Uniformity Testing When You Have the Source Code

Clément L. Canonne
The University of Sydney, Australia

Robin Kothari
Google Quantum AI, Santa Barbara, CA, USA

Ryan O'Donnell
Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

We study quantum algorithms for verifying properties of the output probability distribution of a classical or quantum circuit, given access to the source code that generates the distribution. We consider the basic task of uniformity testing, which is to decide if the output distribution is uniform on $[d]$ or ε-far from uniform in total variation distance. More generally, we consider identity testing, which is the task of deciding if the output distribution equals a known hypothesis distribution, or is ε-far from it. For both problems, the previous best known upper bound was $O(\min\{d^{1/3}/\varepsilon^2, d^{1/2}/\varepsilon\})$. Here we improve the upper bound to $O(\min\{d^{1/3}/\varepsilon^{4/3}, d^{1/2}/\varepsilon\})$, which we conjecture is optimal.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms; Theory of computation → Quantum computation theory

Keywords and phrases distribution testing, uniformity testing, quantum algorithms

Digital Object Identifier 10.4230/LIPIcs.TQC.2025.7

Related Version *Full Version*: `https://arxiv.org/abs/2411.04972`

Funding *Clément L. Canonne*: Supported by an ARC DECRA (DE230101329).
Ryan O'Donnell: Supported by ARO grant W911NF2110001 and by a gift from Google Quantum AI.

1 Introduction

For 30 years we have known that quantum computers can solve certain problems significantly faster than any known classical algorithm. Traditionally, most of the research in this area has focused on decision problems (like SAT) or function problems (like Factoring), where for each possible input there is a unique "correct" output. However, we have also found that quantum computers can yield speedups for the task of *sampling* from certain probability distributions. Prominent examples include boson sampling [1] and random circuit sampling [8]. Sampling tasks have seemed more natural for NISQ-era quantum computation, and indeed many of the first candidate experimental demonstrations of quantum advantage have been for sampling problems [6].

One of the downsides of sampling problems is the challenge of *verifying* the output of an algorithm, whether classical or quantum, that claims to sample from a certain distribution. As a simple example, consider a classical or quantum algorithm that implements a supposed hash function with output alphabet $[d] := \{1, \ldots, d\}$. The algorithm designer claims that the output distribution of this hash function is uniform on $[d]$. If $\mathbf{p}$ denotes the actual output distribution of the algorithm, and $\mathbf{u}_d$ denotes the uniform distribution on $[d]$, then we would like to test whether $\mathbf{p} = \mathbf{u}_d$, and reject the claim if $\mathbf{p}$ is in fact ε-far from $\mathbf{u}_d$ in total variation distance, meaning $\frac{1}{2}\|\mathbf{p} - \mathbf{u}_d\|_1 > \varepsilon$. (We will also consider other distance measures in this work, since the complexity of the testing task is sensitive to this choice.)

20th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2025).
Editor: Bill Fefferman; Article No. 7; pp. 7:1–7:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

This verification task is called “uniformity testing” (in total variation distance) and its complexity is well studied in the classical literature. If we only have access to samples from $\mathbf{p}$, but are not allowed to inspect the algorithm that produces these samples, it is known that $\Theta(d^{1/2}/\varepsilon^2)$ samples are necessary and sufficient to solve this problem; there are various classical algorithms that achieve this bound (starting with that of [28]; see, e.g., [11] for a detailed survey and discussion), and it is also not possible to do better with a quantum algorithm. But what if – as in the examples above – we *do* have access to the algorithm that produces $\mathbf{p}$? Can we improve on this complexity if we have access to the “source code” of the algorithm?

Having the source code

To clarify, the “source code” for a classical randomized sampling algorithm means a randomized circuit (with no input) whose output is one draw from $\mathbf{p}$. More generally, the “source code” for a quantum sampling algorithm means a unitary quantum circuit (with all input qubits fixed to $|0\rangle$) which gives one draw from $\mathbf{p}$ when some of its output bits are measured in the standard basis and the rest are discarded.[1] The simplest way to use the code C for $\mathbf{p}$ is to run it, obtaining one sample. If C has size S, then getting one sample this way has cost S. Another way to use the code C is to deterministically compute all its output probabilities; this gives one perfect information about $\mathbf{p}$, but has cost bound 2^S. But quantum computing has suggested a third way to use the code: “running it in reverse”. For example, Grover’s original algorithm [18] can be seen as distinguishing two possibilities for $\mathbf{p}$ on [2], namely $\mathbf{p}_1 = 0$ or $\mathbf{p}_1 = 1/N$, while using only $O(N^{1/2})$ forwards/backwards executions of C. The total cost here is $O(N^{1/2}) \cdot S$, the same as the cost for $O(N^{1/2})$ samples.

We suggest that the utility of “having the source code” for distribution testing problems remains notably underexplored. Indeed, there is significant room for improvment in the bounds for even the most canonical of all such problems: uniformity testing. Our main theorem is the following:

▶ **Theorem 1.** *There is a computationally efficient quantum algorithm for uniformity testing with the following guarantees: given $\varepsilon \geq 1/\sqrt{d}$, the algorithm makes $O(d^{1/3}/\varepsilon^{4/3})$ uses of “the code” for an unknown distribution $\mathbf{p}$ over $[d]$, and distinguishes with probability at least .99 between*

$$(1)\ \mathbf{p} = \mathbf{u}_d, \qquad \textit{and} \qquad (2)\ \mathrm{d}_{\mathrm{TV}}(\mathbf{p}, \mathbf{u}_d) > \varepsilon. \tag{1}$$

The main idea behind this theorem is to combine very careful classical probabilistic analysis with a black-box use of Quantum Mean Estimation (QME) [19, 9, 21, 25, 20, 22]; see Section 2 for further discussion. Table 1 below compares our result to prior work on the problem. Table 1 has two columns because it seems that different algorithms are necessary depending on how d and ε relate. (Interestingly, this is not the case in the classical no-source-code model.) Thus combining our new result with that of [24], the best known upper bound becomes $O(\min\{d^{1/3}/\varepsilon^{4/3}, d^{1/2}/\varepsilon\})$. We remark that although [24]’s algorithm/analysis is already simple, we give an alternative simple algorithm and analysis achieving $O(d^{1/2}/\varepsilon)$ in Section A, employing the classical analysis + QME approach used in the proof of our main theorem.

[1] This is sometimes termed the “purified quantum query access model”, and is the most natural and general model. The “quantum string oracle”, referenced later in Table 1, refers to a situation in which one assumes a very specific type of source code for $\mathbf{p}$ (thus making algorithmic tasks easier). See Section 3 for details and [7] for a thorough discussion.

Lower bounds?

As for lower bounds (holding even in the quantum string oracle model): complexity $\Omega(1/\varepsilon)$ is necessary even in the case of constant $d = 2$, following from work of [26]; and, [12] showed a lower bound of $\Omega(d^{1/3})$ even in the case of constant ε, by reduction from the collision problem [2]. For reasons discussed in Section 2, we make the (somewhat bold) conjecture that our new upper bound is in fact tight for all d and ε:

▶ **Conjecture 2.** *Any algorithm that distinguishes $\mathbf{p} = \mathbf{u}_d$ from $\mathrm{d}_{\mathrm{TV}}(\mathbf{p}, \mathbf{u}_d) > \varepsilon$ with success probability at least .99 requires $\Omega(\min\{d^{1/3}/\varepsilon^{4/3}, d^{1/2}/\varepsilon\})$ uses of the code for $\mathbf{p}$. (Moreover, we conjecture this lower bound in the stronger quantum string oracle model.)*

Identity testing

Several prior works in this area have also studied the following natural generalization of uniformity testing: testing identity of the unknown distribution $\mathbf{p}$ to a known hypothesis distribution $\mathbf{q}$. An example application of this might be when $\mathbf{q}$ is a Porter–Thomas-type distribution arising as the ideal output of a random quantum circuit. Luckily, fairly recent work has given a completely generic reduction from *any* fixed identity testing problem to the uniformity testing problem; see [16], or [11, Section 2.2.3]. We can therefore immediately extend our new theorem to the general identity-testing setting:

▶ **Corollary 3.** *There is a computationally efficient quantum algorithm for identity testing to a reference distribution $\mathbf{q}$ over $[d]$ with the following guarantees: The algorithm makes $O(\min(d^{1/3}/\varepsilon^{4/3}, d^{1/2}/\varepsilon))$ uses of "the code" for an unknown distribution $\mathbf{p}$ over $[d]$, and distinguishes with probability at least .99 between*

$$(1)\ \mathbf{p} = \mathbf{q}, \qquad \textit{and} \qquad (2)\ \mathrm{d}_{\mathrm{TV}}(\mathbf{p}, \mathbf{q}) > \varepsilon. \tag{2}$$

(For completeness, we verify in Section C that the blackbox reduction does indeed carry through in our setting, preserving access to "the code".)

More fine-grained results

In proving our main theorem, we will in fact prove a strictly stronger version, one which is more fine-grained in two ways:

(1) *Tolerance:* Not only does our test accept with high probability when $\mathbf{p} = \mathbf{u}_d$, it also accepts with high probability when $\mathbf{p}$ is sufficiently close to $\mathbf{u}_d$.

Table 1 "Sample" complexity for uniformity testing with respect to total variation distance.

Reference	Large ε regime	Small ε regime	Access model
[28, 4]	$\Theta(d^{1/2}/\varepsilon^2)$		Classical, no source code
[10]	$O(d^{1/3})$ for $\varepsilon = \Theta(1)^*$		Quantum string oracle
[12]	$O(d^{1/3}/\varepsilon^2)$		Quantum string oracle
[15]		$O(d^{1/2}/\varepsilon) \cdot \log(d/\varepsilon)^3 \log\log(d/\varepsilon)$	Source code
[24]		$O(d^{1/2}/\varepsilon)$	Source code
This work	$O(d^{1/3}/\varepsilon^{4/3})$ for $\varepsilon \geq \frac{1}{\sqrt{d}}$		Source code

*The work states a bound of $O(d^{1/3}/\varepsilon^{4/3})$, but adds that ε must be constant.

