On the Power of Quantum Fourier Sampling

Bill Fefferman* Chris Umans†

Abstract

A line of work initiated by Terhal and DiVincenzo [TD02] and Bremner, Jozsa, and Shepherd [BJS10], shows that restricted classes of quantum computation can efficiently sample from probability distributions that cannot be exactly sampled efficiently on a classical computer, unless the $\mathbf{PH}$ collapses. Aaronson and Arkhipov [AA13] take this further by considering a distribution that can be sampled efficiently by linear optical quantum computation, that under two feasible conjectures, cannot even be approximately sampled classically within bounded total variation distance, unless the $\mathbf{PH}$ collapses.

In this work we use Quantum Fourier Sampling to construct a class of distributions that can be sampled exactly by a quantum computer. We then argue that these distributions cannot be approximately sampled classically, unless the $\mathbf{PH}$ collapses, under variants of the Aaronson-Arkhipov conjectures.

In particular, we show a general class of quantumly sampleable distributions each of which is based on an "Efficiently Specifiable" polynomial, for which a classical approximate sampler implies an average-case approximation. This class of polynomials contains the Permanent but also includes, for example, the Hamiltonian Cycle polynomial, as well as many other familiar $\#\mathbf{P}$-hard polynomials.

Since our distribution likely requires the full power of universal quantum computation, while the Aaronson-Arkhipov distribution uses only linear optical quantum computation with noninteracting bosons, why is this result interesting? We can think of at least three reasons:

1. Since the conjectures required in [AA13] have not yet been proven, it seems worthwhile to weaken them as much as possible. We do this in two ways, by weakening both conjectures to apply to any "Efficiently Specifiable" polynomial, and by weakening the so-called Anti-Concentration conjecture so that it need only hold for one distribution in a broad class of distributions.
2. Our construction can be understood without any knowledge of linear optics. While this may be a disadvantage for experimentalists, in our opinion it results in a very clean and simple exposition that may be more immediately accessible to computer scientists.
3. It is extremely common for quantum computations to employ "Quantum Fourier Sampling" in the following way: first apply a classically efficient function to a uniform superposition of inputs, then apply a Quantum Fourier Transform followed by a measurement. Our distributions are obtained in exactly this way, where the classically efficient function is related to a (presumed) hard polynomial. Establishing rigorously a robust sense in which the central primitive of Quantum Fourier Sampling is classically hard seems a worthwhile goal in itself.

*Joint Center for Quantum Information and Computer Science, University of Maryland/NIST and California Institute of Technology, supported in part by NSF CCF-1423544 and BSF grant 2010120.

†California Institute of Technology, supported by NSF CCF-1423544 and BSF grant 2010120.

1 Introduction

Nearly twenty years after the discovery of Shor's factoring algorithm [Sho94] that caused an explosion of interest in quantum computation, the complexity theoretic classification of quantum computation remains embarrassingly unsettled.

The foundational results of Bernstein and Vazirani [BV97], Adleman, DeMarrais, and Huang [ADH97], and Bennett, Bernstein, Brassard and Vazarani [BBBV97] laid the groundwork for quantum complexity theory by defining $\mathbf{BQP}$ as the class of problems solvable with a quantum computer in polynomial time, and established the upper bound, $\mathbf{BQP} \subseteq \mathbf{PP}$, which hasn't been improved since.

In particular, given that $\mathbf{BPP} \subseteq \mathbf{BQP}$, so quantum computers are surely no less powerful than their classical counterparts, it is natural to compare the power of efficient quantum computation to the power of efficient classical verification. Can every problem with an efficient quantum algorithm be verified efficiently? Likewise can every problem whose solution can be verified efficiently be solved quantumly? In complexity theoretic terms, is $\mathbf{BQP} \subseteq \mathbf{NP}$, and is $\mathbf{NP} \subseteq \mathbf{BQP}$? Factoring is contained in $\mathbf{NP} \cap \mathbf{coNP}$, and so cannot be $\mathbf{NP}$-hard unless $\mathbf{NP} = \mathbf{coNP}$ and the $\mathbf{PH}$ collapses. Thus, while being a problem of profound practical importance, Shor's algorithm does not give evidence that $\mathbf{NP} \subseteq \mathbf{BQP}$.

Even progress towards oracle separations has been agonizingly slow. These same works that defined $\mathbf{BQP}$ established an oracle for which $\mathbf{NP} \not\subset \mathbf{BQP}$ [BBBV97] and $\mathbf{BQP} \not\subset \mathbf{NP}$ [BV97]. This last result can be improved to show an oracle relative to which $\mathbf{BQP} \not\subset \mathbf{MA}$ [BV97], but even finding an oracle relative to which $\mathbf{BQP} \not\subset \mathbf{AM}$ is still wide open. This is particularly troubling given that, under widely believed complexity assumptions, $\mathbf{NP} = \mathbf{MA} = \mathbf{AM}$ [KvM02]. Thus, our failure to provide an oracle relative to which $\mathbf{BQP} \not\subset \mathbf{AM}$ indicates a massive lack of understanding of the classical power of quantum computation.

Recently, two candidate oracle problems with quantum algorithms have been proven to not be contained in the $\mathbf{PH}$, assuming plausible complexity theoretic conjectures [Aar10a, FU11].[1] These advances remain at the forefront of progress on these questions.

A line of work initiated by DiVincenzo and Terhal [TD02], Bremner, Jozsa and Shepherd [BJS10], and Aaronson and Arkhipov [AA13] asks whether we can provide a theoretical basis for quantum superiority by looking at *distribution sampling problems*. In particular, Aaronson and Arkhipov show a *distribution* that can be sampled efficiently by a particular limited form of quantum computation, that assuming the validity of two feasible conjectures, cannot be approximately sampled classically (even by a randomized algorithm with a $\mathbf{PH}$ oracle), unless the $\mathbf{PH}$ collapses. The equivalent result for decision problems, establishing $\mathbf{BQP} \not\subset \mathbf{BPP}$ unless the $\mathbf{PH}$ collapses, would be a crowning achievement in quantum complexity theory. In addition, this research has been very popular not only with the theoretical community, but also with experimentalists who hope to perform this task, "Boson Sampling", in their labs. Experimentally, it seems more pressing to analyze the hardness of approximate quantum sampling, since it is unreasonable to expect that any physical realization of a quantum computer can *itself* exactly sample from the quantum distribution.

Interestingly, it is also known that if we can find such a quantumly sampleable distribution for which no classical approximate sampler exists, there exists a "search" problem that can be solved by a quantum

[1] Although the "Generalized Linial-Nisan" conjecture proposed in [Aar10a] is now known to be false [Aar10b].

computer that cannot be solved classically [Aar10c]. In a search problem we are given an input $x \in \{0,1\}^n$, and our goal is to output an element in a nonempty set, $A_x \subseteq \{0,1\}^{poly(n)}$ with high probability. This would be one of the strongest pieces of evidence to date that quantum computers can outperform their classical counterparts.

In this work we use the same general algorithmic framework used in many quantum algorithms, which we refer to as "Quantum Fourier Sampling", to demonstrate the existence of a general class of distributions that can be sampled exactly by a quantum computer. We then argue that these distributions shouldn't be able to be approximately sampled classically, unless the $\mathbf{PH}$ collapses. Perhaps surprisingly, we obtain and generalize many of the same conclusions as Aaronson and Arkhipov [AA13] with a completely different class of distributions.

Additionally, recently, and independent of us, an exciting result by Bremner, Montanaro and Shepherd [BMS15] obtains similar quantum "approximate sampling" results under related but different conjectures. While our hardness conjectures apply to a broad class of hard "polynomials", their distribution can be sampled by a class of commuting quantum computations known as Instantaneous Quantum Polynomial time, or $\mathbf{IQP}$, whereas our results likely require the full power of universal quantum computation.

2 Overview

2.1 Our Goals

We want to find a class of distributions that can be sampled quantumly that cannot be approximately sampled classically, unless the $\mathbf{PH}$ collapses. A natural methodology toward showing this is to prove that the existence of a classical approximate sampler implies that a $\#\mathbf{P}$-hard function can be computed in the $\mathbf{PH}$. By Toda's Theorem [Tod91], this would imply a collapse of the $\mathbf{PH}$.

In this work, we demonstrate a class of distributions that can be sampled exactly on a quantum computer. We prove that the existence of an approximate sampler for these distributions implies an approximate average case solution to an "Efficiently Specifiable" polynomial. An Efficiently Specifiable polynomial is informally a polynomial in which the variables in each monomial can be computed efficiently from the index of the monomial. This includes, among others, the Permanent and Hamiltonian Cycle polynomial.

Computing a multiplicative approximation to the Permanent with integer entries in the worst-case is $\#\mathbf{P}$-hard, and computing the Permanent on average is $\#\mathbf{P}$-hard (see [AA13] for more details). The challenge to proving our conjectures is to put these two together to prove that an average-case multiplicative approximation to the Permanent (or for that matter, any Efficiently Specifiable polynomial) is still a $\#\mathbf{P}$-hard problem. Since we can't prove these conjectures, and we don't know the ingredients such a proof will require, it seems worthwhile to attempt to generalize the class of distributions that can be sampled quantumly.

2.2 Our Results

In Section 4 we define a general class of distributions that can be sampled exactly on a quantum computer. The probabilities in these distributions are proportional to each different $\{\pm 1\}^n$ evaluation of a particular *Efficiently Specifiable* polynomial (see Definition 2) with n variables. We then show in Section 5 that the existence of an approximate classical sampler for these distributions implies the existence of an *additive approximate average-case* solution to the Efficiently Specifiable polynomial. We generalize this in Section 6 to prove that quantum computers can sample from a class of distributions in which each probability is proportional to polynomially bounded integer evaluations of an Efficiently Specifiable polynomial.

