, S. J. Gortler, *Equivalences and Separations between Quantum and Classical Learnability*, SIAM J. Comput. Vol. **33**, No. 5, pp. 1067–1092 (2004).
[21] Y. Shi, *Lower Bounds of Quantum Black-Box Complexity and Degree of Approximating Polynomials by Influence of Boolean Variables* Information Processing Letters **75**, pp. 79–83 (2000).
[22] D. R. Simon, *On the Power of Quantum Computation*, SIAM J. Comput. Vol. **26**, No. 5, pp. 1474–1483 (1997).
[23] L. G. Valiant, *A Theory of the Learnable*, Comm. ACM, **27**, pp. 1134–1142 (1984).
[24] W. van Dam, *Quantum Oracle Interrogation, Getting All the Information for Almost Half the Price*, Proceedings of the 39th IEEE Symposium on Foundation of Computer Science, pp. 362–367 (1998).