HAL open science

Quantum algorithms for testing properties of distributions

Sergey Bravyi, Aram W. Harrow, Avinatan Hassidim

To cite this version:

Sergey Bravyi, Aram W. Harrow, Avinatan Hassidim. Quantum algorithms for testing properties of distributions. 27th International Symposium on Theoretical Aspects of Computer Science - STACS 2010, Inria Nancy Grand Est & Loria, Mar 2010, Nancy, France. pp.131-142. inria-00455782

HAL Id: inria-00455782

https://inria.hal.science/inria-00455782v1

Submitted on 11 Feb 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Symposium on Theoretical Aspects of Computer Science 2010 (Nancy, France), pp. 131-142
www.stacs-conf.org

QUANTUM ALGORITHMS FOR TESTING PROPERTIES OF DISTRIBUTIONS

SERGEY BRAVYI [1] AND ARAM W. HARROW [2,3] AND AVINATAN HASSIDIM [3]

[1] IBM Watson Research Center, Yorktown Heights, NY 10598 (USA).

[2] Department of Mathematics, University of Bristol, Bristol, BS8 1TW, U.K.

[3] Massachusetts Institute of Technology, Cambridge, MA 02139 (USA)

Abstract. Suppose one has access to oracles generating samples from two unknown probability distributions p and q on some N-element set. How many samples does one need to test whether the two distributions are close or far from each other in the L_1-norm? This and related questions have been extensively studied during the last years in the field of property testing. In the present paper we study quantum algorithms for testing properties of distributions. It is shown that the L_1-distance $\|p-q\|_1$ can be estimated with a constant precision using only $O(N^{1/2})$ queries in the quantum settings, whereas classical computers need $\Omega(N^{1-o(1)})$ queries. We also describe quantum algorithms for testing Uniformity and Orthogonality with query complexity $O(N^{1/3})$. The classical query complexity of these problems is known to be $\Omega(N^{1/2})$. A quantum algorithm for testing Uniformity has been recently independently discovered by Chakraborty et al [14].

1. Introduction

1.1. Problem statement and main results

Suppose one has access to a black box generating independent samples from an unknown probability distribution p on some N-element set. If the number of available samples grows linearly with N, one can use the standard Monte Carlo method to simultaneously estimate the probability p_i of every element $i = 1, \ldots, N$ and thus obtain a good approximation to the entire distribution p. On the other hand, many important questions that one usually encounters in statistical analysis can be answered using only a *sublinear* number of samples. For example, deciding whether p is close in the L_1-norm to another distribution q requires approximately $N^{1/2}$ samples if q is known [8] and approximately $N^{2/3}$ samples if q is also specified by a black-box [9]. Another example is estimating the Shannon entropy $H(p) = -\sum_i p_i \log_2 p_i$. It was shown in [7, 21] that distinguishing whether $H(p) \leq a$ or $H(p) \geq b$ requires approximately $N^{\frac{a}{b}}$ samples. Other examples include deciding whether p is close to a monotone or a unimodal distribution [10], and deciding whether a pair of distributions

1998 ACM Subject Classification: G.3 Probabilistic algorithms.
Key words and phrases: quantum computing, property testing, sampling.

SYMPOSIUM ON THEORETICAL ASPECTS OF COMPUTER SCIENCE

© S. Bravyi, A.W. Harrow, and A. Hassidim
Creative Commons Attribution-NoDerivs License

have disjoint supports [15]. These and other questions fall into the field of *distribution testing* [6, 21] that studies how many samples one needs to decide whether an unknown distribution has a certain property or is far from having this property. The purpose of the present paper is to explore whether quantum computers are capable of solving distribution testing problems more efficiently.

The black-box sampling model adopted in [8, 9, 7, 10, 6, 21] assumes that a tester is presented with a list of samples drawn from an unknown distribution. What does it mean to sample from an unknown distribution in the quantum settings? Let us start by casting the black-box sampling model into a form that admits a quantum generalization. Suppose p is an unknown distribution on an N-element set $[N] \equiv \{1, \ldots, N\}$ and let S be some specified integer. We shall assume that p is represented by an *oracle* $O_p : [S] \to [N]$ such that the probability p_i of any element $i \in [N]$ is proportional to the number of elements in the pre-image of i, that is, the number of inputs $s \in [S]$ such that $O_p(s) = i$. In other words, one can sample from p by querying the oracle O_p on a random input $s \in [S]$ drawn from the uniform distribution[1]. Note that a tester interacting with an oracle can potentially be more powerful due to the possibility of making adaptive queries which could allow him to learn the internal structure of the oracle as opposed to the black-box model. However, the unstructured nature of the problem we consider means that this advantage is restricted to avoiding repeated queries of the same position. This in turn becomes significant only when $\Omega(S)$ queries are made, which is not relevant in our setting where we have assumed that $S \gg N$. We omit the precise formulation of this claim, which is stated as Lemma 6.1 of [13].