We then attempt to extend this result to quantumly sample from a distribution with probabilities proportional to exponentially bounded integer evaluations of Efficiently Specifiable polynomials. To do this, in Section 7, we introduce a variant of the Quantum Fourier Transform which we call the "Squashed QFT". We explicitly construct this unitary operator, and show how to use it in our quantum sampling framework. We leave as an open question whether this unitary can be realized by an efficient quantum circuit. We then prove in Section 9, using a similar argument to Section 5, that if we had a classical approximate sampler for this distribution we'd have an *additive approximate average-case* solution to the Efficiently Specifiable polynomial with respect to the binomial distribution over exponentially bounded integers.

In Section 10 we conclude with conjectures needed to establish the intractability of approximate classical sampling from any of our quantumly sampleable distributions. As shown in Sections 5 and 6 it suffices to prove that an *additive approximate average-case solution* to any Efficiently Specifiable polynomial is $\#\mathbf{P}$-hard, and we conjecture that this is possible. We also propose an "Anti-concentration conjecture" relative to an Efficiently Specifiable polynomial over the binomial distribution, which allows us to reduce the hardness of a *multiplicative approximate average-case* solution to an *additive approximate average-case solution*. Assuming this second conjecture, we can then base our first conjecture around the hardness of *multiplicative*, rather than *additive approximate average-case solutions* to an Efficiently Specifiable polynomial.

Our conjectures generalize conjectures in Aaronson and Arkhipov's results [AA13]. They conjecture that an *additive approximate average-case solution* to the Permanent with respect to the Gaussian distribution with mean 0 and variance 1 is $\#\mathbf{P}$-hard. They further propose an "Anti-concentration" conjecture which allows them to reduce the hardness of *multiplicative approximate average-case solutions* to the Permanent over the Gaussian distribution to the hardness of *additive average case solutions* to the Permanent over the Gaussian distribution. The parameters of our conjectures match the parameters of theirs, but our conjectures are broader, applying to any Efficiently Specifiable polynomial, a class which includes the Permanent, and a wider class of distributions, and thus is formally easier to prove.

3 Quantum Preliminaries

In this section we cover the basic priciples of quantum computing needed to understand the content in the paper. For a much more complete overview there are many references available, e.g., [KSV02, NC00].

The state of an *n-qubit quantum system* is described by a unit vector in $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$, a 2^n-dimensional complex Hilbert space. As per the literature we will denote the standard orthogonal basis vectors of $\mathcal{H}$ by $\{|v\rangle\}$ for $v \in \{0,1\}^n$.

In accordance with the laws of quantum mechanics, transformations of states are described by unitary transformations acting on $\mathcal{H}$, where a *unitary transformation* over $\mathcal{H}$ is a linear transformation specified by a $2^n \times 2^n$ square complex matrix U, such that $UU^* = I$, where U^* is the conjugate transpose. Equivalently, the rows (and columns) of U form an orthonormal basis. A *local* unitary is a unitary that operates only on $b = O(1)$ qubits; i.e. after a suitable renaming of the standard basis by reordering qubits, it is the matrix $U \otimes I_{2^{n-b}}$, where U is a $2^b \times 2^b$ unitary U. A local unitary can be applied in a single step of a Quantum Computer. A *local decomposition* of a unitary is a factorization into local unitaries. We say a $2^n \times 2^n$ unitary is *efficiently quantumly computable* if this factorization has at most $poly(n)$ factors.

We will need the concept of quantum evaluation of an efficiently classically computable function $f : \{0,1\}^n \to \{0,1\}^m$, which in one quantum query to f maps:

$$\sum_{x \in \{0,1\}^n} |x\rangle|z\rangle \to \sum_{x \in \{0,1\}^n} |x\rangle|z \oplus f(x)\rangle$$

Note that this is a unitary map, as applying it again inverts the procedure, and can be done efficiently as long as f is efficiently computable.

Assuming f is $\{0,1\}$-valued, we can use this state together with a simple phase flip unitary gate to prepare:

$$\sum_{x \in \{0,1\}^n} (-1)^{f(x)} |x\rangle|f(x)\rangle$$

And one more quantum query to f, which "uncomputes" it, and allows us to obtain the state $\sum_{x \in \{0,1\}^n} (-1)^{f(x)}|x\rangle$.

Equivalently, if the efficiently computable function is $f : \{0,1\} \to \{\pm 1\}$ we can think of this as a procedure to prepare:

$$\sum_{x \in \{0,1\}^n} f(x)|x\rangle$$

With two quantum queries to the function f.

We close this section with an additional lemma needed for our quantum sampler.

Lemma 1. *Let $h : [m] \to \{0,1\}^n$ be an efficiently computable one-to-one function, and suppose its inverse can also be efficiently computed. Then the superposition $\frac{1}{\sqrt{m}} \sum_{x \in [m]} |h(x)\rangle$ can be efficiently prepared by a quantum algorithm.*

Proof. Our quantum procedure with two quantum registers proceeds as follows:

1. Prepare $\frac{1}{\sqrt{m}} \sum_{x \in [m]} |x\rangle|00...0\rangle$
2. Query h using the first register as input and the second as output:

$$\frac{1}{\sqrt{m}} \sum_{x \in [m]} |x\rangle|h(x)\rangle$$

3. Query h^{-1} using the second register as input and the first as output:

$$\frac{1}{\sqrt{m}} \sum_{x \in [m]} |x \oplus h^{-1}(h(x))\rangle |h(x)\rangle = \frac{1}{\sqrt{m}} \sum_{x \in [m]} |00...0\rangle |h(x)\rangle$$

4. Discard first register

□

4 Efficiently Specifiable Polynomial Sampling on a Quantum Computer

In this section we describe a general class of distributions that can be sampled efficiently on a Quantum Computer.

Definition 2 (Efficiently Specifiable Polynomial)**.** *We say a multilinear homogenous n-variate polynomial Q with coefficients in $\{0,1\}$ and m monomials is Efficiently Specifiable via an efficiently computable, one-to-one function $h : [m] \to \{0,1\}^n$, with an efficiently computable inverse, if:*

$$Q(X_1, X_2..., X_n) = \sum_{z \in [m]} {X_1}^{h(z)_1} {X_2}^{h(z)_2} ... {X_n}^{h(z)_n}$$

Definition 3 ($\mathcal{D}_{Q,\ell}$)**.** *Suppose Q is an Efficiently Specifiable polynomial with m monomials. For fixed Q and ℓ, we define the class of distributions $\mathcal{D}_{Q,\ell}$ over ℓ-ary strings $y \in [0, \ell-1]^n$ given by:*

$$\Pr_{\mathcal{D}_{Q,\ell}}[y] = \frac{|Q(Z_y)|^2}{\ell^n m}$$

Where $Z_y \in \mathbb{T}_\ell^n$ is a vector of complex values encoded by the string y.

Theorem 4 (Quantum Sampling Theorem)**.** *Given an Efficiently Specifiable polynomial, Q with n variables, m monomials, relative to a function h, and $\ell \leqslant exp(n)$, the resulting $\mathcal{D}_{Q,\ell}$ can be sampled in $poly(n)$ time on a Quantum Computer.*

Proof.

1. We start in a uniform superposition $\frac{1}{\sqrt{m}} \sum_{z \in [m]} |z\rangle$.
2. We then apply Lemma 1 to prepare $\frac{1}{\sqrt{m}} \sum_{z \in [m]} |h(z)\rangle$.
3. Apply Quantum Fourier Transform over $\mathbb{Z}_\ell^n$ to attain $\frac{1}{\sqrt{\ell^n m}} \sum_{y \in [0,\ell-1]^n} \sum_{z \in [m]} \omega_\ell^{<y,h(z)>} |y\rangle$

Notice that the amplitude of each y basis state in the final state after Step 3 is proportional to the value of $Q(Z_y)$, as desired.

□

5 Classical Hardness of Efficiently Specifiable Polynomial Sampling

We are interested in demonstrating the existence of some distribution that can be sampled exactly by a uniform family of quantum circuits, that cannot be sampled approximately classically. Approximate here means close in Total Variation distance, where we denote the Total Variation distance between two distributions X and Y by $\|X - Y\|$. Thus we define the notion of a Sampler to be a classical algorithm that approximately samples from a given class of distributions:

Definition 5 (Sampler)**.** *Let $\{D_n\}_{n>0}$ be a class of distributions where each D_n is distributed over $\mathbb{C}^n$. Let $r(n) \in poly(n), \epsilon(n) \in 1/poly(n)$. We say S is a Sampler with respect to $\{D_n\}$ if $\|S(0^n, x \sim U_{\{0,1\}^{r(n)}}, 0^{1/\epsilon(n)}) - D_n\| \leqslant \epsilon(n)$ in (classical) polynomial time.*

We first recall a theorem due to Stockmeyer [Sto85] on the ability to "approximate count" in the $\mathbf{PH}$.

Theorem 6 (Stockmeyer [Sto85])**.** *Given as input a function $f : \{0,1\}^n \to \{0,1\}^m$ and $y \in \{0,1\}^m$, there is a procedure that outputs α such that:*

$$(1-\epsilon) \Pr_{x \sim U_{\{0,1\}^n}}[f(x) = y] \leqslant \alpha \leqslant (1+\epsilon) \Pr_{x \sim U_{\{0,1\}^n}}[f(x) = y]$$

In randomized time $poly(n, 1/\epsilon)$ with access to an $\mathbf{NP}$ oracle.

In this section we use Theorem 6, together with the assumed existence of a Sampler for $\mathcal{D}_{Q,\ell}$ to obtain hardness consequences.