(2) *Stricter distance measure.* Not only does our test reject with high probability when $\mathrm{d_{TV}}(\mathbf{p}, \mathbf{u}_d) > \varepsilon$, it also rejects with high probability when $\mathrm{d_H}(\mathbf{p}, \mathbf{u}_d) > \varepsilon$. (This is stronger, since $\mathrm{d_{TV}}(\mathbf{p}, \mathbf{q}) \leq \mathrm{d_H}(\mathbf{p}, \mathbf{q})$ always.)

To elaborate, recall the below chain of inequalities, which also includes KL- and χ^2-divergence. (We review probability distance measures in Section 3.)

$$\mathrm{d_{TV}}(\mathbf{p}, \mathbf{q})^2 \leq \mathrm{d}^2_{\mathrm{H}}(\mathbf{p}, \mathbf{q}) \leq \mathrm{KL}(\mathbf{p} \mid\mid \mathbf{q}) \leq \chi^2(\mathbf{p} \mid\mid \mathbf{q}). \tag{3}$$

The strictly stronger version of Theorem 1 that we prove is:

▶ **Theorem 4.** *There is a computationally efficient quantum algorithm for uniformity testing with the following guarantees: For $1/d \leq \theta \leq 1$, the algorithm makes $O(d^{1/3}/\theta^{2/3})$ uses of "the code" for an unknown distribution $\mathbf{p}$ over $[d]$, and distinguishes with probability at least $.99$ between*

$$(1)\ \chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) \leq .99\theta \quad \text{and} \quad \|\mathbf{p}\|_\infty \leq 100/d, \qquad \text{and} \qquad (2)\ \mathrm{d}^2_{\mathrm{H}}(\mathbf{p}, \mathbf{u}_d) > \theta. \tag{4}$$

We remark that most prior works on uniformity testing [10, 12, 15, 24] also had some additional such fine-grained aspects, beyond what is stated in Table 1.

Additional results

Speaking of χ^2-divergence, we mention two additional results we prove at the end of our work. These results additionally inform our Conjecture 2.

First, as mentioned earlier, in Section A we give an alternative proof of the $O(d^{1/2}/\varepsilon)$ upper bound of [24], and – like in that work – our result is tolerant with respect to χ^2-divergence. That is, we prove the strictly stronger result that for $\theta \leq 1/d$, one can use the code $O(d^{1/2}/\theta^{1/2})$ times to distinguish $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) \leq c\theta$ from $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) > \theta$ (for some constant $c > 0$).

Second, recall that $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$ can be as large as d. For example, $\chi^2(\mathbf{u}_S \mid\mid \mathbf{u}_d) = \frac{d}{r} - 1$ for any set $S \subseteq [d]$ of size r. Thus it makes sense to consider the uniformity testing problem even with respect to a χ^2-divergence threshold θ that exceeds 1. In Section B we show (albeit only in the quantum string oracle model) that for $\theta \geq 1$, one can use the code $O(d^{1/3}/\theta^{1/3})$ times to distinguish $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) \leq c\theta$ from $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) > \theta$, and this is optimal.

2 Technical overview of our proof

Our main algorithm is concerned with achieving the best possible ε-dependence for uniformity testing while maintaining a d-dependence of $d^{1/3}$; in this way, it is best compared with the older works of [10, 12], the latter of which achieves complexity $O(d^{1/3}/\varepsilon^2)$, as well as the classical (no-source-code) algorithm achieving complexity $O(d^{1/2}/\varepsilon^2)$. In fact, all four algorithms here are almost the same (except in terms of the number of samples they use). Let us describe our viewpoint on this common methodology.

We consider the algorithm as being divided into two Phases, and we may as well assume each Phase uses n samples. Phase 1 will have two properties:

- It will make n *black-box* draws from $\mathbf{p}$ (i.e., the source code is not used in Phase 1).
- Using these draws, Phase 1 will end by constructing a certain "random variable" – in the technical sense of a function $Y : [d] \to \mathbb{R}$.
- The mean of this random variable Y, vis-a-vis the unknown distribution $\mathbf{p}$, will ideally be close to $\chi^2(\mathbf{p} \mid\mid \mathbf{u}) = d \cdot \|\mathbf{p} - \mathbf{u}_d\|_2^2$. That is, ideally $\mu := \mathbb{E}_{\mathbf{p}}[Y] = \sum_{j=1}^{d} \mathbf{p}_j Y(j) \approx \chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$.

Phase 2 then performs a *mean estimation* algorithm on Y (vis-a-vis $\mathbf{p}$) to get an estimate of μ and therefore of $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$. Ideally, the resulting overall algorithm is not just a uniformity tester, but a χ^2-divergence-from-uniformity *estimator*. This could then be weakened to a TV-distance uniformity tester using the inequality $\mathrm{d}_{\mathrm{TV}}(\mathbf{p}, \mathbf{u}_d)^2 \le \chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$.

The mean estimation algorithm used in Phase 2 differs depending on whether one has the source code or not. In the classical (no source code) model, one simply uses the naive mean estimation algorithm based on n more black-box samples; by Chebyshev's inequality, this will (with high probability) give an estimate of μ to within $\pm O(\sigma/n^{1/2})$, where $\sigma \coloneqq \mathrm{stddev}_{\mathbf{p}}[Y] = \sqrt{\sum_{j=1}^{d}(Y(j) - \mu)^2}$. In the case of a quantum tester with the source code access, we can use a *Quantum Mean Estimation* (QME) routine; in particular, the one from [22] will (with high probability) yield an estimate of μ to within $\pm O(\sigma/n)$.[2]

A subtle aspect of this overall plan is that the mean μ and standard deviation σ of Y *are themselves random variables* (in the usual sense), where the randomness comes from Phase 1. Thus it is natural to analyze $\mathbb{E}_{\text{Phase 1}}[\mu]$ and $\mathbb{E}_{\text{Phase 1}}[\sigma]$. Of course, these depend on the definition of Y, which we now reveal: $Y(j) = \frac{d}{n} X_j - 1$, where X_j denotes the number of times $j \in [d]$ was drawn in Phase 1. The point of this definition of Y is that a short calculation implies

$$\mathbb{E}_{\text{Phase 1}}[\mu] = \chi^2(\mathbf{p} \mid\mid \mathbf{u}_d); \tag{5}$$

that is, the random variable μ is an *unbiased estimator* for our quantity of interest, the χ^2-divergence of $\mathbf{p}$ from $\mathbf{u}_d$. This is excellent, because although the algorithm does not see μ at the end of Phase 1, it will likely get a good estimate of it at the end of Phase 2...so long as (the random variable) σ is small.

We therefore finally have two sources of uncertainty about our final error (in estimating $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$):

1. Although $\mathbb{E}_{\text{Phase 1}}[\mu] = \chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$, the random variable μ may have fluctuated around its expectation at the end of Phase 1. One way to control this would be to bound $\mathrm{Var}_{\text{Phase 1}}[\mu]$ (and then use Chebyshev).
2. The Phase 2 mean estimation incurs an error proportional to σ. One way to control this would be to bound $\mathbb{E}_{\text{Phase 1}}[\sigma^2]$ (and then use Markov to get a high-probability bound on σ^2, and hence σ).

The quantities controlling the error here, $\mathrm{Var}_{\text{Phase 1}}[\mu]$ and $\mathbb{E}_{\text{Phase 1}}[\sigma^2]$, are explicitly calculable symmetric polynomials in $\mathbf{p}_1, \dots, \mathbf{p}_d$ of degree at most 4, depending on n. In principle, then, one can relate these quantities to $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) = d \cdot \|\mathbf{p} - \mathbf{u}_d\|_2^2$ itself, and derive a bound on how big n must be to (with high probability) get a good estimate of $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$.

In the classical (no source code) case, this methodology is a way to obtain the $O(d^{1/2}/\varepsilon^2)$ sample complexity, adding to the number of existing classical sample-optimal algorithms for the task. (This method in particular has some potential useful applications; e.g., one could consider decoupling the number of samples used in Phases 1 and 2 to, e.g., obtain tradeoffs for memory-limited settings). On one hand, with this method one can give a very compressed proof of the $O(d^{1/2}/\varepsilon^2)$ that, factoring out routine calculations, fits in half a

[2] This QME routine was not available at the time of [10, 12] which had to make do with Quantum Approximate Counting [9] – essentially, QME for Bernoulli random variables. But this is not the source of our improvement; one can obtain our main theorem with only a (polylog d)-factor loss using just Quantum Approximate Counting.

page (see, e.g., [27, Sec. 10]). On the other hand, one has to execute the calculations and estimations with great care, lest one would obtain a suboptimal result (there is a reason it took 8 years[3] to get the optimal quadratic dependence on ε [17, 28]).

In the case when source code is available, so that one can use the QME algorithm, how well does this methodology fare? On one hand, QME gives a quadratic improvement over naive classical mean estimation, meaning one can try to use signficantly fewer samples in Phase 2. But when one balances out the sample complexity between the two Phases, it implies one is using fewer samples in Phase 1, and hence one gets worse concentration of μ around its mean in Phase 1. So the calcuations become more delicate.

2.1 Heuristic calculations

Instead of diving into complex calculations, let's look at some heuristics. First, let's consider how the algorithm proceeds in the case when $\mathbf{p}$ really is the uniform distribution $\mathbf{u}_d$. In this case, as long as we're in a scenario where $n \ll d^{1/2}$, we will likely get all distinct elements in Phase 1, meaning that X_j will be 1 for exactly n values of j and X_j will be 0 otherwise. Then $Y(j)$ will be $\frac{d}{n} - 1$ for n values of j and will be -1 otherwise. This indeed means $\mu = \mathbb{E}_{\mathbf{p}}[Y] = \frac{1}{d}\sum_{j=1}^{d} Y(j) = 0 = \|\mathbf{p} - \mathbf{u}_d\|_2$ with *certainty* in Phase 1. This is very good; we get no error out of Phase 1. However QME in Phase 2 will not perfectly return the value $\mu = 0$; rather, it will return something in the range $\pm O(\sigma/n)$, where $\sigma = \sqrt{\frac{1}{d}\sum_{j=1}^{d}(Y(j) - 0)^2} = \sqrt{\frac{d}{n} - 1} \sim d^{1/2}/n^{1/2}$. Thus the value returned by QME may well be around $d^{1/2}/n^{3/2}$, which from the algorithm's point of view is consistent with $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) \approx d^{1/2}/n^{3/2}$. Thus the algorithm will only become confident that $\mathrm{d}_{\mathrm{TV}}(\mathbf{p}, \mathbf{u}_d)^2 \lessapprox d^{1/2}/n^{3/2}$, and hence it can only confidently accept in the case $\mathbf{p} = \mathbf{u}_d$ provided $d^{1/2}/n^{3/2} \lessapprox \varepsilon^2$; i.e., $n \gtrapprox d^{1/3}/\varepsilon^{4/3}$. We thereby see that with this algorithm, a uniformity testing upper bound of $O(d^{1/3}/\varepsilon^{4/3})$ is the *best* we can hope for. If one also believes that this algorithm might be optimal (and it *has* been the method of choice for essentially all previously known results), then this could possibly be taken as evidence for our Conjecture 2.