The oracle model admits a standard quantum generalization. Specifically, we shall transform the oracle O_p into a reversible form by keeping a copy of the input and writing the output of O_p into an ancillary register. A quantum oracle generating p is a unitary operator whose action on basis vectors coincides with the reversible version of O_p, as we will explain further in Section 2.

The present paper focuses on testing three particular properties of distributions, namely, *Statistical Difference*, *Orthogonality*, and *Uniformity*. The corresponding property testing problems are promise problems so that a tester is required to give a correct answer (with a bounded error probability) only for those instances that satisfy the promise.

Problem 1.1 (Testing Uniformity).
Instance: Integers N, S, precision $\epsilon > 0$. Access to an oracle generating a distribution p on $[N]$.
Promise: Either p is the uniform distribution or the L_1-distance between p and the uniform distribution is at least ϵ.
Decide which one is the case.

Problem 1.2 (Testing Orthogonality).
Instance: Integers N, S, precision $\epsilon > 0$. Access to oracles generating distributions p, q on $[N]$.
Promise: Either p and q are orthogonal (i.e. have disjoint support) or the L_1-distance between p and q is at most $2 - \epsilon$.
Decide which one is the case.

[1]Although in this model probabilities p_i can only take values that are multiples of $1/S$, choosing sufficiently large S allows one to represent any distribution p with an arbitrarily small error.

Problem 1.3 (Testing Statistical Difference).
Instance: Integers N, S, thresholds $0 \leq a < b \leq 2$. Access to oracles generating distributions p and q on $[N]$.
Promise: Either $\|p - q\|_1 \leq a$ or $\|p - q\|_1 \geq b$.
Decide which one is the case.

We assume that the precision ϵ is bounded from below by a fixed constant independent of N, for instance, $\epsilon \geq 1/10$. The same applies to the decision gap $b-a$ for testing Statistical Difference. Given a function $f(N)$ we shall say that a property is testable in $f(N)$ queries if there exists a testing algorithm making at most $f(N)$ queries that gives a correct answer with a sufficiently high probability (say 2/3) for any distributions p, q satisfying the promise and for any oracles[2] specifying p and q. If a promise is violated, a tester can give an arbitrary answer.

Our main results are the following theorems.

Theorem 1.4. *Statistical Difference is testable on a quantum computer in $O(N^{1/2})$ queries.*

Theorem 1.5. *Uniformity is testable on a quantum computer in $O(N^{1/3})$ queries.*

Theorem 1.6. *Orthogonality is testable on a quantum computer in $O(N^{1/3})$ queries.*

It is known that classically testing Orthogonality and Uniformity requires $\Omega(N^{1/2})$ queries, see Sections 6.1 and 6.2, while Statistical Difference is not testable in $O(N^{\alpha})$ queries for any $\alpha < 1$, see [21]. Therefore quantum computers provide a polynomial speedup for testing Uniformity, Orthogonality, and Statistical Difference in terms of query complexity.

Testing Orthogonality is closely related to the Collision Problem studied in [12]. In Section 6.1 we describe a randomized reduction from the Collision Problem to testing Orthogonality. Using the quantum lower bound for the Collision Problem due to Aaronson and Shi [2] we obtain the following result.

Theorem 1.7. *Testing Orthogonality on a quantum computer requires $\Omega(N^{1/3})$ queries.*

Quite recently Chakraborty, Fischer, Matsliah, and de Wolf [14] independently discovered a quantum Uniformity testing algorithm with query complexity $O(N^{1/3})$ and proved a lower bound $\Omega(N^{1/3})$ for testing Uniformity. These authors also presented a quantum algorithm for testing whether an unknown distribution p coincides with a known distribution q with query complexity $\tilde{O}(N^{1/3})$.

1.2. Discussion and open problems

One motivation for studying distribution testing problems is that testing Orthogonality and Statistical Difference are complete problems for the complexity class SZK (Statistical Zero Knowledge). More precisely, the following problem known as *Statistical Difference* was shown to be SZK-complete by Vadhan [18]:

Input: *description of classical circuits C_p, C_q that implement oracle functions O_p, O_q : $[S] \to [N]$ and a pair of real numbers $0 \leq a < b \leq 2$ such that $2a \leq b^2$.*
Problem: *Decide whether $\|p - q\|_1 \geq b$ (yes-instance) or $\|p - q\|_1 \leq a$ (no-instance)* .

The class SZK includes many interesting algebraic and graph theoretic problems such as Discrete Logarithm, Graph Isomorphism, Graph NonIsomorphism, Quadratic Residuosity,

[2]Note that according to this definition a tester needs at most $f(N)$ queries even in the limit $S \to \infty$.

and The Shortest Vector in Lattice, see [3] and references therein. Thus it is natural to ask whether quantum computers provide a universal speedup for problems in SZK similar to the square-root speedup for problems in NP provided by the Grover search algorithm. Assuming that the circuits C_p, C_q have size $poly(\log(N))$, one can easily translate the testing algorithm described in Section 3 to a quantum circuit of size $\tilde{O}(\sqrt{N})$ solving Statistical Difference problem for any constants a, b as above. On the other hand, any classical algorithm treating the circuits C_p, C_q as black boxes would need roughly $N^{1-o(1)}$ queries, see [21], thus requiring a circuit of size $\Omega(N^{1-o(1)})$.