In particular, we show that a Sampler would imply the existence of an efficient approximation to an Efficiently Specifiable polynomial in the following two contexts:

Definition 7 ($\epsilon-$additive δ-approximate solution)**.** *Given a distribution D over $\mathbb{C}^n$ and $P : \mathbb{C}^n \to \mathbb{C}$ we say $T : \mathbb{C}^n \to \mathbb{C}$ is an $\epsilon-$additive approximate $\delta-$average case solution with respect to D, to $P : \mathbb{C}^n \to \mathbb{C}$, if $\Pr_{x \sim D}[|T(x) - P(x)| \leqslant \epsilon] \geqslant 1 - \delta$.*

Definition 8 ($\epsilon-$multiplicative δ-approximate solution)**.** *Given a distribution D over $\mathbb{C}^n$ and a function $P : \mathbb{C}^n \to \mathbb{C}$ we say $T : \mathbb{C}^n \to \mathbb{C}$ is an $\epsilon-$multiplicative approximate $\delta-$average case solution with respect to D, if $\Pr_{x \sim D}[|T(x) - P(x)| \leqslant \epsilon|P(x)|] \geqslant 1 - \delta$.*

These definitions formalize a notion that we will need, in which an efficient algorithm computes a particular hard function approximately only on most inputs, and can act arbitrarily on a small fraction of remaining inputs.

In this section, we focus on the uniform distribution on $\{\pm 1\}$ strings, and a natural generalization:

Definition 9 ($\mathbb{T}_\ell$)**.** *Given $\ell > 0$, we define the set $\mathbb{T}_\ell = \{\omega_\ell^0, \omega_\ell^1 ..., \omega_\ell^{\ell-1}\}$ where ω_ℓ is a primitive ℓ-th root of unity.*

We note that $\mathbb{T}_\ell$ is just ℓ evenly spaced points on the unit circle, and $\mathbb{T}_2 = \{\pm 1\}$.

Theorem 10 (Complexity consequences of Sampler)**.** *Given an Efficiently Specifiable polynomial Q with n variables and m monomials, and a Sampler S with respect to $\mathcal{D}_{Q,\ell}$, there is a randomized procedure $T : \mathbb{C}^n \to \mathbb{C}$, an $(\epsilon \cdot m)-$additive approximate $\delta-$average case solution with respect to the uniform distribution over $\mathbb{T}_\ell^n$, to the $|Q|^2$ function, that runs in randomized time $poly(n, 1/\epsilon, 1/\delta)$ with access to an* **NP** *oracle.*

Proof. We need to give a procedure that outputs an ϵm-additive estimate to the $|Q|^2$ function evaluated at a uniform setting of the variables, with probability $1 - \delta$ over choice of setting. Setting $\beta = \frac{\epsilon\delta}{16}$, suppose S samples from a distribution $\mathcal{D}'$ such that $\|\mathcal{D}_{Q,\ell} - \mathcal{D}'\| \leqslant \beta$. We let p_y be $\Pr_{\mathcal{D}_{Q,\ell}}[y]$ and q_y be $\Pr_{\mathcal{D}'}[y]$.

Our procedure picks a uniformly chosen encoding of a setting $y \in [0, \ell - 1]^n$, and outputs an estimate $\tilde{q}_y$. Note that $p_y = \frac{|Q(Z_y)|^2}{\ell^n m}$. Thus our goal will be to output a $\tilde{q}_y$ that approximates p_y within additive error $\epsilon \frac{m}{\ell^n m} = \frac{\epsilon}{\ell^n}$, in time polynomial in n, $\frac{1}{\epsilon}$, and $\frac{1}{\delta}$.

We need:

$$\Pr_y[|\tilde{q}_y - p_y| > \frac{\epsilon}{\ell^n}] \leqslant \delta$$

First, define for each y, $\Delta_y = |p_y - q_y|$, and so $\|\mathcal{D}_{Q,\ell} - \mathcal{D}'\| = \frac{1}{2}\sum_y [\Delta_y]$.

Note that:

$$E_y[\Delta_y] = \frac{\sum_y [\Delta_y]}{\ell^n} = \frac{2\beta}{\ell^n}$$

And applying Markov's inequality, $\forall k > 1$,

$$\Pr_y[\Delta_y > \frac{k2\beta}{\ell^n}] < \frac{1}{k}$$

Setting $k = \frac{4}{\delta}, \beta = \frac{\epsilon\delta}{16}$, we have,

$$\Pr_y[\Delta_y > \frac{\epsilon}{2} \cdot \frac{1}{\ell^n}] < \frac{\delta}{4}$$

Then use approximate counting (with an $\mathbf{NP}$ oracle), using Theorem 6 on the randomness of S to obtain an output $\tilde{q}_y$ so that, for all $\gamma > 0$, in time polynomial in n and $\frac{1}{\gamma}$:

$$\Pr[|\tilde{q}_y - q_y| > \gamma \cdot q_y] < \frac{1}{2^n}$$

Because we can amplify the failure probability of Stockmeyer's algorithm to be inverse exponential. Note that:

$$E_y[q_y] = \frac{\sum\limits_y q_y}{\ell^n} = \frac{1}{\ell^n}$$

Thus,

$$\Pr_y[q_y > \frac{k}{\ell^n}] < \frac{1}{k}$$

Now, setting $\gamma = \frac{\epsilon\delta}{8}$ and applying the union bound:

$$\begin{aligned}
\Pr_y[|\tilde{q}_y - p_y| > \frac{\epsilon}{\ell^n}] &\leqslant \Pr_y[|\tilde{q}_y - q_y| > \frac{\epsilon}{2} \cdot \frac{1}{\ell^n}] + \Pr_y[|q_y - p_y| > \frac{\epsilon}{2} \cdot \frac{1}{\ell^n}] \\
&\leqslant \Pr_y[q_y > \frac{k}{\ell^n}] + \Pr[|\tilde{q}_y - q_y| > \gamma \cdot q_y] + \Pr_y[\Delta_y > \frac{\epsilon}{2} \cdot \frac{1}{\ell^n}] \\
&\leqslant \frac{1}{k} + \frac{1}{2^n} + \frac{\delta}{4} \\
&\leqslant \frac{\delta}{2} + \frac{1}{2^n} \leqslant \delta.
\end{aligned}$$

□

Now, as will be proven in Appendix B, the variance, $\mathrm{Var}\,[Q(X)]$, of the distribution over $\mathbb{C}$ induced by an Efficiently Specifiable Q with m monomials, evaluated at uniformly distributed entries over $\mathbb{T}_\ell^n$ is m, and so the preceding Theorem 10 promised us we can achieve an $\epsilon\mathrm{Var}\,[Q(X)]$-additive approximation to Q^2, given a Sampler. We now show that, under a conjecture, this approximation can be used to obtain a good multiplicative estimate to Q^2. This conjecture effectively states that the Chebyshev inequality for this random variable is tight.

Conjecture 1 (Anti-Concentration Conjecture relative to an n-variate polynomial Q and distribution $\mathcal{D}$ over $\mathbb{C}^n$)**.** *There exists a polynomial p such that for all n and $\delta > 0$,*

$$\Pr_{X \sim \mathcal{D}}\left[|Q(X)|^2 < \frac{\mathrm{Var}\,[Q(X)]}{p(n, 1/\delta)}\right] < \delta$$

Theorem 11. *Assuming Conjecture 1, relative to an Efficiently Specifiable polynomial Q and a distribution $\mathcal{D}$, an $\epsilon \mathrm{Var}\left[Q(X)\right]$-additive approximate δ-average case solution with respect to D, to the $|Q|^2$ function can be used to obtain an $\epsilon' \leqslant poly(n)\epsilon$-multiplicative approximate $\delta' = 2\delta$-average case solution with respect to $\mathcal{D}$ to $|Q|^2$.*

Proof. Suppose λ is, with high probability, an $\epsilon \mathrm{Var}\left[Q(X)\right]$-additive approximation to $|Q(X)|^2$, as guaranteed in the statement of the Theorem. This means:

$$\Pr_{X\sim\mathcal{D}}\left[\left|\lambda - |Q(X)|^2\right| > \epsilon \mathrm{Var}\left[Q(X)\right]\right] < \delta$$

Now assuming Conjecture 1 with polynomial p, we will show that λ is also a good multiplicative approximation to $|Q(X)|^2$ with high probability over X.

By the union bound,

$$\begin{aligned}\Pr_{X\sim\mathcal{D}}\left[\frac{\left|\lambda - |Q(X)|^2\right|}{\epsilon p(n,1/\delta)} > |Q(X)|^2\right] \leqslant & \Pr_{X\sim D}\left[\left|\lambda - |Q(X)|^2\right| > \epsilon \mathrm{Var}\left[Q(X)\right]\right] + \\ & \Pr_{X\sim\mathcal{D}}\left[\frac{\epsilon \mathrm{Var}\left[Q(X)\right]}{\epsilon p(n,1/\delta)} > |Q(X)|^2\right] \\ & \leqslant 2\delta\end{aligned}$$

Where the second line comes from Conjecture 1. Thus we can achieve any desired multiplicative error bounds (ϵ', δ') by setting $\delta = \delta'/2$ and $\epsilon = \epsilon'/p(n, 1/\delta)$.

□

For the results in this section to be meaningful, we simply need the Anti-Concentration conjecture to hold for some Efficiently Specifiable polynomial that is $\#\mathbf{P}$-hard to compute, relative to any distribution we can sample from (either U_n, or $\mathcal{B}(0,k)^n$). We note that Aaronson and Arkhipov [AA13] conjectures the same statement as Conjecture 1 for the special case of the $\mathbf{Permanent}$ function relative to matrices with entries distributed independently from the complex Gaussian distribution of mean 0 and variance 1.