At this point, one might try to prove that complexity $O(d^{1/3}/\varepsilon^{4/3})$ *is* achievable; so far we have only argued that with this many samples, the algorithm will correctly accept when $\mathbf{p} = \mathbf{u}_d$ (with high probability). Again, before jumping into calculations, one might try to guess the "hardest" kind of ε-far distributions one might face, and try to work out the calculations for these cases. The hardest distributions in the classical case (i.e., the ones that lead to the matching $\Omega(d^{1/2}/\varepsilon^2)$ lower bound) are very natural: they are the $\mathbf{p}$'s in which half of the elements $j \in [d]$ have $\mathbf{p}_j = \frac{1+2\varepsilon}{d}$ and half have $\mathbf{p}_j = \frac{1-2\varepsilon}{d}$. Assuming this is the "worst case", one can calculate what $\mathrm{Var}_{\mathrm{Phase\ 1}}[\mu]$ and $\mathbb{E}_{\mathrm{Phase\ 1}}[\sigma^2]$ will be, and the calculations turn out just as desired. That is, with $n = O(d^{1/3}/\varepsilon^{4/3})$, these two error quantities can be shown to be suffciently small so that the overall algorthm will correctly become confident that $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) = d \cdot \|\mathbf{p} - \mathbf{u}_d\|_2^2 \le 4d \cdot \mathrm{d}_{\mathrm{TV}}(\mathbf{p}, \mathbf{u}_d)^2$ significantly exceeds ε^2, and hence the algorithm can correctly reject.

Everything therefore looks good, but there is a fly in the ointment. Even though this particular $\mathbf{p}$ with its values of $\frac{1\pm 2\varepsilon}{d}$ seems like the "hardest" distribution to face, one still has to reason about all possible $\mathbf{p}$'s with $\mathrm{d}_{\mathrm{TV}}(\mathbf{p}, \mathbf{u}_d)$. And when one does the calculations of $\mathrm{Var}[\mu]$ and $\mathbb{E}[\sigma^2]$ as prescribed by the standard methodology, the plan ends up *failing.*

[3] Technically, it took more than 8 years, as the proof of [28] was later shown to have a flaw: so the tight dependence had to wait until [4]. See [11, Section 2.3] for a discussion.

Specifically one gets too much error for $\mathbf{p}$'s that have somewhat "heavy" elements, meaning $\mathbf{p}_j$'s with $\mathbf{p}_j \gg 1/d$. The prior works [10, 12] cope with this failure by taking more samples; i.e., setting $n = O(d^{1/3}/\varepsilon^c)$ for $c > 4/3$ (specifically, [12] achieves $c = 2$). But our goal is to show that this is unnecessary – that the algorithm itself works, even though the standard and natural way of analyzing it fails.

In short, the reason the standard analysis of the algorithm fails is due to "rare events" that are caused by heavy elements in $\mathbf{p}$. These j's with $\mathbf{p}_j \gg 1/d$ may well still have $\mathbf{p}_j \ll 1/n$ (for our desired $n = O(d^{1/3}/\varepsilon^{4/3})$), and thus be drawn only rarely in Phase 1. The major difficulty is that when they *are* drawn, they generate a *very* large contribution to σ^2, causing $\mathbb{E}_{\text{Phase 1}}[\sigma^2]$ to be "misleadingly large". That is, when there are heavy elements, σ^2 may have the property of typically being much smaller than its expectation. Thus controlling the QME error using the *expected* value of σ^2 is a bad strategy.

Perhaps the key insight in our analysis is to show: *In those rare Phase 1 outcomes when σ^2 is unusually large, μ is* also *unusually large compared to its expectation.* The latter event is helpful, because if μ ends up much bigger than its expectation, we can tolerate a correspondingly worse error-bar from QME. In short, we show that the rare *bad* outcomes for σ^2 coincide with the rare *good* outcomes for μ.

In order to make this idea work out quantitatively, we (seem to) need to weaken our ambitions and get something a bit worse than a χ^2-divergence-from-uniform estimation algorithm, in two ways. (This is fine, as our main goal is just a non-tolerant uniformity tester with respect to TV.) First, rather than insisting that we accept with high probability when $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \leq .99\theta$ and reject with high probability when $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) > \theta$, we need to only require rejection when $\mathrm{d}_{\mathrm{H}}^2(\mathbf{p}, \mathbf{u}_d) > \theta$. The reason is that the rare large values of σ^2 that we face are only comparable with the larger value $\mathrm{d}_{\mathrm{H}}^2(\mathbf{p}, \mathbf{u}_d)$, and not with $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d)$.[4]

As for the second weakening we need to make: We explicitly add to our tester a check that the value of $\max_j\{X_j\}$ arising after Phase 1 is not too large. Roughly speaking, this extra test ensures that there are no very heavy elements. (Of course, this is satisfied when $\mathbf{p} = \mathbf{u}_d$, so we don't mind adding this test.) The reason we need to add this check is so that we can bound the quadratic expression $\sum_{j=1}^d X_j^2$ (which enters into the value of σ^2) by $\max_j\{X_j\} \cdot \sum_{j=1}^d X_j$; in turn, once $\max_j\{X_j\}$ is checked to be small, this expression can be bounded by the linear quantity $\sum_{j=1}^d X_j$, which can be related to μ. It is by relating σ^2 to μ in this way that we are able to show the correlation between rare events – that when σ^2 is big, μ is also big.

To conclude, we apologize to the reader for writing a "technical overview" whose length is nearly comparable to that of the actual proof itself. While we tried to make our argument as streamlined and concise as possible, we felt that it was worth conveying the ideas and detours which led us there, and which, while now hidden, motivated the final proof.

3 Preliminaries

3.1 Probability distances

Throughout, log and ln the binary and natural logarithms, respectively. We identify a probability distribution $\mathbf{p}$ over $[d] = \{1, 2, \ldots, d\}$ with its probability mass function (pmf), or,

[4] We remark that this χ^2-versus-Hellinger-squared dichotomy is quite reminiscent of the one that occurs in classical works on identity testing, such as [4].

equivalently, a vector $\mathbf{p} \in \mathbb{R}^d$ such that $\mathbf{p}_i \geq 0$ for all i and $\sum_{i=1}^d \mathbf{p}_i = 1$. For a subset $S \subseteq [d]$, we accordingly let $\mathbf{p}(S) = \sum_{i\in S} \mathbf{p}_i$. The *total variation distance* between two distributions $\mathbf{p}, \mathbf{q}$ over $[d]$ is defined as

$$\mathrm{d_{TV}}(\mathbf{p}, \mathbf{q}) = \sup_{S\subseteq[d]} \{\mathbf{p}(S) - \mathbf{q}(S)\} = \frac{1}{2}\|\mathbf{p} - \mathbf{q}\|_1 \in [0, 1], \tag{6}$$

where the last equality is from Scheffé's lemma. By Cauchy–Schwarz, this gives us the relation

$$\frac{1}{2}\|\mathbf{p} - \mathbf{q}\|_2 \leq \mathrm{d_{TV}}(\mathbf{p}, \mathbf{q}) \leq \frac{\sqrt{d}}{2}\|\mathbf{p} - \mathbf{q}\|_2. \tag{7}$$

We will in this paper also consider other notions of distance between probability distributions: the *squared Hellinger distance*, defined as

$$\mathrm{d}^2_{\mathrm{H}}(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^d (\sqrt{\mathbf{p}_i} - \sqrt{\mathbf{q}_i})^2 = \|\sqrt{\mathbf{p}} - \sqrt{\mathbf{q}}\|_2^2 \in [0, 2]. \tag{8}$$

(Some texts normalize this by a factor of $\frac{1}{2}$; we do not do so, as it makes our statements cleaner.) The *chi-squared divergence* is then defined as

$$\chi^2(\mathbf{p} \mid\mid \mathbf{q}) = \sum_{i=1}^d \frac{(\mathbf{p}_i - \mathbf{q}_i)^2}{\mathbf{q}_i} = \left(\sum_{i=1}^d \frac{\mathbf{p}_i^2}{\mathbf{q}_i}\right) - 1\,, \tag{9}$$

while the *Kullback–Leibler divergence* (least relevant to us, but quite common in the literature), in nats, is defined as

$$\mathrm{KL}(\mathbf{p} \mid\mid \mathbf{q}) = \sum_{i=1}^d \mathbf{q}_i \ln \frac{\mathbf{q}_i}{\mathbf{p}_i}\,. \tag{10}$$

As mentioned in Equation (3), we have the following well known [14] chain of inequalities:

$$\mathrm{d_{TV}}(\mathbf{p}, \mathbf{q})^2 \leq \mathrm{d}^2_{\mathrm{H}}(\mathbf{p}, \mathbf{q}) \leq \mathrm{KL}(\mathbf{p} \mid\mid \mathbf{q}) \leq \chi^2(\mathbf{p} \mid\mid \mathbf{q}). \tag{11}$$

Moreover, for the special case of the uniform distribution $\mathbf{u}_d$ over $[d]$, we have

$$\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) = d \cdot \|\mathbf{p} - \mathbf{u}_d\|_2^2\,. \tag{12}$$

3.2 Distribution access models

For a probability distribution $\mathbf{p}$ on $[d]$, we say a unitary $U_\mathbf{p}$ is a *synthesizer* for $\mathbf{p}$ if for some k

$$U_\mathbf{p} \left|0^k\right\rangle = \sum_{i\in[d]} \sqrt{p_i} \left|i\right\rangle \left|\psi_i\right\rangle, \tag{13}$$

where the $\left|\psi_i\right\rangle$'s are normalized states often called "garbage states". Note that any classical randomized circuit using S gates that samples from $\mathbf{p}$ can be converted to a synthesizer $U_\mathbf{p}$ in a purely black-box way with gate complexity $O(S)$. (See [22] for details and a more thorough discussion of synthesizers.)