Note that the Statistical Difference problem with $b = 2$ is equivalent to testing Orthogonality. It can be solved classically in time $\tilde{O}(N^{1/2})$ using the classical collision finding algorithm. Unfortunately, the circuit complexity of the quantum Orthogonality testing algorithm described in Section 5 may be different from its query complexity since it uses a quantum membership oracle for a randomly generated set. It is an open problem whether Statistical Difference problem with $b = 2$ can be solved by a quantum circuit of size $\tilde{O}(N^{1/3})$, although with a suitably powerful model of quantum RAM, such membership queries can be done in time $\operatorname{poly}\log(N)$. A related question is that of space-time tradeoffs: our algorithms generally require storing $N^{O(1)}$ classical bits and then querying them with quantum algorithms that use $\operatorname{poly}(\log(N)$ qubits. We suspect that this amount of storage cannot be reduced without increasing the run-time, but do not have a proof of this conjecture. Similar issues of quantum data structures for set membership and conjectured space-time tradeoffs have arisen for the element distinctness problem[5, 16].

It is worth mentioning that all distribution properties studied in this paper are *symmetric*, that is, these properties are invariant under relabeling of elements in the underlying set $\{1, \ldots, N\}$. Testing symmetric properties of distributions is equivalent to testing properties of functions from $[S]$ to $[N]$ that are invariant under any permutations of inputs and outputs of the function. It was recently shown by Aaronson and Ambainis that quantum computers can provide at most polynomial speedup for testing properties of such symmetric functions [1].

More interesting than the mere fact of polynomial speedups provided by Theorems 1.4, 1.5, 1.6 is the way in which our algorithms achieve it. Classically, the results of Ref. [21] provide a simple characterization of an asymptotically optimal testing algorithm for any symmetric property of a distribution (satisfying certain natural continuity conditions). By contrast, our algorithms use a variety of different strategies both to query the oracles and to analyze the results of those queries. These strategies appear not to be special cases of the quantum walk framework which has been responsible for most of the polynomial quantum speedups found to date [20, 19]. A major challenge for future research is to give a quantum version of Ref. [21]'s Canonical Tester algorithm; in other words, we would like to characterize optimal quantum algorithms for testing any symmetric property of a distribution (or a pair of distributions).

Finally, let us remark that the algorithm for estimating statistical difference described in Section 3 can be easily generalized to construct a quantum algorithm for estimating the von Neumann entropy of a black-box distribution with query complexity $\tilde{O}(N^{1/2})$. Using similar ideas one can construct an $\tilde{O}(N^{1/2})$-time algorithm for estimating the fidelity between two black-box distributions (i.e. $\sum_{i=1}^{N} \sqrt{p_i q_i}$).

The rest of the paper is organized as follows. Section 2 introduces necessary notations and basic facts about the quantum counting algorithm by Brassard, Hoyer, Mosca, and

Tapp [11]. The distribution testing algorithms described in the rest of the paper are actually classical probabilistic algorithms using the quantum counting as a subroutine. Theorem 1.4 is proved in Section 3. Theorem 1.5 is proved in Section 4. Theorem 1.6 is proved in Section 5. We discuss lower bounds for the above distribution testing problems in Section 6.

2. Preliminaries

Let $\mathcal{D}_N$ be the set of probability distributions $p = (p_1, \ldots, p_N)$ such that a probability p_i of any element $i \in [N]$ is a rational number. Let us say that an oracle $O : [S] \to [N]$ generates a distribution $p \in \mathcal{D}_N$ iff for all $i \in [N]$ the probability p_i equals the fraction of inputs $s \in [S]$ such that $O(s) = i$,

$$p_i = \frac{1}{S} \#\{s \in [S] : O(s) = i\}.$$

Note that the identity of elements in the domain of an oracle O is irrelevant, so if O generates p and σ is any permutation on $[S]$ then $O \circ \sigma$ also generates p. By definition, any map $O : [S] \to [N]$ generates some distribution $p \in \mathcal{D}_N$.

For any oracle $O : [S] \to [N]$ we shall define a quantum oracle $\hat{O}$ by transforming O into a reversible form and allowing it to accept coherent superpositions of queries. Specifically, a quantum oracle $\hat{O}$ is a unitary operator acting on a Hilbert space $\mathbb{C}^S \otimes \mathbb{C}^{N+1}$ equipped with a standard basis $\{|s\rangle \otimes |i\rangle\}$, $s \in [S]$, $i \in \{0\} \cup [N]$ such that

$$\hat{O}\,|s\rangle \otimes |0\rangle = |s\rangle \otimes |O(s)\rangle \quad \text{for all } s \in [S]. \tag{2.1}$$

In other words, querying $\hat{O}$ on a basis vector $|s\rangle \otimes |0\rangle$ one gets the output of the classical oracle $O(s)$ in the second register while the first register keeps a copy of s to maintain unitarity. The action of $\hat{O}$ on a subspace in which the second register is orthogonal to the state $|0\rangle$ can be arbitrary. We shall assume that a quantum tester can execute operators $\hat{O}$, $\hat{O}^\dagger$ and the controlled versions of them. Execution of any one of these operators counts as one query.