Additionally, we acknowledge a result of Tao and Vu who show:

Theorem 12 (Tao & Vu [TV08])**.** *For all $\epsilon > 0$ and sufficiently large n,*

$$\Pr_{X\in\{\pm 1\}^{n\times n}}\left[|\mathbf{Permanent}[X]| < \frac{\sqrt{n!}}{n^{\epsilon n}}\right] < \frac{1}{n^{0.1}}$$

Which comes quite close to our conjecture for the case of the $\mathbf{Permanent}$ function and uniformly distributed $\{\pm 1\}^{n\times n} = \mathbb{T}_2^{n\times n}$ matrix. More specifically, for the above purpose of relating the additive hardness to the multiplicative, we would need an upper bound of any inverse polynomial δ, instead of a fixed $n^{-0.1}$.

6 Sampling from Distributions with Probabilities Proportional to $[-k, k]$ Evaluations of Efficiently Specifiable Polynomials

In the prior sections we discussed quantum sampling from distributions in which the probabilities are proportional to evaluations of Efficiently Specifiable polynomials evaluated at points in $\mathbb{T}_\ell^n$. In this section we show how to generalize this to quantum sampling from distributions in which the probabilities are proportional to evaluations of Efficiently Specifiable polynomials evaluated at polynomially bounded integer values. In particular, we show a simple way to take an Efficiently Specifiable polynomial with n variables and create another Efficiently Specifiable polynomial with kn variables, in which evaluating this new polynomial at $\{-1, +1\}^{kn}$ is equivalent to evaluation of the old polynomial at $[-k, k]^n$.

Definition 13 (k-valued equivalent polynomial)**.** *For every Efficiently Specifiable polynomial Q with m monomials and every fixed $k > 0$ consider the polynomial $Q'_k : \mathbb{T}_2^{kn} \to \mathbb{R}$ defined by replacing each variable x_i in Q with the sum of k new variables $x_i^{(1)} + x_i^{(2)} + ... + x_i^{(k)}$. We will call Q'_k the k-valued equivalent polynomial with respect to Q.*

Note that a uniformly chosen $\{\pm 1\}$ assignment to the variables in Q'_k induces an assignment to the variables in Q, distributed from a distribution we call $\mathcal{B}(0, k)$:

Definition 14 ($\mathcal{B}(0, k)$)**.** *For k a positive integer, we define the distribution $\mathcal{B}(0, k)$ supported over the odd integers in the range $[-k, k]$ (if k is odd), or even integers in the range $[-k, k]$ (if k is even), so that:*

$$\Pr_{\mathcal{B}(0,k)}[y] = \begin{cases} \frac{\binom{k}{\frac{k+y}{2}}}{2^k} & if\ y\ and\ k\ are\ both\ odd\ or\ both\ even \\ 0 & otherwise \end{cases}$$

Theorem 15. *Given an Efficiently Specifiable polynomial Q with n variables and m monomials, let Q'_k be its k-valued equivalent polynomial. For all $\ell < exp(n)$, we can quantumly sample from the distribution $\mathcal{D}_{Q'_k,\ell}$ in time $poly(n, k)$.*

Proof. Our proof follows from the following lemma, which proves that Q'_k is Efficiently Specifiable.

Lemma 16. *Suppose Q is an n-variate, homogeneous degree d Efficiently Specifiable polynomial with m monomials relative to a function $h : [m] \to \{0, 1\}^n$. Let $k \leqslant poly(n)$ and let Q'_k be the k-valued equivalent polynomial with respect to Q. Then Q'_k is Efficiently Specifiable with respect to an efficiently computable function $h' : [m] \times [k]^d \to \{0, 1\}^{kn}$.*

Proof. We first define and prove that h' is efficiently computable. We note that if there are m monomials in Q, there are mk^d monomials in Q'. As before, we'll think of the new variables in Q'_k as indexed by a pair of indices, a "top index" in $[k]$ and a "bottom index" in $[n]$. Equivalently we are labeling each variable in Q'_k as $x_i^{(j)}$, the j-th copy of the i-th variable in Q. We are given $x \in [m]$ and $y_1, y_2, ..., y_d \in [k]$. Then, for all $i \in [n]$ and $j \in [k]$, we define the output, $z = h'(x, y_1, y_2, ..., y_d)_{i,j} = 1$ iff:

1. $h(x)_i = 1$
2. If $h(x)_i$ is the $\ell \leqslant d$-th non-zero element of $h(x)$, then we require $y_\ell = j$

We will now show that h'^{-1} is efficiently computable. As before we will think of $z \in \{0,1\}^{kn}$ as being indexed by a pair of indices, a 'top index" in $[k]$ and a "bottom index" in $[n]$. Then we compute $h'^{-1}(z)$ by first obtaining from z the bottom indices $j_1, j_2, ..., j_d$ and the corresponding top indices, $i_1, i_2, ..., i_d$. Then obtain from the bottom indices the string $x \in \{0,1\}^n$ corresponding to the indices of variables used in Q and output the concatenation of $h^{-1}(x)$ and $j_1, j_2, ..., j_d$.

□

Theorem 15 now follows from Lemma 16, where we established that Q'_k is Efficiently Specifiable, and Theorem 4, where we established that we can sample from $\mathcal{D}_{Q'_k,\ell}$ quantumly. □

Theorem 17. *Let* $\mathrm{Var}\left[Q(X)\right] = \mathrm{Var}\left[Q(X_1, X_2, ..., X_n)\right]$ *denote the variance of the distribution over* $\mathbb{R}$ *induced by* Q *with assignments distributed from* $\mathcal{B}(0,k)^n$*. Given a Sampler* S *with respect to* $\mathcal{D}_{Q'_k,2}$*, we can find a randomized procedure* $T : \mathbb{R}^n \to \mathbb{R}$*, an* $\epsilon\mathrm{Var}\left[Q(X)\right]$*-additive approximate* δ*-average case solution to* Q^2 *with respect to* $\mathcal{B}(0,k)^n$ *that runs in time* $poly(n, 1/\epsilon, 1/\delta)$ *with access to an* **NP** *oracle.*

Proof. We begin by noting that Q'_k is a polynomial of degree d that has kn variables and $m' = mk^d$ monomials. By Theorem 10 we get that a Sampler with respect to $\mathcal{D}_{Q'_{k,2}}$ implies there exists A, an $\epsilon m'$-additive approximate δ-average case solution to ${Q'_k}^2$ with respect to $U_{\{\pm 1\}^{kn}}$ that runs in time $poly(n, 1/\epsilon, 1/\delta)$ with access to an **NP** oracle. We need to show the existence of an A', an $\epsilon m'$-additive approximate δ-average case solution to ${Q'_k}^2$ with respect to the $\mathcal{B}(0,k)^n$ distribution.

We think of A' as receiving an input, $z \in [-k,k]^n$ drawn from $\mathcal{B}(0,k)^n$. A' picks y uniformly from the orbit of z over $\{\pm 1\}^{kn}$ and outputs $A(y)$. Now:

$$\Pr_{z\sim\mathcal{B}(0,k)^n}\left[\left|A'(z) - Q^2(z)\right| \leqslant \epsilon m'\right] = \Pr_{z\sim\mathcal{B}(0,k)^n, y\sim_R orbit(z)}\left[\left|A(y) - Q^2(z)\right| \leqslant \epsilon m'\right] \tag{1}$$

$$= \Pr_{y\sim U_{\{\pm 1\}^{kn}}}\left[\left|A(y) - Q'_k(y)\right| \leqslant \epsilon m'\right] \geqslant 1 - \delta \tag{2}$$

$$\tag{3}$$

Thus, because a uniformly chosen $\{\pm 1\}^{kn}$ assignment to the variables in Q'_k induces a $\mathcal{B}(0,k)^n$ distributed assignment to the variables in Q, this amounts to an $\epsilon m'$-additive approximate δ-average case solution to Q^2 with respect to $\mathcal{B}(0,k)^n$. In Appendix B we prove that $\mathrm{Var}\left[Q(X)\right]$ is m' as desired. □

7 The "Squashed" QFT

In this section we begin to prove that Quantum Computers can sample efficiently from distributions with probabilities proportional to evaluations of Efficiently Specifiable polynomials at points in $[-k,k]^n$ for $k = exp(n)$. Note that in the prior quantum algorithm of Section 4 we would need to invoke the QFT over $\mathbb{Z}_2^{kn}$, of dimension doubly-exponential in n. Thus we need to define a new Polynomial Transform that can be obtained from the standard Quantum Fourier Transform over $\mathbb{Z}_2^n$, which we refer to as the "Squashed QFT". Now we describe the unitary matrix which implements the Squashed QFT.

Consider the $2^k \times 2^k$ matrix D_k, whose columns are indexed by all possible 2^k multilinear monomials of the variables $x_1, x_2, ..., x_k$ and the rows are indexed by the 2^k different $\{-1, +1\}$ assignments to the variables. The (i, j)-th entry is then defined to be the evaluation of the j-th monomial on the i-th assignment. We note in passing that, defining $\bar{D}_k$ to be the matrix whose entries are the entries in D_k normalized by $1/\sqrt{2^k}$ gives us the Quantum Fourier Transform matrix over $\mathbb{Z}_2^k$. It is clear, by the unitarity of the Quantum Fourier Transform, that the columns (and rows) in D_k are pairwise orthogonal.

Now we define the "Elementary Symmetric Polynomials":

Definition 18 (Elementary Symmetric Polynomials)**.** *We define the j-th Elementary Symmetric Polynomial on k variables for $j \in [0, k]$ to be:*

$$p_j(X_1, X_2, ..., X_k) = \sum_{1 \leqslant \ell_1 < \ell_2 < ... < \ell_j \leqslant k} X_{\ell_1} X_{\ell_2} ... X_{\ell_j}$$

In this work we will care particularly about the first two elementary symmetric polynomials, p_0 and p_1 which are defined as $p_0(X_1, X_2, ..., X_k) = 1$ and $p_1(X_1, X_2, ..., X_k) = \sum_{1 \leqslant \ell \leqslant k} X_\ell$.