In this paper, we say an algorithm makes t uses of "the code for $\mathbf{p}$" to mean that we use (as a black box) the unitaries $U_\mathbf{p}$, $U_\mathbf{p}^\dagger$, and controlled-$U_\mathbf{p}$ a total of t times in the algorithm.

Each of these unitaries is easy to construct given an explicit gate decomposition of $U_{\mathbf{p}}$ with the same gate complexity up to constant factors.

The *quantum string oracle*, which is used in many prior works, is a specific type of source code for $\mathbf{p}$. Here we have standard quantum oracle access to an m-bit string $x \in [d]^m$ for some m. For any symbol $i \in [d]$, the probability $\mathbf{p}_i$ is defined as the frequency with which that symbol appears in x, i.e., $\mathbf{p}_i = \frac{1}{m} |\{j : x_j = i\}|$. Note that calling this oracle on the uniform superposition over m gives us a synthesizer for $\mathbf{p}$. When a randomized sampler for $\mathbf{p}$ is converted to a synthesizer, we get a quantum string oracle, but quantum string oracles are not as general as arbitrary synthesizers. For example, all probabilities described by a quantum string oracle will be integer multiples of $\frac{1}{m}$, whereas an arbitrary synthesizer has no such constraint.

3.3 Quantum Mean Estimation

When we use QME, we will have the source code for some distribution $\mathbf{p}$ on $[d]$, and we will also have explicitly constructed some (rational-valued) random variable $Y : [d] \to \mathbb{Q}$ (say, simply as a table). From this, one can easily generate code that outputs a sample from Y (i.e., outputs $Y(\boldsymbol{j})$ for $\boldsymbol{j} \sim [d]$), using the code for $\mathbf{p}$ just one time. We will then use the following QME result from [22]:

▶ **Theorem 5.** *There is a computationally efficient quantum algorithm with the following guarantee: Given the source code for a random variable Y, as well as parameters n and δ, the algorithm uses the code $O(n \log(1/\delta))$ times and outputs an estimate $\widehat{\boldsymbol{\mu}}$ such that* $\Pr[|\widehat{\mu} - \mu| > \sigma/n] \leq \delta$*, where $\mu = \mathbb{E}[Y]$ and $\sigma = \mathrm{stddev}[Y]$.*

4 Algorithm in the Large Distance Regime

In this section, we establish Theorem 1, our main technical contribution. We do this by proving the strictly stronger Theorem 4, which we restate more formally:

▶ **Theorem 6.** *For any constant $B > 0$, there exists a computationally efficient quantum algorithm (Algorithm 1) with the following guarantees: on input $\frac{1}{d} \leq \theta \leq 1$, it makes $O(d^{1/3}/\theta^{2/3})$ uses (where the hidden constant depends on B) of "the code" for an unknown probability distribution $\mathbf{p}$ over $[d]$, and satisfies*

1. *If $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \leq .99\theta$ and $\|\mathbf{p}\|_\infty \leq B/d$, then the algorithm will* accept *with probability at least .99.*
2. *If $\mathrm{d}_{\mathrm{H}}^2(\mathbf{p}, \mathbf{u}_d) \geq \theta$, then the algorithm will* reject *with probability at least .99.*

Proof. Let us start by recording the following inequalities that we will frequently use:

$$n = \lceil c d^{1/3}/\theta^{2/3} \rceil, \ \theta \geq 1/d \quad \Longrightarrow \quad c/\theta \leq n \leq cd. \tag{14}$$

We begin with a simple lemma regarding the check on Section 4:

▶ **Lemma 7.** *If $\|\mathbf{p}\|_\infty \leq B/d$, then Section 4 will* reject *with probability at most .001. Conversely, if $\|\mathbf{p}\|_\infty > 2L/n$, then Section 4 will* reject *with probability at least .999.*

Proof. Let $\boldsymbol{X}_j \sim \mathrm{Bin}(n, p_j)$ denote the number of times j is drawn. The second ("conversely") part of of the proposition follows from a standard Chernoff bound. As for the first part, suppose $\|\mathbf{p}\|_\infty \leq B/d$. Now on one hand, if $n \leq d^{.99}/B$, so that $L = 100$, we have

$$\Pr[\mathrm{Bin}(n, p_j) \geq 100] \leq \binom{n}{100} p_j^{100} \leq ((en/100)p_j)^{100} \leq (e/(100d^{.01}))^{100} \leq .001/d, \tag{15}$$

Algorithm 1 for the large distance regime.

```
Require: Parameter $\frac{1}{d} \le \theta \le 1$, constant $B \ge 1$.
1: Let $c = c(B)$ and let $C = C(c)$ be sufficiently large, and let $L$ be defined as
       $L := \begin{cases} 100 & \text{if } n \le d^{.99}/B, \\ Bc\ln d & \text{if } n > d^{.99}/B. \end{cases}$
2: Set $n := \lceil c d^{1/3}/\theta^{2/3} \rceil$.
3: Make $n$ draws $\boldsymbol{J}_1, \dots, \boldsymbol{J}_n$, and let $\boldsymbol{X}_j = \sum_{t=1}^{n} \mathbb{1}_{\{J_t = j\}}$ be the number of times $j \in [d]$ is seen.
4: if $\boldsymbol{X}_j \ge L$ for any $j$ then reject
5: Do QME with $Cn$ “samples” on the random variable $\boldsymbol{Y}$ defined by $\boldsymbol{Y}_j = \frac{d}{n} X_j - 1$, obtaining $\widehat{\boldsymbol{\mu}}$.
6: if $\widehat{\boldsymbol{\mu}} \le .995\theta$ then accept
7: else reject
```

and thus $\boldsymbol{X}_j < 100$ for all j except with probability at most .001, as desired. Otherwise, $L = Bc\ln d$, and since $\mathbb{E}[\boldsymbol{X}_j] \le Bn/d \le Bc$, the desired result follows from a standard Chernoff and union bound (provided c is large enough). ◀

From this, we conclude:

- In Case (1), Line 4 rejects with probability at most .001.
- In Case (2), we may assume $\|\mathbf{p}\|_\infty \le 2L/n$ and $\|\boldsymbol{X}\|_\infty \le L$, else Line 4 rejects with probability $\ge .999$. Call this observation ($\diamondsuit$).

Now to begin the QME analysis, write $p_j = \frac{1+\varepsilon_j}{d}$, where $\varepsilon_j \in [-1, d-1]$, and let $\boldsymbol{\mu} = \sum_{j=1}^{d} p_j \boldsymbol{Y}_j$, the mean of $\boldsymbol{Y}$ (from QME's point of view). Writing $\eta := \mathrm{d}_{\mathrm{H}}^2(p, \mathbf{u}_d)$, our first goal will be to show:

$$\text{In Case (1),} \quad \boldsymbol{\mu} \le .991\theta \qquad \text{except with probability at most .001;} \tag{16}$$

$$\text{In Case (2),} \quad \boldsymbol{\mu} \ge .997\eta \qquad \text{except with probability at most .002.} \tag{17}$$

Starting with Equation (16), a short calculation (using $\sum_{j=1}^{d} \varepsilon_j = 0$) shows

$$\boldsymbol{\mu} = \operatorname*{avg}_{t=1}^{n}\{\varepsilon_{\boldsymbol{J}_t}\} \quad \Longrightarrow \quad \mathbb{E}[\boldsymbol{\mu}] = \frac{1}{d}\sum_{j=1}^{d} \varepsilon_j^2 = \chi^2(p \,||\, \mathbf{u}_d) \quad \Longrightarrow \quad \mathbb{E}[\boldsymbol{\mu}] \le .99\theta \text{ in Case (1).} \tag{18}$$

Also in Case (1) we get from Equation (18) that

$$\mathrm{Var}[\boldsymbol{\mu}] = \frac{1}{n}\mathrm{Var}_{\boldsymbol{j}\sim p}[\varepsilon_{\boldsymbol{j}}] \le \frac{1}{n}\mathbb{E}_{\boldsymbol{j}\sim p}[\varepsilon_{\boldsymbol{j}}^2] \le \frac{B}{nd}\sum_{j=1}^{n} \varepsilon_j^2 = \frac{B}{n}\chi^2(p \,||\, \mathbf{u}_d) \le \frac{.99B\theta}{n} \le \frac{B\theta^2}{c}, \tag{19}$$

the last inequality using Equation (14). Combining the preceding two inequalities and using Chebyshev, we indeed conclude Equation (16) (provided $c = c(B)$ is sufficiently large).