Another apparently natural quantum model of a probability distribution is the ability to prepare the state $\sum_{i=1}^N \sqrt{p_i}|i\rangle$; i.e. the ability to "$q$-sample" from the distribution p, c.f. Ref. [3]. However, this ability turns out to be far stronger than the oracle model we will use, since it would allow us to solve Problems 1, 2 and 3 with $O(1)$ q-samples of the distributions p and q. This follows from the well-known result that the observable $\textsc{swap} = \sum_{i,j=1}^N |i,j\rangle\langle j,i|$ has expectation value $|\langle p|q\rangle|^2$ when measured on the state $(\sum_{i=1}^N \sqrt{p_i}|i\rangle) \otimes (\sum_{j=1}^N \sqrt{q_j}|j\rangle)$. Moreover, the ability to efficiently classically sample from a distribution p implies the ability to efficiently construct a quantum oracle $\hat{O}$ corresponding to p, but does not generally imply the ability to q-sample from p. Accordingly, in the rest of the paper we will consider probability distributions to be encoded in quantum oracles.

We shall see that all testing problems posed in Section 1 can be reduced (via classical randomized reductions) to the following problem.

Problem 2.1 (**Probability Estimation**).
<u>Instance:</u> Integers S, N, description of a subset $A \subset [N]$, precision δ, error probability ω, and access to an oracle generating some distribution $p \in \mathcal{D}_N$. Let $p_A = \sum_{i \in A} p_i$ be the total probability of A.

Task: Generate an estimate $\tilde{p}_A$ satisfying

$$\Pr\left[|\tilde{p}_A - p_A| \le \delta\right] \ge 1 - \omega. \tag{2.2}$$

Our main technical tool will be the quantum counting algorithm by Brassard et al. [11]. Specifically, we shall use the following version of Theorem 12 from [11], whose precise form is proved in [13].

Theorem 2.2. *There exists a quantum algorithm* $\mathbf{EstProb}(p, A, M)$ *taking as input a distribution* $p \in \mathcal{D}_N$ *specified by an oracle, a subset* $A \subset [N]$*, and an integer* M*. The algorithm makes exactly* M *queries to the oracle generating* p *and outputs an estimate* $\tilde{p}_A$ *such that*

$$\Pr\left[|\tilde{p}_A - p_A| \le \delta\right] \ge 1 - \omega \tag{2.3}$$

for all $\delta > 0$ *and* $0 \le \omega \le 1/2$ *satisfying*

$$M \ge \frac{c\sqrt{p_A}}{\omega\delta} \quad \text{and} \quad M \ge \frac{c}{\omega\sqrt{\delta}}. \tag{2.4}$$

Here $c = O(1)$ *is some constant. If* $p_A = 0$ *then* $\tilde{p}_A = 0$ *with certainty.*

(In Eq. 2.4, is is possible to replace $1/\omega$ with $\log(1/\omega)$, but we will not need this improvement.)

3. Quantum algorithm for estimating statistical difference

In this section we sketch the proof of Theorem 1.4. Let $p, q \in \mathcal{D}_N$ be unknown distributions specified by oracles. Define an auxiliary distribution $r \in \mathcal{D}_N$ such that $r_i = (p_i + q_i)/2$ for all $i \in [N]$. If we can sample i from both p and q then by choosing randomly between these two options we can also sample i from r. Let $x \in [0, 1]$ be a random variable which takes value

$$x_i = \frac{|p_i - q_i|}{p_i + q_i}$$

with probability r_i. It is evident that

$$\mathbb{E}(x) = \sum_{i\in[N]} r_i x_i = \frac{1}{2}\sum_{i\in[N]} |p_i - q_i| = \frac{1}{2}\|p - q\|_1. \tag{3.1}$$

Thus in order to estimate the distance $\|p - q\|_1$ it suffices to estimate the expectation value $\mathbb{E}(x)$ which can be done using the standard Monte Carlo method. Since we have to estimate $\mathbb{E}(x)$ only with a constant precision, it suffices to generate $O(1)$ samples of x_i. Given a sample of i (which is easy to generate classically) we can estimate x_i by calling the probability estimation algorithm to get estimates of p_i and q_i. Based on this intuition, we propose the following algorithm for estimating the distance $\|p - q\|_1$.

EstDist(p, q, ϵ, τ)
Set $n = 27/\tau\epsilon^2$, $M = c\sqrt{N}/\epsilon^6\tau^4$.
Let $i_1, \ldots, i_n \in [N]$ be a list of n independent samples drawn from r.
For $a = 1, \ldots, n$
{
  Let $\tilde{p}_{i_a}$ be an estimate of p_{i_a} obtained using **EstProb**$(p, \{i_a\}, M)$.
  Let $\tilde{q}_{i_a}$ be an estimate of q_{i_a} obtained using **EstProb**$(q, \{i_a\}, M)$.
  Let $\tilde{x}_{i_a} = |\tilde{p}_{i_a} - \tilde{q}_{i_a}|/(\tilde{p}_{i_a} + \tilde{q}_{i_a})$ be our estimate of x_{i_a}.
}
Output $\tilde{x} = (1/n)\sum_{a=1}^{n} \tilde{x}_{i_a}$.