Consider the $(k + 1) \times (k + 1)$ matrix, $\tilde{D}_k$, whose columns are indexed by elementary symmetric polynomials on k variables and whose rows are indexed by equivalence classes of assignments in $\mathbb{Z}_2^k$ under S_k symmetry. We obtain $\tilde{D}_k$ from D_k using two steps.

First obtain a $2^k \times (k + 1)$ rectangular matrix $\tilde{D}_k^{(1)}$ whose rows are indexed by assignments to the variables $x_1, x_2, ..., x_k \in \{\pm 1\}^k$ and columns are the entry-wise sum of the entries in each column of D_k whose monomial is in each respective elementary symmetric polynomial.

Then obtain the final $(k+1) \times (k+1)$ matrix $\tilde{D}_k$ by taking $\tilde{D}_k^{(1)}$ and keeping only one representative row in each equivalence class of assignments under S_k symmetry. We label the equivalence classes of assignments under S_k symmetry $o_0, o_1, o_2, ..., o_k$ and note that for each $i \in [k]$, $|o_i| = \binom{k}{i}$. Observe that $\tilde{D}_k$ is precisely the matrix whose (i, j)-th entry is the evaluation of the j-th symmetric polynomial evaluated on an assignment in the i-th symmetry class.

Theorem 19. *The columns in the matrix $\tilde{D}_k^{(1)}$ are pairwise orthogonal.*

Proof. Note that each column in the matrix $\tilde{D}_k^{(1)}$ is the sum of columns in D_k each of which are orthogonal. We can prove this theorem by observing that if we take any two columns in $D_k^{(1)}$, called c_1, c_2, where c_1 is the sum of columns $\{u_i\}$ of D_k and c_2 is the sum of columns $\{v_i\}$ of D_k. The inner product, $\langle c_1, c_2 \rangle$ can be written:

$$\langle \sum_i u_i, \sum_j v_j \rangle = \sum_{i,j} \langle u_i, v_j \rangle = 0$$

□

Theorem 20. *Let L be the $(k+1)\times(k+1)$ diagonal matrix with i-th entry equal to $\sqrt{o_i}$. Then the columns of $L\cdot\tilde{D}_k$ are orthogonal.*

Proof. Note that the value of the symmetric polynomial at each assignment in an equivalence class is the same. We have already concluded the orthogonality of columns in $\tilde{D}_k^{(1)}$. Therefore if we let a and b be any two columns in the matrix $\tilde{D}_k$, and their respective columns be $\bar{a},\bar{b}$ in $\tilde{D}_k^{(1)}$, we can see:

$$\sum_{i=0}^{k}(a_ib_i|o_i|)=\sum_{i=0}^{2^k}\bar{a}_i\bar{b}_i=0$$

From this we conclude that the columns of the matrix $L\cdot\tilde{D}_k$, in which the i-th row of $\tilde{D}_k$ is multiplied by $\sqrt{o_i}$, are orthogonal. □

Theorem 21. *We have just established that the columns in the matrix $L\cdot\tilde{D}_k$ are orthogonal. Let the $k+1\times k+1$ diagonal matrix R be such that so that the columns in $L\cdot\tilde{D}_k\cdot R$ are orthonormal, and thus $L\cdot\tilde{D}_k\cdot R$ is unitary. Then the first two nonzero entries in R, which we call r_0,r_1, corresponding to the normalization of the column pertaining to the zero-th and first elementary symmetric polynomial, are $1/\sqrt{2^k}$ and $\frac{1}{\sqrt{\sum_{i=0}^{k}\left[\binom{k}{i}(k-2i)^2\right]}}$.*

Proof. First we calculate r_0. Since we wish for a unitary matrix, we want the ℓ_2 norm of the first column of $\tilde{D}_k$ to be 1, and so need:

$$r_0^2\sum_{i=0}^{k}\left(\sqrt{o_i}\right)^2=r_0^2\sum_{i=0}^{k}\binom{k}{i}=1$$

And so r_0 is $1/\sqrt{2^k}$ as desired.

Now we calculate r_1, the normalization in the column of $\tilde{D}_k$ corresponding to the first elementary symmetric polynomial. Note that in i-th equivalence class of assignments we have exactly i negative ones and $k-i$ positive ones. Thus the value of the first symmetric polynomial is the sum of these values, which for the $i-th$ equivalence class is precisely $k-2i$. Then we note the normalization in each row is $\sqrt{\binom{k}{i}}$. Thus we have

$$r_1^2\sum_{i=0}^{k}\left[\sqrt{\binom{k}{i}}\,(k-2i)\right]^2=1$$

Thus $r_1=\frac{1}{\sqrt{\sum_{i=0}^{k}\left[\binom{k}{i}(k-2i)^2\right]}}$ as desired.

□

8 Using our "Squashed QFT" to Quantumly Sample from Distributions of Efficiently Specifiable Polynomial Evaluations

In this section we use the unitary matrix developed earlier to quantumly sample distributions with probabilities proportional to evaluations of Efficiently Specifiable polynomials at points in $[-k,k]^n$ for $k = exp(n)$. Here we assume that we have an efficient quantum circuit decomposition for this unitary. The prospects for this efficient decomposition are discussed in Section 10.

For convenience, we'll define a map $\psi : [-k,k] \to [0,k]$, for k even, with

$$\psi(y) = \begin{cases} \frac{k+y}{2} & if\ y\ is\ even \\ 0 & otherwise \end{cases}$$

Definition 22. *Suppose Q is an Efficiently Specifiable polynomial Q with n variables and m monomials, and, for $k \leqslant exp(n)$, let Q'_k be its k-valued equivalent polynomial. Let $\mathrm{Var}\left[Q(X)\right]$ be the variance of the distribution over $\mathbb{R}$ induced by Q with assignments to the variables distributed over $\mathcal{B}(0,k)^n$ (or equivalently, we can talk about $\mathrm{Var}\left[Q'_k\right]$ where each variable in Q'_k is independently uniformly chosen from $\{\pm 1\}$), as calculated in Appendix B. Then we define the of distribution $\mathcal{D}_{Q,k}$ over n tuples of integers in $[-k,k]$ by:*

$$\Pr_{\mathcal{D}_{Q,k}}[y] = \frac{Q(y)^2 \binom{k}{\psi(y_1)} \binom{k}{\psi(y_2)} ... \binom{k}{\psi(y_n)}}{2^{kn} \mathrm{Var}\left[Q(X)\right]}$$

Theorem 23. *By applying $(L \cdot \tilde{D}_k \cdot R)^{\otimes n}$ in place of the Quantum Fourier Transform over $\mathbb{Z}_2^n$ in Section 4 we can efficiently quantumly sample from $\mathcal{D}_{Q,k}$.*

Proof. Since we are assuming Q is Efficiently Specifiable, let $h : [m] \to \{0,1\}^n$ be the invertible function describing the variables in each monomial. We start by producing the state over $k+1$ dimensional qudits:

$$\frac{1}{\sqrt{m}} \sum_{z \in [m]} |h(z)\rangle$$

Which we prepare via the procedure described in Lemma 1.

Instead of thinking of h as mapping an index of a monomial from $[m]$ to the variables in that monomial, we now think of h as taking an index of a monomial in Q to a polynomial expressed in the $\{1, x^{(1)} + x^{(2)} + ... + x^{(k)}\}^n$ basis.

Now take this state and apply the unitary (which we assume can be realized by an efficient quantum circuit) $(L \cdot \tilde{D}_k \cdot R)^{\otimes n}$.

Notice each $y \in [-k,k]^n$ has an associated amplitude:

$$\alpha_y = \frac{r_0^{n-d} r_1^d Q(y) \sqrt{\binom{k}{\psi(y_1)} \binom{k}{\psi(y_2)} ... \binom{k}{\psi(y_n)}}}{\sqrt{m}}$$

Letting $p_y = \Pr_{\mathcal{D}_{Q,k}}[y]$, note that, by plugging in r_0, r_1 from Section 7:

$$\begin{aligned}\alpha_y^2 &= \frac{Q(y)^2\binom{k}{\psi(y_1)}\binom{k}{\psi(y_2)}\cdots\binom{k}{\psi(y_n)}r_0^{2(n-d)}r_1^{2d}}{m} \\ &= \frac{Q(y)^2\binom{k}{\psi(y_1)}\binom{k}{\psi(y_2)}\cdots\binom{k}{\psi(y_n)}}{m2^{k(n-d)}\left(\sum_{i=0}^{k}\left[\binom{k}{i}(k-2i)^2\right]\right)^d} \\ &= \frac{Q(y)^2\binom{k}{\psi(y_1)}\binom{k}{\psi(y_2)}\cdots\binom{k}{\psi(y_n)}}{2^{kn-kd}\mathrm{Var}\left[Q(X)\right]2^{kd}} = \frac{Q(y)^2\binom{k}{\psi(y_1)}\binom{k}{\psi(y_2)}\cdots\binom{k}{\psi(y_n)}}{2^{kn}\mathrm{Var}\left[Q(X)\right]} = p_y\end{aligned}$$

□

9 The Hardness of Classical Sampling from the Squashed Distribution

In this section, as before, we use Stockmeyer's Theorem (Theorem 6), together with the assumed existence of a Sampler for $\mathcal{D}_{Q,k}$ to obtain hardness consequences for classical sampling with $k \leqslant exp(n)$.