Towards Equation (17), let $b \ge 2$ be a certain universal constant to be chosen later, and say that $j \in [d]$ is *light* if $p_j \le b/d$ (i.e., $\varepsilon_j \le b-1$), *heavy* otherwise. We will write

$$\boldsymbol{\mu}_1 = \operatorname*{avg}_{t=1}^{n}\{\varepsilon_{\boldsymbol{J}_t} : \boldsymbol{J}_t \text{ heavy}\} \ge 0, \quad \boldsymbol{\mu}_2 = \operatorname*{avg}_{t=1}^{n}\{\varepsilon_{\boldsymbol{J}_t} : \boldsymbol{J}_t \text{ light}\} \qquad (\text{so } \boldsymbol{\mu} = \boldsymbol{\mu}_1 + \boldsymbol{\mu}_2), \tag{20}$$

and also observe

$$\eta = \mathrm{d}_{\mathrm{H}}^2(p, \mathbf{u}_d) = \frac{1}{d}\sum_{j=1}^{d}(\sqrt{1+\varepsilon_j}-1)^2 \leq \frac{1}{d}\sum_{j=1}^{d}\min\{|\varepsilon_j|, \varepsilon_j^2\} \leq \frac{1}{d}\sum_{\text{heavy } j}\varepsilon_j + \frac{1}{d}\sum_{\text{light } j}\varepsilon_j^2 =: \eta_1 + \eta_2. \quad (21)$$

Let us now make some estimates. First:

$$p_{\text{heavy}} \coloneqq \sum_{j \text{ heavy}} p_j = \frac{1}{d}\sum_{j \text{ heavy}}(1+\varepsilon_j) \geq \eta_1. \quad (22)$$

Also, similar to our Case (1) estimates we have

$$\mathbb{E}[\boldsymbol{\mu}_2] = \frac{1}{d}\sum_{\text{light } j}(\varepsilon_j^2 + \varepsilon_j) = \eta_2 - \eta_1 \quad \text{(where we used } \sum_{j=1}^{d}\varepsilon_j = 0\text{)}, \quad (23)$$

and

$$\begin{aligned}&\mathrm{Var}[\boldsymbol{\mu}_2]\\ &= \frac{1}{n}\mathrm{Var}_{\boldsymbol{j}\sim p}[\mathbb{1}_{\boldsymbol{j} \text{ light}}\cdot\varepsilon_{\boldsymbol{j}}] \leq \frac{1}{n}\mathbb{E}_{\boldsymbol{j}\sim p}[\mathbb{1}_{\boldsymbol{j} \text{ light}}\cdot\varepsilon_{\boldsymbol{j}}^2] \leq \frac{b}{nd}\sum_{j \text{ light}}\varepsilon_j^2\\ &= \frac{b}{n}\eta_2 \leq \frac{b}{c}\theta\eta_2 \leq \frac{b}{c}\eta_2\eta \text{ (in Case (2)).} \quad (24)\end{aligned}$$

We will now establish Equation (17); in fact, we we even will show the following very slightly stronger fact:

$$\text{In Case (2),} \qquad \boldsymbol{\mu} \geq .997(\eta_1+\eta_2) \geq .997\eta \quad \text{except with probability at most .002.} \quad (25)$$

We divide into two subcases:

Case (2a): $\eta_1 \leq .001\eta_2$. In this case we have $\eta_2 \geq \frac{1}{1.001}(\eta_1+\eta_2)$, and $\mathbb{E}[\boldsymbol{\mu}_2] \geq .999\eta_2$ from Equation (23). Since Equation (24) implies $\mathrm{Var}[\boldsymbol{\mu}_2] \leq 1.001\frac{b}{c}\eta_2^2$, Chebyshev's inequality tells us that $\boldsymbol{\mu}_2 \geq .998\eta_2$ except with probability at most .001 (provided c is large enough). But then $\boldsymbol{\mu} \geq \boldsymbol{\mu}_2 \geq \frac{.998}{1.001}(\eta_1+\eta_2)$, confirming Equation (25).

Case (2b): $\eta_1 > .001\eta_2$. In this case we have $\eta_1 \geq \frac{.001}{1.001}(\eta_1+\eta_2) \geq .0009(\eta_1+\eta_2)$. We now use that heavy j have $\varepsilon_j \geq b-1$ to observe that

$$\boldsymbol{\mu}_1 = \operatorname*{avg}_{t=1}^{n}\{\varepsilon_{\boldsymbol{J}_t} : \boldsymbol{J}_t \text{ heavy}\} \geq (b-1)\cdot(\text{fraction of } \boldsymbol{J}_t\text{'s that are heavy}) = (b-1)\cdot\frac{\mathrm{Bin}(n, p_{\text{heavy}})}{n} \quad (26)$$

(in distribution). We see that $\mathbb{E}[\boldsymbol{\mu}_1] \geq (b-1)p_{\text{heavy}}$, and moreover concentration of Binomials and Equation (22) imply that

$$\boldsymbol{\mu}_1 \geq \frac{1}{2}(b-1)p_{\text{heavy}} \geq \frac{1}{2}(b-1)\eta_1 \text{ except with probability at most .001,} \quad (27)$$

provided that $p_{\text{heavy}}n$ is a sufficiently large constant. But we can indeed ensure this by taking c sufficient large: by Equation (22), being in Case (2b), and Equation (14), it holds that

$$p_{\text{heavy}}n \geq \eta_1 n \geq .0009(\eta_1+\eta_2)n \geq .0009\eta n \geq .0009\theta n \geq .0009c. \quad (28)$$

At the same time, Equation (23) certainly implies $\mathbb{E}[\boldsymbol{\mu}_2] \geq -\eta_1$, and Equation (24) implies $\mathrm{Var}[\boldsymbol{\mu}_2] \leq \frac{b}{c}\eta_2(\eta_1 + \eta_2) \leq \frac{1000 \cdot 1001 b}{c}\eta_1^2$ (using Case (2b)). Thus Chebyshev implies

$$\boldsymbol{\mu}_2 \geq -1.1\eta_1 \text{ except with probability at most .001,} \tag{29}$$

provided c is large enough. Combining Equations (27) and (29) yields

$$\boldsymbol{\mu} = \boldsymbol{\mu}_1 + \boldsymbol{\mu}_2 \geq (\tfrac{b-1}{2} - 1.1)\eta_1 \geq .0009(\tfrac{b-1}{2} - 1.1)(\eta_1 + \eta_2) \text{ except with probability at most .002,} \tag{30}$$

which verifies Equation (25) provided b is a large enough constant.

We have now verified the properties of $\boldsymbol{\mu}$ claimed in Equations (16) and (25). Next we analyze the random variable $\boldsymbol{\sigma}^2$ that represents the variance of $\boldsymbol{Y}$ (from QME's point of view). Our goal will be to show:

$$\text{In Case (1),} \quad \boldsymbol{\sigma}^2/(Cn)^2 \leq 10^{-6} \cdot \theta^2 \quad \text{except with probability at most .001,} \tag{31}$$

$$\text{In Case (2),} \quad \boldsymbol{\sigma}^2/(Cn)^2 \leq 10^{-6} \cdot \boldsymbol{\mu}^2 \quad \text{except with probability at most .001.} \tag{32}$$

Together with Equations (16) and (25), these facts are sufficient to complete the proof of the theorem, by the QME guarantee of Theorem 5.

We have:

$$\boldsymbol{\sigma}^2 := \sum_{j=1}^d p_j \boldsymbol{Y}_j^2 - \boldsymbol{\mu}^2 = (d/n)^2 \sum_{j=1}^d p_j \boldsymbol{X}_j^2 - (\boldsymbol{\mu}+1)^2 \leq (d/n)^2 \sum_{j=1}^d p_j \boldsymbol{X}_j^2 = \boldsymbol{\sigma}_S^2 + \boldsymbol{\sigma}_{S^c}^2, \tag{33}$$

where we've defined $\boldsymbol{\sigma}_S^2 := (d/n)^2 \sum_{j\in S} p_j \boldsymbol{X}_j^2$ and $S^c = [d] \setminus S$. We will be making two different choices for S later, but we will always assume

$$S \supseteq \{j : j \text{ light}\}, \quad \text{which implies } \sum_{j \in S} \varepsilon_j \leq 0 \tag{34}$$

(the implication because $\sum_{j=1}^d \varepsilon_j = 0$ and S^c contains only j's with $\varepsilon_j \geq b - 1 \geq 0$). Now since $\mathbb{E}[\boldsymbol{X}_j^2] = np_j(1-p_j) + (np_j)^2 \leq np_j + (np_j)^2$, we have

$$\mathbb{E}[\boldsymbol{\sigma}_S^2] \leq (d^2/n) \sum_{j\in S} p_j^2 + d^2 \sum_{j \in S} p_j^3 \tag{35}$$

$$\leq d/n + (2/n)\sum_{j\in S}\varepsilon_j + (1/n)\sum_{j\in S}\varepsilon_j^2 + 1/d + (3/d)\sum_{j\in S}\varepsilon_j + (3/d)\sum_{j\in S}\varepsilon_j^2 + (1/d)\sum_{j\in S}\varepsilon_j^3 \tag{36}$$

$$\leq (5cd/n)\left(1 + \frac{1}{d}\sum_{j\in S}\varepsilon_j + \frac{1}{d}\sum_{j\in S}\varepsilon_j^2\right) + \frac{1}{d}\sum_{j\in S}\varepsilon_j^3 \tag{37}$$

(where the last inequality used $1/d \leq c/n \leq (c-1)d/n$ from Equation (14)). Using Equation (34) to drop the term of Equation (37) that's linear in the ε_j's, we thereby conclude

$$\mathbb{E}[\boldsymbol{\sigma}_S^2/(Cn)^2] \leq \mathbb{E}[\boldsymbol{\sigma}_S^2/n^2] \leq (5cd/n^3)\left(1 + \frac{1}{d}\sum_{j\in S}\varepsilon_j^2\right) + (d^{1/2}/n^2)\left(\frac{1}{d}\sum_{j\in S}\varepsilon_j^2\right)^{3/2} \tag{38}$$

$$\leq (5\theta^2/c^2)(1+\eta_S) + \frac{\theta^{4/3}}{c^2 d^{1/6}}\eta_S^{3/2}, \tag{39}$$

where $\eta_S \coloneqq \frac{1}{d}\sum_{j \in S} \varepsilon_j^2$. In Case (1) we select $S = [d]$, so $\eta_S = \chi^2(p \,||\, \mathbf{u}_d) \leq .99\theta \leq \theta \leq 1$, and the above bound gives

$$\text{Case (1)} \implies \mathbb{E}[\boldsymbol{\sigma}^2/(Cn)^2] \leq 10\theta^2/c^2 + \frac{\theta^{17/6}}{c^2 d^{1/6}} \leq \cdot 10^{-9} \cdot \theta^2 \tag{40}$$

(provided c is large enough). Now Equation (31) follows by Markov's inequality.