Here $c = O(1)$ is a constant whose precise value will not be important for us.

Lemma 3.1. *The algorithm* **EstDist**(p, q, ϵ, τ) *outputs an estimate* $\tilde{x}$ *satisfying*

$$\Pr\left[|\tilde{x} - \mathbb{E}(x)| < \epsilon\right] \geq 1 - \tau, \tag{3.2}$$

where $\mathbb{E}(x) = (1/2)\|p - q\|_1$.

The proof can be found in Ref. [13] and is omitted from this extended abstract. The rough idea is that we define an element i to be *bad* iff $\max(p_i, q_i) \leq \tau/3nN$. Then the total probability that any element is bad is $\leq \tau/3$. Conditioned on all the elements being good, we can use Theorem 2.2 to show that we can estimate each p_i and q_i up to multiplicative error $1 - o(1)$, and thereby can also get good estimates of x_i.

Theorem 1.4 follows directly from Lemma 3.1 since **EstDist**(p, q, ϵ, τ) makes $O(\sqrt{N})$ queries to the quantum oracles generating p and q.

4. Quantum algorithm for testing Uniformity

In this section we sketch the proof of Theorem 1.5. Let $p \in \mathcal{D}_N$ be an unknown distribution specified by an oracle. We are promised that either p is the uniform distribution, or p is ϵ-*nonuniform*, that is, the L_1-distance between p and the uniform distribution is at least ϵ. The algorithm described below is based on the following simple observation. Choose some integer $M \ll N$ and let $S = (i_1, \ldots, i_M)$ be a list of M independent samples drawn from the distribution p. Define a random variable $p_S = \sum_{a=1}^{M} p_{i_a}$. It coincides with the total probability of all elements in S unless S contains a collision (that is, $i_a = i_b$ for some $a \neq b$). The characteristic property of the uniform distribution is that $p_S = M/N$ with certainty. On the other hand, we shall see that for any ϵ-nonuniform distribution p_S takes values greater than $(1 + \delta)M/N$ for some constant $\delta > 0$ depending on ϵ with a non-negligible probability. This observation suggests the following algorithm for testing uniformity (the constants K and M below will be chosen later).

UTest(p, K, M, ϵ)

- Let $S = (i_1, \ldots, i_M)$ be a list of M independent samples drawn from p.
- Reject unless all elements in S are distinct.
- Let $p_S = \sum_{a=1}^{M} p_{i_a}$ be the total probability of elements in S.
- Let $\tilde{p}_S$ be an estimate of p_S obtained using **EstProb**(p, S, K).
- If $\tilde{p}_S > (1 + \epsilon^2/8)M/N$ then reject. Otherwise accept.

This procedure will need to be repeated several times to achieve the desired bound on the error probability, as we will discuss below.

The main technical result needed is the following lemma.

Lemma 4.1. *Let $p \in \mathcal{D}_N$ be an ϵ-nonuniform distribution. Let $S = (i_1, \ldots, i_M)$ be a list of M independent samples drawn from p, where*

$$M = \left(\frac{32N}{\epsilon^4}\right)^{\frac{1}{3}}. \tag{4.1}$$

Let $p_S = \sum_{a=1}^{M} p_{i_a}$ and $\alpha = 2^8\epsilon^{-4}$. Then

$$\Pr\left[p_S \geq (1+\epsilon^2/2)\frac{M}{N}\right] \geq \frac{1}{2}\exp(-\alpha). \tag{4.2}$$

Theorem 1.4 follows straightforwardly from the above lemma and Theorem 2.2.

Proof of Theorem 1.4. Let M be chosen as in Eq. (4.1) and

$$K = c\frac{e^{\alpha}N^{1/3}}{\epsilon^{4/3}},$$

where $c = O(1)$ is a constant to be chosen later. Consider the following algorithm:

Perform $L = 4\exp(\alpha)$ independent tests **UTest**(p, K, M, ϵ). If at least one of the tests outputs 'reject' then reject. Otherwise accept.

In the full version of this paper [13], we prove that this algorithm rejects any ϵ-nonuniform distribution with probability at least 2/3 and accepts the uniform distribution with probability at least 2/3.

■

In the rest of this section we sketch the proof of Lemma 4.1 again deferring full proofs to [13]. We shall adopt notations introduced in the statement of Lemma 4.1, that is, the number of samples M is defined by

$$M^3 = 32\epsilon^{-4}N,$$

$\alpha \equiv 2^8\epsilon^{-4}$, $S = (i_1, \ldots, i_M)$ is a list of M independent samples drawn from p, and $p_S = \sum_{a=1}^{M} p_{i_a}$.