Theorem 24. *Given an Efficiently Specifiable polynomial Q with n variables and m monomials, let Q'_k be its k-valued equivalent polynomial, for some fixed $k \leqslant exp(n)$. Suppose we have a Sampler S with respect to our quantumly sampled distribution class, $\mathcal{D}_{Q,k}$, and let $\mathrm{Var}\left[Q(X)\right]$ denote the variance of the distribution over $\mathbb{R}$ induced by Q with assignments distributed from $\mathcal{B}(0,k)^n$. Then we can find a randomized procedure $T : \mathbb{R}^n \to \mathbb{R}$, an $\epsilon\mathrm{Var}\left[Q(X)\right]$-additive approximate δ-average case solution to Q^2 with respect to $\mathcal{B}(0,k)^n$ that runs in time $poly(n, 1/\epsilon, 1/\delta)$ with access to an* **NP** *oracle.*

Proof. Setting $\beta = \epsilon\delta/16$, suppose S samples from a class of distributions $\mathcal{D}'$ so that $\|\mathcal{D}_{Q,k} - \mathcal{D}'\| \leqslant \beta$. Let $q_y = \Pr_{\mathcal{D}'}[y]$.

We define $\phi : \{\pm 1\}^{kn} \to [-k,k]^n$ to be the map from each $\{\pm 1\}^{kn}$ assignment to its equivalence class of assignments, which is n blocks of even integral values in the interval $[-k,k]$. Note that, given a uniformly random $\{\pm 1\}^{kn}$ assignment, ϕ induces the $\mathcal{B}(0,k)$ distribution over $[-k,k]^n$.

Our procedure picks a $y \in [-k,k]^n$ distributed[2] via $\mathcal{B}(0,k)^n$, and outputs an estimate $\tilde{q}_y$. Equivalently, we analyze this procedure by considering a uniformly distributed $x \in \{\pm 1\}^{kn}$ and then returning an approximate count, $\tilde{q}_{\phi(x)}$ to $q_{\phi(x)}$. We prove that our procedure runs in time $poly(n, 1/\epsilon, 1/\delta)$ with the guarantee that:

$$\Pr_x\left[\frac{|\tilde{q}_{\phi(x)} - p_{\phi(x)}|}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \frac{\epsilon}{2^{kn}}\right] \leqslant \delta$$

[2]We can do this when $k = exp(n)$ by approximately sampling from the Normal distribution, with only $poly(n)$ bits of randomness, and using this to approximate $\mathcal{B}(0,k)$ to within additive error $1/poly(n)$ e.g., [BM58, Ber41].

And by our above analysis of the quantum sampler:

$$p_{\phi(x)} = \frac{Q(\phi(x))^2 \binom{k}{\psi(\phi(x)_1)} \binom{k}{\psi(\phi(x)_2)} \cdots \binom{k}{\psi(\phi(x)_n)}}{2^{kn} \mathrm{Var}\left[Q(X)\right]}$$

Note that: $\frac{1}{2} \sum_{y \in [-k,+k]^n} |p_y - q_y| \leqslant \beta$, which, in terms of x, because we are summing over all strings in the orbit under $(S_k)^n$ symmetry, can be written:

$$\frac{1}{2} \sum_{x \in \{\pm 1\}^{kn}} \frac{\left|p_{\phi(x)} - q_{\phi(x)}\right|}{\binom{k}{\psi(\phi(x)_1)} \binom{k}{\psi(\phi(x)_2)} \cdots \binom{k}{\psi(\phi(x)_n)}} \leqslant \beta$$

First we define for each x, $\Delta_x = \frac{|p_{\phi(x)} - q_{\phi(x)}|}{\binom{k}{\psi(\phi(x)_1)} \binom{k}{\psi(\phi(x)_2)} \cdots \binom{k}{\psi(\phi(x)_n)}}$ and so $\|\mathcal{D}_{Q,k} - \mathcal{D}'\| = \frac{1}{2} \sum_x \Delta_x$.

Note that:

$$\mathrm{E}_x[\Delta_x] = \frac{\sum_x \Delta_x}{2^{kn}} = \frac{2\beta}{2^{kn}}$$

And applying Markov, $\forall j > 1$,

$$\Pr_x[\Delta_x > \frac{j 2\beta}{2^{kn}}] < \frac{1}{j}$$

Setting $j = \frac{4}{\delta}, \beta = \frac{\epsilon\delta}{16}$, we have,

$$\Pr_x[\Delta_x > \frac{\epsilon}{2} \cdot \frac{1}{2^{kn}}] < \frac{\delta}{4}$$

Then use approximate counting (with an $\mathbf{NP}$ oracle), using Theorem 6 on the randomness of S to obtain an output $\tilde{q}_y$ so that, for all $\gamma > 0$, in time polynomial in n and $\frac{1}{\gamma}$:

$$\Pr[|\tilde{q}_y - q_y| > \gamma \cdot q_y] < \frac{1}{2^n}$$

Because we can amplify the failure probability of Stockmeyer's algorithm to be inverse exponential.

Equivalently in terms of x:

$$\Pr_x\left[\frac{|\tilde{q}_{\phi(x)} - q_{\phi(x)}|}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \gamma \cdot \frac{q_{\phi(x)}}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}}\right] < \frac{1}{2^n}$$

And we have:

$$\mathrm{E}_x\left[\frac{q_{\phi(x)}}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}}\right] \leqslant \frac{\sum_x \frac{q_{\phi(x)}}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}}}{2^{kn}} = \frac{1}{2^{kn}}$$

Thus, by Markov,

$$\Pr_x[\frac{q_{\phi(x)}}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \frac{j}{2^{kn}}] < \frac{1}{j}$$

Now, setting $\gamma = \frac{\epsilon\delta}{8}$ and applying the union bound:

$$\begin{aligned}
&\Pr_x\left[\frac{|\tilde{q}_{\phi(x)} - p_{\phi(x)}|}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \frac{\epsilon}{2^{kn}}\right] \\
&\quad\leqslant \Pr_x\left[\frac{|\tilde{q}_{\phi(x)} - q_{\phi(x)}|}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \frac{\epsilon}{2}\cdot\frac{1}{2^{kn}}\right] + \Pr_x\left[\frac{|q_{\phi(x)} - p_{\phi(x)}|}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \frac{\epsilon}{2}\cdot\frac{1}{2^{kn}}\right] \\
&\quad\leqslant \Pr_x\left[\frac{q_{\phi(x)}}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \frac{j}{2^{kn}}\right] \\
&\quad+ \Pr\left[\frac{|\tilde{q}_{\phi(x)} - q_{\phi(x)}|}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}} > \gamma\cdot\frac{q_{\phi(x)}}{\binom{k}{\psi(\phi(x)_1)}\binom{k}{\psi(\phi(x)_2)}\cdots\binom{k}{\psi(\phi(x)_n)}}\right] + \Pr_x\left[\Delta_x > \frac{\epsilon}{2}\cdot\frac{1}{2^{kn}}\right] \\
&\quad\leqslant \frac{1}{j} + \frac{1}{2^n} + \frac{\delta}{4} \\
&\quad\leqslant \frac{\delta}{2} + \frac{1}{2^n} \leqslant \delta.
\end{aligned}$$

□

10 Putting it All Together

In this section we put our results in perspective and conclude.

As mentioned before, our goal is to find a class of distributions $\{\mathcal{D}_n\}_{n>0}$ that can be sampled exactly in $poly(n)$ time on a Quantum Computer, with the property that there does not exist a (classical) Sampler relative to that class of distributions, $\{\mathcal{D}_n\}_{n>0}$.

Using the results in Sections 5 and 6 we can quantumly sample from a class of distributions $\{\mathcal{D}_{Q,k}\}_{n>0}$, where $k \in poly(n)$ with the property that, if there exists a classical Sampler relative to this class of distributions, there exists an $\epsilon \mathrm{Var}\left[Q(X)\right]$-additive δ-average case solution to the Q^2 function with respect to the $\mathcal{B}(0,k)^n$ distribution. If we had an efficient decomposition for the "Squashed QFT" unitary matrix, we could use the results from Sections 8 and 9 to make k as large as $\exp(n)$. We would like this to be an infeasible proposition, and so we conjecture:

Conjecture 2. *There exists some Efficiently Specifiable polynomial Q with n variables, so that $\epsilon \mathrm{Var}\left[Q(X)\right]$-additive δ-average case solutions with respect to $\mathcal{B}(0,k)^n$, for any fixed $k < exp(n)$, to Q^2, cannot be computed in (classical) randomized $poly(n, 1/\epsilon, 1/\delta)$ time with a $\mathbf{PH}$ oracle.*

At the moment we don't know of such a decomposition for the "Squashed QFT". However, we do know that we can classically evaluate a related fast (time $n \log^2 n$) polynomial transform by a theorem of Driscoll, Healy, and Rockmore [DJR97]. We wonder if there is some way to use intuition gained by the existence of this fast polynomial transform to show the existence of an efficient decomposition for our "Squashed QFT".

Additionally, if we can prove the Anti-Concentration Conjecture (Conjecture 1) relative to some Efficiently Specifiable polynomial Q and the $\mathcal{B}(0,k)^n$ distribution, we appeal to Theorem 11 to show that it suffices to prove:

Conjecture 3. *There exists some Efficiently Specifiable polynomial Q with n variables, so that Q satisfies Conjecture 1 relative to $\mathcal{B}(0,k)^n$, for $k \leqslant \exp(n)$, and ϵ-multiplicative δ-average case solutions, with respect to $\mathcal{B}(0,k)^n$, to Q^2 cannot be computed in (classical) randomized $poly(n, 1/\epsilon, 1/\delta)$ time with a $\mathbf{PH}$ oracle.*

We would be happy to prove that either of these two solutions (additive or multiplicative) are $\#\mathbf{P}$-hard. In this case we can simply invoke Toda's Theorem [Tod91] to show that such a randomized classical solution would collapse $\mathbf{PH}$ to some finite level.

We note that at present, both of these conjectures seem out of reach, because we do not have an example of a polynomial that is $\#\mathbf{P}$-hard to approximate (in either multiplicative or additive) on average, in the sense that we need. Hopefully this is a consequence of a failure of proof techniques, and can be addressed in the future with new ideas.