In Case (2) we select $S = \{j : j \text{ light}\}$, so $\eta_S = \eta_2$ and we conclude (using obvious notation)

$$\text{Case (2)} \implies \mathbb{E}[\boldsymbol{\sigma}^2_{\text{light}}/(Cn)^2] \leq (5\theta^2/c^2)(1+\eta_2) + \frac{\theta^{4/3}}{c^2 d^{1/6}}\eta_2^{3/2} \leq .4 \cdot 10^{-9} \cdot (\eta_1+\eta_2)^2, \tag{41}$$

(provided c large enough), where we used $\theta \leq \eta \leq \eta_1 + \eta_2$ and also $\theta \leq 1$. We now complete the bounding of $\boldsymbol{\sigma}^2$ in Case (2) by two different strategies:

Case (2.i): $n > d^{.99}/B$. In this case, $L = Bc \ln d$, and ($\diamondsuit$) tells us $\|\mathbf{p}\|_\infty \leq 2L/n$, so we have

$$\|\mathbf{p}\|_\infty \leq \frac{2Bc\ln d}{n} \leq \frac{2B^2 c \ln d}{d^{.99}}. \tag{42}$$

Now returning to Equation (37), we get

$$\mathbb{E}[\boldsymbol{\sigma}^2_{\text{heavy}}/(Cn)^2] \leq \frac{5cd}{C^2n^3} + \frac{5cd}{C^2n^3}\left(1 + \varepsilon_{\max} + \frac{n}{5cd}\varepsilon^2_{\max}\right) \cdot \frac{1}{d}\sum_{j \text{ heavy}} \varepsilon_j \tag{43}$$

$$\leq \frac{5\theta^2}{(Cc)^2} + \frac{5cd^2}{C^2n^3}\left(\|\mathbf{p}\|_\infty + \frac{n}{5c}\cdot\|\mathbf{p}\|^2_\infty\right)\eta_1 \leq \frac{5\theta^2}{(Cc)^2} + \frac{14B^6c^2\ln^2 d}{C^2 d^{1.96}}\eta_1, \tag{44}$$

where we used Equation (42) and $n > d^{.99}/B$. We can again bound the first expression in Equation (44) as $\frac{5\theta^2}{(Cc)^2} \leq 10^{-6}\cdot(\eta_1+\eta_2)^2$. As for the second expression, either $\eta_1 = 0$ (there are no heavy j's) or else $\eta_1 \geq \frac{b-1}{d}$ (there is at least one heavy j). In either case, we have $\eta_1 \leq \frac{d}{b-1}\eta_1^2 \leq d\eta_1^2$, so we can bound this second expression by

$$\frac{14B^6c^2\ln^2 d}{C^2 d^{.96}}\eta_1^2 \leq .4 \cdot 10^{-9} \cdot (\eta_1+\eta_2)^2 \tag{45}$$

where we used $C = C(c)$ sufficiently large (and we could have taken $C = 1$ were willing to assume d sufficiently large). Putting this bound together with Equation (41) we obtain:

$$\text{Case (2.i)} \implies \mathbb{E}[\boldsymbol{\sigma}/(Cn)^2] \leq .8\cdot 10^{-9}\cdot(\eta_1+\eta_2)^2 \leq \tfrac{.8}{.997}\cdot 10^{-9}\cdot\boldsymbol{\mu}^2 \leq \cdot 10^{-9}\cdot\boldsymbol{\mu}^2, \tag{46}$$

using Equation (25). Equation (32) now follows (in this Case (2.i)) by Markov's inequality.

Case (2.ii): $n \leq d^{.99}/B$. In this case we use a different strategy. Recall from Equation (33) that

$$\boldsymbol{\sigma}^2 \leq (d/n)^2 \sum_{j=1}^d p_j \boldsymbol{X}_j^2 \leq (d/n)^2\|\boldsymbol{X}\|_\infty \sum_{j=1}^d p_j \boldsymbol{X}_j = (d/n)\|\boldsymbol{X}\|_\infty(1+\boldsymbol{\mu}). \tag{47}$$

By ($\diamondsuit$) we may assume $\|\boldsymbol{X}\|_\infty \leq L = 100$, the equality because we are in Case (2.ii). Thus

$$\boldsymbol{\sigma}^2/(Cn)^2 \leq \boldsymbol{\sigma}^2/n^2 \leq 100(d/n^3)(1+\boldsymbol{\mu}) \leq \frac{100\theta^2}{c^3} + \frac{100\theta^2}{c^3}\boldsymbol{\mu} \leq 10^{-6}\cdot\boldsymbol{\mu}^2. \tag{48}$$

(provided c large enough), where we used $\theta \leq \eta \leq \frac{1}{.997}\boldsymbol{\mu}$ (from Equation (25)) and also $\theta \leq 1$. This verifies Equation (32) in Case (2.ii), completing the proof. ◀

References

1 Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. *Theory of Computing*, 9(4):143–252, 2013. doi:10.4086/toc.2013.v009a004.

2 Scott Aaronson and Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. *J. ACM*, 51(4):595–605, July 2004. doi:10.1145/1008731.1008735.

3 Jayadev Acharya, Clément L. Canonne, Yanjun Han, Ziteng Sun, and Himanshu Tyagi. Domain compression and its application to randomness-optimal distributed goodness-of-fit. In *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 3–40. PMLR, July 2020. URL: http://proceedings.mlr.press/v125/acharya20a.html.

4 Jayadev Acharya, Constantinos Daskalakis, and Gautam C. Kamath. Optimal Testing for Properties of Distributions. In *Advances in Neural Information Processing Systems 28*, pages 3577–3598. Curran Associates, Inc., 2015.

5 Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007. doi:10.1137/S0097539705447311.

6 Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, October 2019. doi:10.1038/s41586-019-1666-5.

7 Aleksandrs Belovs. Quantum algorithms for classical probability distributions. In *Proceedings of the 27th Annual European Symposium on Algorithms (ESA)*, pages 50–59. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.1007/978-3-030-19955-5_5.

8 Sergio Boixo, Sergei V. Isakov, Vadim N. Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J. Bremner, John M. Martinis, and Hartmut Neven. Characterizing quantum supremacy in near-term devices. *Nature Physics*, 14(6):595–600, April 2018. doi:10.1038/s41567-018-0124-x.

9 Gilles Brassard, Peter Høyer, and Alain Tapp. Quantum counting. In *Proceedings of the 25th Annual International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 820–831. Springer–Verlag, 1998. doi:10.1007/bfb0055105.

10 Sergey Bravyi, Aram Harrow, and Avinatan Hassidim. Quantum algorithms for testing properties of distributions. *Transactions on Information Theory*, 57(6):3971–3981, 2011. doi:10.1109/TIT.2011.2134250.

11 Clément L. Canonne. Topics and techniques in distribution testing: A biased but representative sample. *Foundations and Trends® in Communications and Information Theory*, 19(6):1032–1198, 2022. doi:10.1561/0100000114.

12 Sourav Chakraborty, Eldar Fischer, Arie Matsliah, and Ronald de Wolf. New Results on Quantum Property Testing. In *30th International Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2010)*, volume 8 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 145–156, Dagstuhl, Germany, 2010. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.FSTTCS.2010.145.

13 Ilias Diakonikolas and Daniel M. Kane. A new approach for testing properties of discrete distributions. In *57th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2016*. IEEE Computer Society, 2016.

14 Alison Gibbs and Francis Su. On choosing and bounding probability metrics. *International Statistical Rreview*, 70(3):419–435, 2002.

15 András Gilyén and Tongyang Li. Distributional Property Testing in a Quantum World. In *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*, volume 151 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 25:1–25:19, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.ITCS.2020.25`.

16 Oded Goldreich. The uniform distribution is complete with respect to testing identity to a fixed distribution. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:15, 2016. URL: `http://eccc.hpi-web.de/report/2016/015`.

17 Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. Technical Report TR00-020, Electronic Colloquium on Computational Complexity (ECCC), 2000.

18 Lov Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the* 28*th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 212–219. ACM, New York, 1996. `doi:10.1145/237814.237866`.

19 Lov Grover. A framework for fast quantum mechanical algorithms. In *Proceedings of the* 30*th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 53–62. ACM, New York, 1998. `doi:10.1145/276698.276712`.

20 Yassine Hamoudi. *Quantum Algorithms for the Monte Carlo Method*. PhD thesis, Université de Paris, 2021.

21 Stefan Heinrich. Quantum summation with an application to integration. *Journal of Complexity*, 18(1):1–50, 2002. `doi:10.1006/jcom.2001.0629`.

22 Robin Kothari and Ryan O'Donnell. *Mean estimation when you have the source code; or, quantum Monte Carlo methods*, pages 1186–1215. Society for Industrial and Applied Mathematics, January 2023. `doi:10.1137/1.9781611977554.ch44`.

23 Samuel Kutin. Quantum lower bound for the collision problem with small range. *Theory of Computing*, 1(2):29–36, 2005. `doi:10.4086/toc.2005.v001a002`.

24 Jingquan Luo, Qisheng Wang, and Lvzhou Li. Succinct quantum testers for closeness and k-wise uniformity of probability distributions. *IEEE Trans. Inf. Theory*, 70(7):5092–5103, 2024.

25 Ashley Montanaro. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A*, 471(2181):20150301, 20, 2015. `doi:10.1098/rspa.2015.0301`.

26 Ashwin Nayak and Felix Wu. The quantum query complexity of approximating the median and related statistics. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing*, STOC '99, pages 384–393, New York, NY, USA, 1999. Association for Computing Machinery. `doi:10.1145/301250.301349`.

27 Ryan O'Donnell and John Wright. Learning and testing quantum states via probabilistic combinatorics and representation theory. In *Current developments in mathematics 2021*, pages 43–94. International Press, Somerville, MA, 2023.

28 Liam Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 54(10):4750–4755, 2008. `doi:10.1109/TIT.2008.928987`.

A Algorithm in the Small Distance Regime

In this appendix, we provide an alternative (and arguably simpler) proof of the main result of [24]:

▶ **Theorem 8.** *There is a computationally efficient quantum algorithm (Algorithm 2) for uniformity testing with the following guarantees: it takes $O(d^{1/2}/\varepsilon)$ "samples" from an* *unknown probability distribution* $\mathbf{p}$ *over* $[d]$*, and distinguishes with probability at least* $2/3$ *between (1)* $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \leq \frac{\varepsilon^2}{144}$*, and (2)* $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) > \varepsilon^2$*.*

This in turn will follow from the more general result on tolerant ℓ_2 closeness testing, where one is given access to the source code for *two* unknown probability distributions $\mathbf{p}, \mathbf{q}$ over $[d]$, and one seeks to distinguish $\|\mathbf{p} - \mathbf{q}\|_2 \leq c \cdot \tau$ from $\|\mathbf{p} - \mathbf{q}\|_2 \geq \tau$.