Definition 4.2. An element $i \in [N]$ is called big iff $p_i > 1/(2M^2)$.

Define the set Big $\subset [N]$ of all big elements and their total probability:

$$\mathrm{Big} = \{i \in [N] : p_i > 1/(2M^2)\}, \quad w_{\mathrm{big}} = \sum_{i \in \mathrm{Big}} p_i. \tag{4.3}$$

Also, observe that

$$\mathbb{E}(p_S) = M\langle p|p\rangle \quad \text{and} \tag{4.4a}$$

$$\mathrm{Var}(p_S) = M\left(\sum_{i=1}^{N} p_i^3 - \langle p|p\rangle^2\right). \tag{4.4b}$$

The proof of Lemma 4.1 is divided into three cases. We shall start by proving the Lemma in the special case when $p \in \mathcal{D}_N$ is ϵ-nonuniform and has no big elements. Using

(4.4), we find that the ϵ-nonuniformity of p implies that $\mathbb{E}(p_S) \geq \frac{M}{N}(1+\epsilon^2)$ while the lack of big elements implies that $\mathrm{Var}(p_S) \leq \langle p|p\rangle/2M$. Then we use Chebyshev's inequality to argue that p_S is likely to be larger than $\frac{M}{N}(1+\epsilon^2/2)$. The second case is when the total weight of big elements is $\leq \alpha/M$, for $\alpha \equiv \epsilon^{-4}/256$. In this case, our sampling is unlikely to encounter any big elements and we can reduce the proof to the case when there are no big elements. Finally, if the total weight of big elements is $> \alpha/M$, then there is a substantial probability that we sample $> \alpha/2$ of them, which will result in p_S being larger than $2M/N$.

5. Quantum algorithm for testing orthogonality

Consider distributions $p, q \in \mathcal{D}_N$ and let $S = (i_1, \ldots, i_M)$ be a list of M independent samples drawn from p. Let $A \subseteq [N]$ be the set of all elements that appear in S at least once. Define the *collision probability*

$$q_A = \sum_{i \in A} q_i.$$

Note that q_A is a deterministic function of A, so the probability distribution of q_A is determined by the probability distribution of A (which depends on p and M). For a fixed A the variable q_A is the probability that a sample drawn from q belongs to A.

Clearly if p and q are orthogonal then $q_A = 0$ with probability 1. On the other hand, if p and q have a constant overlap, we will show that q_A takes values of order M/N with constant probability. Specifically, we shall prove the following lemma.

Lemma 5.1. *Consider a pair of distributions $p, q \in \mathcal{D}_N$ such that $\|p-q\|_1 \leq 2-\epsilon$. Let q_A be a collision probability constructed using M samples. Suppose $M \geq 2^9\epsilon^{-2}$. Then*

$$\Pr\left[q_A \geq \frac{\epsilon^3 M}{2^{11}N}\right] \geq \frac{1}{2}. \tag{5.1}$$

This Lemma suggests the following algorithm for testing orthogonality.

OTest(p, q, M, K)

- Let $S = \{i_1, \ldots, i_M\}$ be a list of M independent samples drawn from p.
- Let $A \subseteq [N]$ be the set of elements that appear in S at least once.
- Let $q_A = \sum_{i \in A} q_i$ be the total probability of elements in A with respect to q.
- Let $\tilde{q}_A$ be estimate of q_A obtained using **EstProb**(q, A, K).
- If $\tilde{q}_A \geq \frac{\epsilon^3 M}{2^{12}N}$ then reject. Otherwise accept.

We note that if $q_A = 0$ then $\tilde{q}_A = 0$ with certainty (see Theorem 2.2) and so **OTest** accepts any pair of orthogonal distributions with certainty. Again the full proof of Theorem 1.6 is left to [13]. The idea it to choose $M = K = O\left(\frac{N^{1/3}}{\epsilon}\right)$ and apply **OTest**(p, q, M, K) to distributions $p, q \in \mathcal{D}_N$. According to Lemma 5.1, if $\|p-q\|_1 \leq 2-\epsilon$ then $q_A \geq \epsilon^3 M/(2^{11}N)$ with probability $\geq 1/2$. When this holds, the algorithm rejects whenever $|\tilde{q}_A - q_A| \leq \frac{q_A}{2}$ since this implies $\tilde{q}_A \geq q_A/2 \geq \epsilon^3 M/(2^{12}N)$. By Theorem 2.2, our choice of K is sufficient to achieve this with $\Omega(1)$ probability.

It remains only to prove Lemma 5.1.