References

[AA13] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. *Theory of Computing*, 9:143–252, 2013.

[Aar10a] Scott Aaronson. BQP and the polynomial hierarchy. In Leonard J. Schulman, editor, *STOC*, pages 141–150. ACM, 2010.

[Aar10b] Scott Aaronson. A counterexample to the Generalized Linial-Nisan conjecture. *ECCC Report 109*, 2010.

[Aar10c] Scott Aaronson. The equivalence of sampling and searching. *Electronic Colloquium on Computational Complexity (ECCC)*, 17:128, 2010.

[Aar11] Scott Aaronson. A linear-optical proof that the Permanent is #P-hard. *Electronic Colloquium on Computational Complexity (ECCC)*, 18:43, 2011.

[ADH97] Leonard M. Adleman, Jonathan DeMarrais, and Ming-Deh A. Huang. Quantum computability. *SIAM J. Comput.*, 26(5):1524–1540, 1997.

[BBBV97] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh V. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997.

[Ber41] Andrew C Berry. The accuracy of the gaussian approximation to the sum of independent variates. *Transactions of the American Mathematical Society*, 49(1):122–136, 1941.

[BJS10] Michael J. Bremner, Richard Jozsa, and Dan J. Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. 2010.

[BM58] G. E. P. Box and M. E. Muller. A note on the generation of random normal deviates. *Annals of Mathematical Statistics*, 29:610–611, 1958.

[BMS15] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd. Average-case complexity versus approximate simulation of commuting quantum computations. *CoRR*, abs/1504.07999, 2015.

[BV97] Ethan Bernstein and Umesh V. Vazirani. Quantum complexity theory. *SIAM J. Comput.*, 26(5):1411–1473, 1997.

[DJR97] James R. Driscoll, Dennis M. Healy Jr., and Daniel N. Rockmore. Fast discrete polynomial transforms with applications to data analysis for distance transitive graphs. *SIAM J. Comput.*, 26(4):1066–1099, 1997.

[FU11] Bill Fefferman and Chris Umans. On pseudorandom generators and the BQP vs PH problem. *QIP*, 2011.

[Knu73] Donald E. Knuth. *The Art of Computer Programming, Volume III: Sorting and Searching*. Addison-Wesley, 1973.

[KSV02] A.Y Kitaev, A.H Shen, and M.N Vyalyi. *Quantum and Classical Computation*. AMS, 2002.

[KvM02] Adam Klivans and Dieter van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM J. Comput.*, 31(5):1501–1526, 2002.

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quanum Information*. Cambridge U.P., 2000.

[Sho94] Peter W. Shor. Polynominal time algorithms for discrete logarithms and factoring on a quantum computer. In Leonard M. Adleman and Ming-Deh A. Huang, editors, *ANTS*, volume 877 of *Lecture Notes in Computer Science*, page 289. Springer, 1994.

[Sto85] Larry J. Stockmeyer. On approximation algorithms for #P. *SIAM J. Comput.*, 14(4):849–861, 1985.

[TD02] Barbara M. Terhal and David P. DiVincenzo. Adaptive quantum computation, constant depth quantum circuits and Arthur-Merlin games. *CoRR*, quant-ph/0205133, 2002.

[Tod91] Seinosuke Toda. PP is as hard as the Polynomial-Time Hierarchy. *SIAM J. Comput.*, 20(5):865–877, 1991.

[TV08] Terence Tao and Van Vu. On the permanent of random Bernoulli matrices. In *Advances in Mathematics*, page 75, 2008.

A The Power of Exact Quantum Sampling

For the sake of completeness, in this section we prove a folklore result (that is implicit in e.g., [Aar11]) showing that, unless the $\mathbf{PH}$ collapses to a finite level, there is a class of distributions that can be sampled efficiently on a Quantum Computer, that cannot be sampled exactly classically.

Note that as a consequence of Theorem 6, given an efficiently computable $f : \{0,1\}^n \to \{0,1\}$ we can compute a multiplicative approximation to $\Pr_{x \sim U_{\{0,1\}^n}}[f(x) = 1] = \frac{\sum_{x \in \{0,1\}^n} f(x)}{2^n}$ in the $\mathbf{PH}$.

Now we show the promised class of quantumly sampleable distributions:

Definition 25 ($\mathcal{D}_f$)**.** *Given $f : \{0,1\}^n \to \{\pm 1\}$, we define the distribution $\mathcal{D}_f$ over $\{0,1\}^n$ as follows:*

$$\Pr_{\mathcal{D}_{f,n}}[y] = \frac{\left(\sum_{x \in \{0,1\}^n} (-1)^{\langle x,y \rangle} f(x)\right)^2}{2^{2n}}$$

The fact that this is a distribution will follow from the proceeding discussion.

Theorem 26. *For all efficiently computable $f : \{0,1\}^n \to \{\pm 1\}$ we can sample from $\mathcal{D}_f$ in $poly(n)$ time on a Quantum Computer.*

Proof. Consider the following quantum algorithm:

1. Prepare the uniform superposition over n qubits, $\frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} |x\rangle$
2. Since by assumption f is efficiently computable, we can apply f to the phases (as discussed in Section 3), with two quantum queries to f resulting in:

$$|f\rangle = \frac{1}{2^{n/2}} \sum_{x \in \{0,1\}^n} f(x)|x\rangle$$

3. Apply the n qubit Hadamard, $H^{\otimes n}$

4. Measure in the standard basis

Note that $H^{\otimes n}|f\rangle = \frac{1}{2^n} \sum_{y \in \{0,1\}^n} \sum_{x \in \{0,1\}^n} (-1)^{\langle x,y \rangle} f(x)|y\rangle$ and therefore the distribution sampled by the above quantum algorithm is $\mathcal{D}_f$. □

As before, the key observation is that $\left(\langle 00...0|H^{\otimes n}|f\rangle\right)^2 = \frac{\left(\sum_{x \in \{0,1\}^n} f(x)\right)^2}{2^{2n}}$, and therefore encodes a $\#\mathbf{P}$-hard quantity in an exponentially small amplitude. We can exploit this hardness classically if we assume the existence of a classical sampler, which we define to mean an efficient random algorithm whose output is distributed via this distribution.

Theorem 27 (Folklore, e.g., [Aar11])**.** *Suppose we have a classical randomized algorithm B, which given as input 0^n, samples from $\mathcal{D}_f$ in time $poly(n)$, then the* $\mathbf{PH}$ *collapses to* $\mathbf{BPP^{NP}}$.

Proof. The proof follows by applying Theorem 6 to obtain an approximate count to the fraction of random strings r so that $B(0^n, r) = 00..0$. Formally, we can output an α so that:

$$(1-\epsilon)\frac{\left(\sum_{x \in \{0,1\}^n} f(x)\right)^2}{2^{2n}} \leqslant \alpha \leqslant \frac{\left(\sum_{x \in \{0,1\}^n} f(x)\right)^2}{2^{2n}}(1+\epsilon)$$

In time $poly(n, 1/\epsilon)$ using an $\mathbf{NP}$ oracle. Multiplying through by 2^{2n} allows us to get a multiplicative approximation to $\left(\sum_{x \in \{0,1\}^n} f(x)\right)^2$ in the $\mathbf{PH}$. It is clear that, given efficiently computable $f : \{0,1\}^n \to \{\pm 1\}$ computing $\sum_{x \in \{0,1\}^n} f(x)$ is $\#\mathbf{P}$-hard. Aaronson [Aar11] has shown that even calculating this relative error estimate to $\left(\sum_{x \in \{0,1\}^n} f(x)\right)^2$ is $\#\mathbf{P}$-hard. Since we know by Toda's Theorem [Tod91], $\mathbf{PH} \subseteq \mathbf{P}^{\#\mathbf{P}}$, we now have that $\mathbf{P}^{\#\mathbf{P}} \subseteq \mathbf{BPP^{NP}} \Rightarrow \mathbf{PH} \subseteq \mathbf{BPP^{NP}}$ leading to our theorem. Note also that this theorem would hold even under the weaker assumption that the sampler is contained in $\mathbf{BPP^{PH}}$. □

We end this Section by noting that Theorem 27 is extremely sensitive to the exactness condition imposed on the classical sampler, because the amplitude of the quantum state on which we based our hardness is only exponentially small. Thus it is clear that by weakening our sampler to an "approximate" setting in which the sampler is free to sample any distribution Y so that the Total Variation distance $\|Y - \mathcal{D}_f\| \leqslant 1/poly(n)$ we no longer can guarantee any complexity consequence using the above construction. Indeed, this observation makes the construction quite weak– for instance, it may even be unfair to demand that any physical realization of this quantum circuit *itself* samples exactly from this distribution! In the preceding sections we are motivated by this apparent weakness and discuss the intractability of approximately sampling in this manner from quantumly sampleable distributions.

B Computation of the Variance of Efficiently Specifiable Polynomial

In this section we compute the variance of the distribution over $\mathbb{R}$ induced by an Efficiently Specifiable polynomial Q with assignments to the variables chosen independently from the $\mathcal{B}(0,k)$ distribution. We will denote this throughout the section by $\mathrm{Var}\left[Q(X)\right]$. Recall, by the definition of Efficiently Specifiable, we have that Q is an n variate homogenous multilinear polynomial with $\{0,1\}$ coefficients. Assume Q is of degree d and has m monomials. Let each $[-k,k]$ valued variable X_i be independently distributed from $\mathcal{B}(0,k)$.

We adopt the notation whereby, for $j \in [m], l \in [d]$, x_{jl} is the l-th variable in the j-th monomial of Q.