▶ **Theorem 9.** *There is a computationally efficient quantum algorithm (Algorithm 2) for closeness testing with the following guarantees: it takes* $O(1/\tau)$ *"samples" from two unknown probability distributions* $\mathbf{p}, \mathbf{q}$ *over* $[d]$*, and distinguishes with probability at least* $2/3$ *between (1)* $\|\mathbf{p} - \mathbf{q}\|_2 \leq \frac{\tau}{12}$*, and (2)* $\|\mathbf{p} - \mathbf{q}\|_2 > \tau$*.*

Theorem 8 can then be obtained as a direct corollary by setting $\tau = \varepsilon/\sqrt{d}$, recalling that when $\mathbf{q}$ is the uniform distribution $\mathbf{u}_d$, ℓ_2 distance and χ^2 divergence are equivalent:

$$\|\mathbf{p} - \mathbf{u}_d\|_2^2 = \sum_{i=1}^{d} (\mathbf{p}_i - 1/d)^2 = \frac{1}{d} \sum_{i=1}^{d} \frac{(\mathbf{p}_i - 1/d)^2}{1/d} = \frac{1}{d} \chi^2(\mathbf{p} \,||\, \mathbf{u}_d)$$

We emphasize that the result of Theorem 9 itself is not new, as a quantum algorithm achieving the same sample complexity (in the same access model) was recently obtained by [24].[5] However, our algorithm differs significantly from the one in [24], and we believe it to be of independent interest for several reasons:

- it is *conceptually very simple*: (classically) hash the domain down to *two* elements, and use QME to estimate the bias of the resulting Bernoulli;
- it neatly *separates the quantum and classical aspects* of the task, only using QME (as a blackbox) in a single step of the algorithm;
- in contrast to the algorithm of [24], it *decouples the use of the source code from* $\mathbf{p}$ *and* $\mathbf{q}$, allowing one to run our algorithm when the accesses to the two distributions are on different machines, locations, or even will be granted at different points in time (i.e., one can run part of the algorithm using the source code for $\mathbf{p}$, and, one continent and a year apart, run the remaining part on the now-available source code for $\mathbf{q}$ without needing $\mathbf{p}$ anymore).

The idea behind Theorem 9 is relatively simple: previous work (in the classical setting) showed that hashing the domain from d to a much smaller $d' \ll d$ could yield sample-optimal testing algorithms in some settings, e.g., when testing under privacy bandwidth, or memory constraints. Indeed, while this "domain compression" reduces the total variation distance by a factor $\Theta(\sqrt{d'/d})$, this shrinkage is, in these settings, balanced by the reduction in domain size. The key insight in our algorithm is then to (1) use this hashing with respect to ℓ_2 distance, not total variation distance, and show that one can in this case get a two-sided guarantee in the distance (low-distortion embedding) instead of a one-sided one; and (2) compress the domain all the way to $d' = 2$, so that one can then invoke the QME algorithm to simply estimate the bias of a coin to an additive $\pm\tau$, a task for which a quantum quadratic speedup is well known.

Proof of Theorem 9. As mentioned above, a key building block of our algorithm is the following "binary hashing lemma," a simple case of the domain compression primitive of [3]:

[5] Technically, [24]'s result can be seen as slightly stronger, in that it allows to test $\|\mathbf{p} - \mathbf{q}\|_2 \leq (1-\gamma)\tau$ vs. $\|\mathbf{p} - \mathbf{q}\|_2 \mathbf{u}_d > \tau$, for arbitrarily small constant $\gamma > 0$.

▶ **Lemma 10** (Random Binary Hashing (Lemma 2.9 and Remark 2.4 of [11]). *Let $\mathbf{p}, \mathbf{q} \in \Delta(d)$. Then, for every $\alpha \in [0, 1/2]$,*

$$\Pr_S[\,|\mathbf{p}(S) - \mathbf{q}(S)| \geq \alpha \|\mathbf{p} - \mathbf{q}\|_2\,] \geq \frac{1}{12}(1 - 4\alpha^2)^2\,,$$

where $S \subseteq [d]$ is a uniformly random subset of $[d]$.

Given our goal of tolerant testing, we also require a converse to Lemma 10, stated and proven below:

▶ **Lemma 11.** *Let $\mathbf{p}, \mathbf{q} \in \Delta(d)$. Then, for every $\beta \in [1/2, \infty)$,*

$$\Pr_S[\,|\mathbf{p}(S) - \mathbf{q}(S)| \geq \beta \|\mathbf{p} - \mathbf{q}\|_2\,] \leq \frac{1}{4\beta^2}\,,$$

where $S \subseteq [d]$ is a uniformly random subset of $[d]$.

Proof. As in the proof of Lemma 10, we write $\delta := \mathbf{p} - \mathbf{q} \in \mathbb{R}^d$ and $\mathbf{p}(S) - \mathbf{q}(S) = \frac{1}{2}Z$, where $Z := \sum_{i=1}^d \delta_i \xi_i$ for $\xi_1, \dots, \xi_d$ i.i.d. Rademacher. We will use the following fact established in the proof of this lemma, which we reproduce for completeness:

$$\mathbb{E}\left[Z^2\right] = \sum_{1 \leq i,j \leq d} \delta_i \delta_j \mathbb{E}[\xi_i \xi_j] = \sum_{i=1}^d \delta_i^2 = \|\delta\|_2^2\,. \tag{49}$$

By Markov's inequality, we then have

$$\Pr_S[\,|\mathbf{p}(S) - \mathbf{q}(S)| > \beta \|\mathbf{p} - \mathbf{q}\|_2\,] = \Pr_S\left[\,Z^2 > 4\beta^2 \mathbb{E}\left[Z^2\right]\,\right] \leq \frac{1}{4\beta^2}$$

concluding the proof. ◀

While the above two lemmas allow us to obtain a slightly more general result than in the theorem statement by keeping α, β as free parameters, for concreteness, set $\alpha := 1/(2\sqrt{2})$ and $\beta = 4$. This implies the following:

- If $\|\mathbf{p} - \mathbf{q}\|_2 \geq \tau$, then

$$\Pr_S\left[\left|\mathbf{p}(S) - \frac{|S|}{d}\right| \geq \frac{\tau}{\sqrt{8}}\right] \geq \frac{1}{48}$$

- If $\|\mathbf{p} - \mathbf{q}\|_2 \leq \frac{\tau}{12}$, then

$$\Pr_S\left[\left|\mathbf{p}(S) - \frac{|S|}{d}\right| \geq \frac{\tau}{\sqrt{9}}\right] \leq \frac{1}{64}\,.$$

where $S \subseteq [d]$ is a uniformly random subset of $[d]$. This allows us to distinguish between the two cases with only $O(1)$ repetitions:

Algorithm 2 QME+Binary Hashing Tester.

```
Set $T = O(1)$, $\delta := \frac{1}{600}$, $\tau := \frac{1/48+1/64}{2}$.                ▷ $\delta \leq \frac{1}{3}\left(\frac{1}{48} - \frac{1}{64}\right)$.
for $t = 1$ to $T$ do
    Pick a u.a.r. subset $S_t \subseteq [d]$ (independently of previous iterations)
    Estimate $\mathbf{p}(S_t), \mathbf{q}(S_t)$ by $\hat{p}_t, \hat{q}_t$ to within $\pm\frac{\tau}{100}$ with error probability $\delta$.    ▷ QME
    if $|\hat{p}_t - \hat{q}_t| \leq \frac{\varepsilon}{\sqrt{8d}}$ then $b_t \leftarrow 0$
    else $b_t \leftarrow 1$
return accept if $\frac{1}{T}\sum_{t=1}^T b_t \leq \tau$                ▷ Estimate of the probability accept
```

A standard analysis shows that, for T a sufficiently large constant, with probability at least $2/3$ the estimate $\frac{1}{T}\sum_{t=1}^{T} b_t$ will be within an additive $\delta + \frac{1}{1000}$ of the corresponding value (either $1/48$ or $1/64$), in which case the output is correct. The total number of samples required is T times the sample of the Quantum Mean Estimation call on Line 4, which is $O(1/\tau)$: the complexity of getting a $O(\tau)$-additive estimate of the mean of a Bernoulli random variable with high (constant) probability. This concludes the proof. ◀

B Algorithm in the Giant Distance Regime

In this appendix, we show that, in the (stronger) quantum string oracle model, one can perform tolerant uniformity testing with respect to χ^2 divergence in the "very large parameter regime," that is, to distinguish $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \leq c\theta$ from $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) > \theta$ for $\theta \geq 1$:

▶ **Theorem 12.** *There is a computationally efficient quantum algorithm for uniformity testing with the following guarantees: For $\theta \geq 1$, the algorithm makes $O(d^{1/3}/\theta^{1/3})$ calls to the quantum string oracle for an unknown distribution $\mathbf{p}$ over $[d]$, and distinguishes with probability at least .99 between*

$$(1)\ \chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \leq c \cdot \theta, \qquad \textit{and} \qquad (2)\ \chi^2(\mathbf{p} \,||\, \mathbf{u}_d) > \theta\,, \tag{50}$$

where $c > 0$ is an absolute constant. Moreover, this query complexity is optimal.

Note that, as discussed in the introduction, this result does not imply anything in terms of total variation distance, as the latter is always at most 1; however, we believe this result to be of interest for at least two reasons: (1) it is in itself a reasonable (and often useful) testing question, when total variation distance is not the most relevant distance measure, and implies, for instance, testing $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \leq c \cdot \theta$ from $\mathrm{KL}(\mathbf{p} \,||\, \mathbf{u}_d) > \theta$; and (2) one can show that this complexity is tight, by a reduction to the θ-to-1 collision problem, which provides additional evidence for Conjecture 2.

Proof. The main ingredient of the proof is the following lemma, which guarantees that taking $N = \Theta(d/\theta)$ from the unknown distribution $\mathbf{p}$ is enough to obtain (with high constant probability) a multiset of elements with, in one case, no collisions, and in the other at least one collision:

▶ **Lemma 13.** *For $\theta \geq 1$, there exists a constant $c \in (0,1)$ such that taking N i.i.d. samples from an unknown $\mathbf{p}$ over $[d]$ results in a multiset S satisfying the following with probability at least .99:*

- *If $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \leq c \cdot \theta$, then all elements in S are distinct;*
- *If $\chi^2(\mathbf{p} \,||\, \mathbf{u}_d) \geq \theta$, then at least two elements in S are identical;*

as long as $1601 \cdot \frac{d}{\theta} \leq N \leq \frac{1}{10c} \cdot \frac{d}{\theta}$. (In particular, taking $c := \frac{1}{16010}$ suffices to ensure such a choice of N is possible.)