Proof. Begin by defining two sets of indices:

$$B \equiv \{i : q_i < \frac{\epsilon}{4} p_i\} \qquad \text{and} \qquad C \equiv \{i : p_i \leq \frac{\epsilon}{32} N^{-1}\} \tag{5.2}$$

Let B^c, C^c denote the complements of B and C respectively. We will prove that

$$\Pr\left[|A \cap B^c \cap C^c| \geq \frac{\epsilon}{16} M\right] \geq 1/2, \tag{5.3}$$

which will imply the Lemma since

$$q_A \geq \sum_{i \in A \cap B^c \cap C^c} q_i \geq \frac{\epsilon}{4} \sum_{i \in A \cap B^c \cap C^c} p_i \geq \frac{\epsilon^2}{2^7 N} |A \cap B^c \cap C^c|.$$

This is achieved by using a Chernoff-Hoeffding bound to show that $|A \cap B|$ and $|A \cap C$ are each unlikely to be much larger than their expectations. The details are in [13]. ∎

6. Lower bounds

6.1. Reduction from the Collision Problem to testing Orthogonality

One can get lower bounds on the query complexity of testing Orthogonality using the lower bounds for the Collision problem [2]. Indeed, let $H : [N] \to [N]$ be an oracle function such that either H is one-to-one (yes-instance) or H is two-to-one (no-instance). The Collision Problem is to decide which one is the case. It was shown by Refs. [2, 4, 17] that the quantum query complexity of the Collision problem is $\Omega(N^{1/3})$. Below we show that the Collision problem can be reduced to testing Orthogonality. As a result, testing Orthogonality will be shown to require $\Omega(N^{1/2})$ queries classically and $\Omega(N^{1/3})$ queries quantumly.

Indeed, choose a random permutation $\sigma : [N] \to [N]$ and define functions $O_p, O_q : [N/2] \to [3N/2]$ by restricting the composition $H \circ \sigma$ to the subsets of odd and even integers respectively:

$$O_p(s) = H(\sigma(2s-1)), \quad O_q(s) = H(\sigma(2s))$$

where $s \in [N/2]$.

For any yes-instance (i.e. H is one-to-one), the distributions $p, q \in \mathcal{D}_{3N/2}$ generated by O_p and O_q are uniform distributions on some pair of disjoint subsets of $[3N/2]$; that is, p and q are orthogonal.

We need to show that for any no-instance (H is two-to-one) the distance $\|p - q\|_1$ takes values smaller than $2 - \epsilon$ with a sufficiently high probability for some constant ϵ. This is established by the following Lemma, whose proof can be found in [13].

Lemma 6.1. *Let $H : [N] \to [3N/2]$ be any two-to-one function. Let $\sigma : [N] \to [N]$ be a random permutation drawn from the uniform distribution. Then*

$$\Pr\left[\|p - q\|_1 \leq \frac{7}{4}\right] \geq \frac{1}{2}.$$

6.2. Classical lower bound for testing Uniformity

In this section we prove that classically testing Uniformity requires $\Omega(N^{1/2})$. A proof uses the machinery developed by Valiant in [21]. Valiant's techniques apply to testing *symmetric* properties of distributions, that is, properties that are invariant under relabeling of elements in the domain of a distribution. Clearly, Uniformity is a symmetric property.

We shall need two technical tools from [21], namely, the Positive-Negative Distance lemma and Wishful Thinking theorem (see Theorem 4 and Lemma 3 in [21]). Let us start from introducing some notations. Let $p \in \mathcal{D}_N$ be an unknown distribution and $S = (i_1, \ldots, i_M)$ be a list of M independent samples drawn from p. We shall say that S has a collision of order r iff some element $i \in [N]$ appears in S exactly r times. Let c_r be the total number of collisions of order r, where $r \geq 1$. A sequence of integers $\{c_r\}_{r\geq 1}$ is called a *fingerprint* of S. Define a probability distribution D_p^M on a set of fingerprints as follows: (1) draw k from the Poisson distribution $\mathrm{Poi}(k) = e^{-M} M^k/k!$. (2) Generate a list S of k independent samples drawn from p. (3) Output a fingerprint of S.

An important observation made in [21] is that a fingerprint contains all relevant information about a sample list as far as testing symmetric properties is concerned. Thus without loss of generality, a testing algorithm has to make its decision by looking only on a fingerprint of a sample list. Applying Positive-Negative Distance lemma from [21] to testing Uniformity we get the following result.

Lemma 6.2 ([21])**.** *Let u be the uniform distribution on $[N]$ and $p \in \mathcal{D}_N$ be any distribution such that $\|p - u\|_1 \geq 1$. If for some integer M*

$$\|D_p^M - D_u^M\|_1 < \frac{1}{12} \tag{6.1}$$

then Uniformity is not testable in M samples.

The second technical tool is a usable upper bound on the distance between the distributions of fingerprints. For any integer k define an k-th moment of p as $m_k(p) = \sum_{i=1}^N p_i^k$. Clearly $m_k(u) = N^{1-k}$ which is the smallest possible value of a k-th moment for distributions on $[N]$. Applying Wishful Thinking theorem from [21] to testing Uniformity we get the following result (again proved in [13]).