Using the notation we can express $Q(X_1,...,X_n) = \sum_{j=1}^{m} \prod_{l=1}^{d} X_{jl}$. By independence of these random variables and since they are mean 0, it suffices to compute the variance of each monomial and multiply by m:

$$\mathrm{Var}\left[Q(X)\right] = \mathrm{Var}\ \left[Q(X_1,...,X_n)\right] = \mathrm{E}\ \left[\sum_{j=1}^{m}\prod_{l=1}^{d} X_{jl}^2\right] = \sum_{j=1}^{m}\mathrm{E}\ \left[\prod_{l=1}^{d} X_{jl}^2\right] \tag{4}$$

$$= m\mathrm{E}\ \left[\prod_{l=1}^{d} X_{1_l}^2\right] = m\prod_{l=1}^{d}\mathrm{E}\ \left[X_{1_l}^2\right] \tag{5}$$

$$= m\left(\mathrm{E}\ \left[X_{1_1}^2\right]\right)^d \tag{6}$$

Now since these random variables are independent and identically distributed, we can calculate the variance of an arbitrary X_{jl} for any $j \in [m]$ and $l \in [d]$:

$$\mathrm{E}\left[X_{jl}^2\right] = \frac{1}{2^k}\sum_{i=0}^{k}\left[(k-2i)^2\binom{k}{i}\right] \tag{7}$$

$$\tag{8}$$

Thus, the variance of Q is:

$$m\frac{1}{2^{kd}}\left(\sum_{i=0}^{k}\left[(k-2i)^2\binom{k}{i}\right]\right)^d$$

It will be useful to calculate this variance in a different way, and obtain a simple closed form. In this way we will consider the k-valued equivalent polynomial $Q'_k : \mathbb{T}_2^{nk} \to \mathbb{R}$ which is a sum of $m' = mk^d$ multilinear monomials, each of degree d. As before we can write $Q'_k(X_1,...,X_{nk}) = \sum_{j=1}^{m'}\prod_{l=1}^{d} X_{jl}$. Note that the uniform distribution over assignments in $\mathbb{T}_2^{kn}$ to Q'_k induces $\mathcal{B}(0,k)^n$ over $[-k,k]^n$ assignments to Q. By the same argument as above, using symmetry and independence of random variables, we have:

$$\mathrm{Var}\left[Q(X)\right] = \mathrm{Var}\left[Q(X_1, X_2, ..., X_n)\right] = \mathrm{Var}\left[Q'_k(X_1, X_2, ..., X_{nk})\right] \tag{9}$$

$$= m' \prod_{l=1}^{d} \mathrm{E}\left[X_{1_l}^2\right] \tag{10}$$

$$= m' \mathrm{E}\left[X_{1_1}^2\right]^d = 1^d m' = m' = k^d m \tag{11}$$

C Examples of Efficiently Specifiable Polynomials

In this section we give two examples of Efficiently Specifiable polynomials.

Theorem 28. $\mathbf{Permanent}\left(x_1, ..., x_{n^2}\right) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} x_{i,\sigma(i)}$ *is Efficiently Specifiable.*

Proof. We note that it will be convenient in this section to index starting from 0. The theorem follows from the existence of an $h_{\mathbf{Permanent}} : [0, n! - 1] \to \{0,1\}^{n^2}$ that efficiently maps the i-th permutation over n elements to a string representing its obvious encoding as an $n \times n$ permutation matrix. We will prove that such an efficiently computable $h_{\mathbf{Permanent}}$ exists and prove that its inverse, $h^{-1}_{\mathbf{Permanent}}$ is also efficiently computable.

The existence of $h_{\mathbf{Permanent}}$ follows from the so-called "factorial number system" [Knu73], which gives an efficient bijection that associates each number in $[0, n! - 1]$ with a permutation in S_n. It is customary to think of the permutation encoded by the factorial number system as a permuted sequence of n numbers, so that each permutation is encoded in $n \log n$ bits. However, it is clear that we can efficiently transform this notation into the $n \times n$ permutation matrix.

To go from an integer $j \in [0, n! - 1]$ to its permutation we:

1. Take j to its "factorial representation", an n number sequence, where the i-th place value is associated with $(i-1)!$, and the sum of the digits multiplied by the respective place value is the value of the number itself. We achieve this representation by starting from $(n-1)!$, setting the leftmost value of the representation to $j' = \lfloor \frac{j}{(n-1)!} \rfloor$, letting the next value be $\lfloor \frac{j - j' \cdot (n-1)!}{(n-2)!} \rfloor$ and continuing until 0. Clearly this process can be efficiently achieved and efficiently inverted, and observe that the largest each value in the i-th place value can be is i.

2. In each step we maintain a list ℓ which we think of as originally containing n numbers in ascending order from 0 to $n-1$.

3. Repeat this step n times, once for each number in the factorial representation. Going from left to right, start with the left-most number in the representation and output the value in that position in the list, ℓ. Remove that position from ℓ.

4. The resulting n number sequence is the encoding of the permutation, in the standard $n \log n$ bit encoding

□

Now we show that the Hamiltonian Cycle Polynomial is Efficiently Specifiable.

Given a graph G on n vertices, we say a Hamiltonian Cycle is a path in G that starts at a given vertex, visits each vertex in the graph exactly once and returns to the start vertex.

We define an n-cycle to be a Hamiltonian cycle in the complete graph on n vertices. Note that there are exactly $(n-1)!$ n-cycles in S_n.

Theorem 29. $\mathbf{HamiltonianCycle}\,(x_1, ..., x_{n^2}) = \sum_{\sigma:\, n-cycle} \prod_{i=1}^{n} x_{i,\sigma(i)}$ *is Efficiently Specifiable.*

Proof. We can modify the algorithm for the Permanent above to give us an efficiently computable $h_{HC} : [0, (n-1)! - 1] \to \{0,1\}^{n^2}$ with an efficiently computable h_{HC}^{-1}.

To go from a number $j \in [0, (n-1)! - 1]$ to its n-cycle we:

1. Take j to its factorial representation as above. Now this is an $n-1$ number sequence where the i-th place value is associated with $(i-1)!$, and the sum of the digits multiplied by the respective place value is the value of the number itself.
2. In each step we maintain a list ℓ which we think of as originally containing n numbers in ascending order from 0 to $n-1$.
3. Repeat this step $n-1$ times, once for each number in the factorial representation. First remove the smallest element of the list. Then going from left to right, start with the left-most number in the representation and output the value in that position in the list, ℓ. Remove that position from ℓ.
4. We output 0 as the n-th value of our n-cycle.

To take an n-cycle to a factorial representation, we can easily invert the process:

1. In each step we maintain a list ℓ which we think of as originally containing n numbers in order from 0 to $n-1$.
2. Repeat this step $n-1$ times. Remove the smallest element of the list. Going from left to right, start with the left-most number in the n-cycle and output the position of that number in the list ℓ (where we index the list starting with the 0 position). Remove the number at this position from ℓ.

□

D A Simple Example of "Squashed" QFT, for $k = 2$

In this Section we explicitly construct the matrix $L \cdot \tilde{D}_2 \cdot R$ from the QFT over $\mathbb{Z}_2^2$. Note that the matrix we referred to as D_2 is:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

Where we can think of the columns as identified with the monomials $\{1, x_1, x_2, x_1x_2\}$ in this order (from left to right) and the rows (from top to bottom) as identified with the assignments $\{(1,1),(-1,1),(1,-1),(-1,-1)\}$ where the first element in each pair is the assignment to x_1 and the second is to x_2. Note that as desired, the (i,j)-th element of D_2 is the evaluation of the j-th monomial on the i-th assignment.

Now we create $\tilde{D}_2^{(1)}$ by combining columns of monomials that belong to each elementary symmetric polynomial, as described in the prior section. We identify the columns with elementary symmetric polynomials on variables x_1, x_2 in order from left to right: $1, x_1 + x_2, x_1x_2$ and the rows remain the same. This gives us:

$$\begin{pmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & -2 & 1 \end{pmatrix}$$

It can easily be verified that the columns are still orthogonal. Now we note that the rows corresponding to assignments $(1,-1)$ and $(-1,1)$ are in the same orbit with respect to S_2 symmetry. And thus we obtain $\tilde{D}_2$:

$$\begin{pmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & -2 & 1 \end{pmatrix}$$

Now L is the diagonal matrix whose i-th entry is $\sqrt{o_i}$, the size of the i-th equivalence class of assignments under S_2 symmetry. Note that $|o_0| = \sqrt{\binom{2}{0}} = 1$, $|o_1| = \sqrt{\binom{2}{1}} = \sqrt{2}$, and $|o_2| = \sqrt{\binom{2}{2}} = 1$, and so L is:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \sqrt{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

And $L \cdot \tilde{D}_2 =$

$$\begin{pmatrix} 1 & 2 & 1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & -2 & 1 \end{pmatrix}$$

And we note that the columns are now orthogonal. As before, this implies there exists a diagonal matrix R so that $L \cdot \tilde{D}_2 \cdot R$ is unitary. It is easily verified that this is the matrix R:

$$\begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{\sqrt{8}} & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix}$$

And the first two elements r_0, r_1 can be easily seen to be $\frac{1}{\sqrt{2^k}} = \frac{1}{2}$ and $\frac{1}{\sqrt{\sum_{i=0}^{k}\left[\binom{k}{i}(k-2i)^2\right]}} = \frac{1}{\sqrt{8}}$, as claimed in the prior section. Thus the final $k+1 \times k+1$ matrix $L \cdot \tilde{D}_2 \cdot R$ is:

$$\begin{pmatrix} \frac{1}{2} & \frac{2}{\sqrt{8}} & \frac{1}{2} \\ \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ \frac{1}{2} & -\frac{2}{\sqrt{8}} & \frac{1}{2} \end{pmatrix}$$

Which is unitary, as desired.