Before proving this lemma, we describe how it implies our stated complexity upper bound. Lemma 13 guarantees that we can reduce our testing problem to that of deciding if, given oracle access to a string of size $N = \Theta(\sqrt{d/\theta})$, whether all the elements in it are distinct. This problem is solved by Ambainis' element distinctness quantum-walk algorithm [5] using $O(N^{2/3}) = O(d^{1/3}/\theta^{1/3})$ quantum queries.

Proof of Lemma 13. Suppose we take N i.i.d. samples $X_1, \dots, X_N$ from $\mathbf{p}$, and count the number Z of collisions among them:

$$Z := \sum_{1 \leq i < j \leq N} \mathbb{1}_{\{X_i = X_j\}}$$

Letting $\delta \coloneqq \mathbf{p} - \mathbf{u}_d$ and $\operatorname{pow}_t(x) \coloneqq \sum_{i=1}^d x_i^t$ for all integer $t \geq 0$ and vector $x \in \mathbb{R}^d$ (so that $\delta_i = \mathbf{p}_i - 1/d$ for all i), we have, $\operatorname{pow}_1(\delta) = 0$, and

$$\operatorname{pow}_2(\delta) = \|\mathbf{p} - \mathbf{u}_d\|_2^2 = \frac{1}{d}\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d)$$

Now, it is not hard to verify that $\mathbb{E}[Z] = \binom{N}{2}\|\mathbf{p}\|_2^2 = \binom{N}{2}(\operatorname{pow}_2(\delta) + 1/d)$, and

$$\begin{aligned}\operatorname{Var}[Z] &= \binom{N}{2}\|\mathbf{p}\|_2^2\Big(1 - \|\mathbf{p}\|_2^2\Big) + 6\binom{N}{3}\Big(\|\mathbf{p}\|_3^3 - \|\mathbf{p}\|_2^4\Big) \\ &\leq \mathbb{E}[Z] + 6\binom{N}{3}\left(\operatorname{pow}_3(\delta) + \frac{3}{d}\operatorname{pow}_2(\delta)\right) \qquad (51)\end{aligned}$$

From this, we get, setting $\tau \coloneqq \sqrt{\theta/d} \geq 1/\sqrt{d}$:

- If $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) \leq c \cdot \theta$, then $\operatorname{pow}_2(\delta) \leq c^2 \cdot \tau^2$, and as long as $N \leq \frac{1}{10c\tau}$ we have $\binom{N}{2}(c^2 \cdot \tau^2 + 1/d) \leq 1/100$, so that by Markov's inequality

$$\Pr[\,Z \geq 1\,] \leq \Pr[\,Z \geq 100\mathbb{E}[Z]\,] \leq \frac{1}{100}$$

- If $\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) \geq \theta$, then $\operatorname{pow}_2(\delta) \geq \tau^2$, and by Chebyshev's inequality and Equation (51)

$$\begin{aligned}\Pr[\,Z = 0\,] \leq \Pr[\,|Z - \mathbb{E}[Z]| \geq \mathbb{E}[Z]\,] &\leq \frac{1}{\mathbb{E}[Z]} + \frac{4}{N} \cdot \frac{\operatorname{pow}_3(\delta) + \frac{3}{d}\operatorname{pow}_2(\delta)}{(\operatorname{pow}_2(\delta) + 1/d)^2} \\ &\leq \frac{2}{N(N-1)\tau^2} + \frac{4}{N} \cdot \frac{\operatorname{pow}_2(\delta)^{3/2} + \frac{3}{d}\operatorname{pow}_2(\delta)}{\operatorname{pow}_2(\delta)^2} \\ &\leq \frac{3}{N^2\tau^2} + \frac{4}{N\tau} + \frac{12}{Nd\tau^2} \\ &\leq \frac{3}{N^2\tau^2} + \frac{4}{N\tau} + \frac{12}{N} \qquad (\tau \geq 1/\sqrt{d}) \\ &\leq \frac{3}{N^2\tau^2} + \frac{16}{N\tau}\end{aligned}$$

  which is at most $\frac{1}{100}$ for $N \geq \frac{1601}{\tau}$.

This proves the lemma. ◀

This concludes the proof of the upper bound part of Theorem 12. To conclude, it only remains to show that this is, indeed, optimal. For this, we need a lower bound of [23], which generalized a lower bound of Aaronson and Shi [2]:

▶ **Theorem 14** ([23]). *Let $d > 0$ and $r \geq 2$ be integers such that $r|d$, and let $f : [d] \to [d]$ be a function to which we have quantum oracle access. Then deciding if f is 1-to-1 or r-to-1, promised that one of these holds, requires $\Omega((d/r)^{1/3})$ quantum queries.*

When we view this function as a quantum string oracle for a probability distribution, the function being 1-to-1 corresponds to the uniform distribution on $[d]$. In the other case, the distribution is uniform on a subset of size $[d/r]$, for any $r \geq \theta + 1$ dividing d. An easy calculation shows that the second distribution is at χ^2 divergence

$$\chi^2(\mathbf{p} \mid\mid \mathbf{u}_d) = \sum_{i \in [d]}\left(\frac{\mathbf{p}_i^2}{1/d}\right) - 1 = d \cdot \frac{r^2}{d^2} \cdot \frac{d}{r} - 1 = r - 1 \geq \theta, \qquad (52)$$

from uniform, which completes the proof. ◀

C Reduction from Identity to Uniformity Testing

As mentioned in the introduction, there is a known reduction from identity to uniformity testing, due to Goldreich [16] and inspired by [13]: which, in a blackbox way, converts an instance of uniformity testing (in total variation distance) with reference distribution $\mathbf{q}$ over $[d]$ and distance parameter ε to an instance of uniformity testing over $[4d]$ and distance parameter $\varepsilon/4$. (Here, we follow the exposition and parameter setting of [11, Section 2.2.3].)

To be able to use it in our setting, all we need to check is that this blackbox reduction $\Phi_{\mathbf{q}}$ preserves access to "the code": that is, given the code $C_{\mathbf{p}}$ for a probability distribution $\mathbf{p}$ over $[d]$, that we can efficiently have access to the code $C_{\mathbf{p}'}$ for the resulting distribution $\mathbf{p}' = \Phi_{\mathbf{q}}(\mathbf{p})$ over $[4d]$. To do so, note that $\Phi_{\mathbf{q}}$ is the composition of 3 successive mappings,

$$\Phi_{\mathbf{q}} = \Phi_{\mathbf{q}}^{(1)} \circ \Phi_{\mathbf{q}}^{(2)} \circ \Phi_{\mathbf{q}}^{(3)}$$

where $\Phi_{\mathbf{q}}^{(3)} \colon [d] \to [d]$, $\Phi_{\mathbf{q}}^{(2)} \colon [d] \to [d+1]$, and , $\Phi_{\mathbf{q}}^{(2)} \colon [d+1] \to [4d]$. So it suffices to show that each of these 3 mappings does preserve access to the code generating a sample from the resulting distribution.

- The first, $\Phi_{\mathbf{q}}^{(3)}$, is the easier, as it consists only in mixing its input with the uniform distribution:

  $$\Phi_{\mathbf{q}}^{(3)}(\mathbf{p}) = \frac{1}{2}\mathbf{p} + \frac{1}{2}\mathbf{u}_d$$

  for which a circuit can be easily obtained, given a circuit for $\mathbf{p}$.
- The second, $\Phi_{\mathbf{q}}^{(2)}$, "rounds down" the probability of each of the d elements of the domain, and sends the remaining probability mass to a $(d+1)$-th new element:

  $$\Phi_{\mathbf{q}}^{(2)}(\mathbf{p})_i = \begin{cases} \frac{\lfloor 4d\mathbf{q}_i \rfloor}{4d\mathbf{q}_i} \cdot \mathbf{p}_i, & i \in [d] \\ 1 - \sum_{i=1}^{d} \frac{\lfloor 4d\mathbf{q}_i \rfloor}{4d\mathbf{q}_i} \cdot \mathbf{p}_i, & i = d+1 \end{cases}$$

  This corresponds to adding to the circuit $C_{\mathbf{p}}$ for $\mathbf{p}$ a "postprocessing circuit" which, if the output of $C_{\mathbf{p}}$ is i, outputs i with probability $\frac{\lfloor 4d\mathbf{q}_i \rfloor}{4d\mathbf{q}_i}$ (and $d+1$ otherwise).
- The third, $\Phi_{\mathbf{q}}^{(1)}$, assumes that the reference distribution $\mathbf{q}$ is "grained" (namely, all its probabilities are positive multiples of $1/(4d)$), which will be the case after the first two mappings[6] fully known). Having partitioned $[4d]$ in sets $S_1, \dots, S_d$ where

  $$|S_i| = 4d \cdot \mathbf{q}_i \geq 1$$

  and $\Phi_{\mathbf{q}}^{(1)}$ is given by

  $$\Phi_{\mathbf{q}}^{(3)}(\mathbf{p})_i = \sum_{j=1}^{d} \frac{\mathbf{p}_i}{|S_i|} \mathbb{1}_{\{j \in S_i\}}, \qquad i \in [4d]\,.$$

  This corresponds to adding to the circuit $C_{\mathbf{p}}$ for $\mathbf{p}$ a "postprocessing circuit" which, if the output of $C_{\mathbf{p}}$ is i, outputs an element of S_i uniformly at random. (Importantly, $S_1, \dots, S_d$ are uniquely determined by $\mathbf{q}$, and do not depend on $\mathbf{p}$ or $C_{\mathbf{p}}$ at all.)

To summarize, each of these three mappings can be implemented to provide, given a circuit $C_{\mathbf{p}}$ for $\mathbf{p}$, a circuit $C_{\mathbf{p}'}$ for the output $\mathbf{p}'$, so that altogether the reduction can be implemented in a way which preserves access to "the code."

[6] Specifically, when chaining the three mappings, the reference distribution called $\mathbf{q}$ here is actually $\Phi_{\mathbf{q}}^{(2)} \circ \Phi_{\mathbf{q}}^{(3)}(\mathbf{q})$.