Lemma 6.3 ([21])**.** *Let $p \in \mathcal{D}_N$ be any distribution such that $\|p\|_\infty \leq \delta/M$ for some $\delta > 0$. Then*

$$\|D_p^M - D_u^M\|_1 \leq 40\delta + 10 \sum_{k\geq 2} M^k \frac{m_k(p) - N^{1-k}}{\lfloor k/2 \rfloor! \sqrt{1 + M^k m_k(p)}}. \tag{6.2}$$

Corollary 6.4. *Uniformity is not testable classically in $32^{-1} N^{1/2}$ queries.*

Acknowledgments

We are grateful to Ronald de Wolf for numerous comments that helped to improve the paper. We would like to thank Sourav Chakraborty for informing us about the results in [14]. S.B. thanks CWI for hospitality while this work was being done and was funded by the DARPA QUEST program under contract no. HR0011-09-C-0047. A.W.H. is grateful to IBM and MIT for their hospitality while this work was being done, and is funded by the U.K. EPRSC grant "QIP IRC" and the QAP project (contract IST-2005-15848). A.H. was supported by an xQIT Keck fellowship.

References

[1] S. Aaronson and A. Ambainis. The need of structure in quantum speedups, 2009. arXiv:0911.0996.

[2] S. Aaronson and Y. Shi. Quantum lower bounds for the collision and the element distinctness problems. *J. ACM*, 51(4):595–605, 2004. arXiv:quant-ph/0112086.

[3] D. Aharonov and A. Ta-Shma. Adiabatic quantum state generation and statistical zero knowledge. In *Proceedings of the 35th Annual ACM Symposium on Theory of computing (STOC)*, pages 20–29. ACM Press New York, NY, USA, 2003. arXiv:quant-ph/0301023.

[4] A. Ambainis. Polynomial degree and lower bounds in quantum complexity: Collision and element distinctness with small range. *Theory of Computing*, 1:37–46, 2005. arXiv:quant-ph/0305179.

[5] A. Ambainis. Quantum walk algorithm for element distinctness. *SIAM J. Comput.*, 37(1):210–239, 2007. arXiv:quant-ph/0311001.

[6] T. Batu. *Testing properties of distributions*. PhD thesis, Cornell University, 2001.

[7] T. Batu, S. Dasgupta, R. Kumar, and R. Rubinfeld. The complexity of approximating the entropy. *SIAM J. Comput.*, 35(1):132–150, 2005.

[8] T. Batu, L. Fortnow, E. Fischer, R. Kumar, R. Rubinfeld, and P. White. Testing random variables for independence and identity. In *FOCS '01: Proceedings of the 42nd IEEE symposium on Foundations of Computer Science*, page 442, Washington, DC, USA, 2001. IEEE Computer Society.

[9] T. Batu, L. Fortnow, R. Rubinfeld, W. D. Smith, and P. White. Testing that distributions are close. In *FOCS '00: Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, page 259, Washington, DC, USA, 2000. IEEE Computer Society.

[10] T. Batu, R. Kumar, and R. Rubinfeld. Sublinear algorithms for testing monotone and unimodal distributions. In *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 381–390, New York, NY, USA, 2004. ACM.

[11] G. Brassard, P. Høyer, M. Mosca, and A. Tapp. Quantum amplitude amplification and estimation. In S. J. Lomonaco, editor, *Quantum Computation & Information*, volume 305 of *Contemporary Mathematics Series Millenium Volume*, pages 53–74. AMS, 2002. arXiv:quant-ph/0005055.

[12] G. Brassard, P. Høyer, and A. Tapp. Quantum algorithm for the collision problem. *ACM SIGACT News*, 28:14–19, 1997. arXiv:quant-ph/9705002.

[13] S. Bravyi, A. Harrow, and A. Hassidim. Quantum algorithms for testing properties of distributions, 2009. arXiv:0907.3920.

[14] S. Chakraborty, E. Fischer, A. Matsliah, , and R. de Wolf. Quantum Queries for Testing Distributions, 2009. unpublished.

[15] O. Goldreich and D. Ron. A sublinear bipartiteness tester for bounded degree graphs. In *STOC '98: Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 289–298, New York, NY, USA, 1998. ACM.

[16] L. Grover and T. Rudolph. How significant are the known collision and element distinctness quantum algorithms? *Quant. Inf. & Comp.*, 4:201–206, 2004. arXiv:quant-ph/0309123.

[17] S. Kutin. A quantum lower bound for the collision problem. *Theory of Computing*, 1:29–36, 2005. arXiv:quant-ph/0304162.

[18] A. Sahai and S. Vadhan. A complete promise problem for statistical zero-knowledge. In *FOCS '97: Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, page 448, Washington, DC, USA, 1997. IEEE Computer Society.

[19] M. Santha. Quantum walk based search algorithms. In *TAMC*, volume 4978 of *Lecture Notes in Computer Science*, pages 31–46. Springer, 2008. arXiv:0808.0059.

[20] M. Szegedy. Quantum Speed-Up of Markov-Chain-Based Algorithms. In *FOCS '04: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 32–41, Washington, DC, USA, 2004. IEEE Computer Society.

[21] P. Valiant. Testing symmetric properties of distributions. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 383–392, New York, NY, USA, 2008. ACM.

This work is licensed under the Creative Commons Attribution-NoDerivs License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nd/3.0/.