Classical verification of quantum circuits containing few basis changes

Tommaso F. Demarie, Yingkai Ouyang, and Joseph F. Fitzsimons

Phys. Rev. A **97**, 042319 — Published 11 April 2018

DOI: 10.1103/PhysRevA.97.042319

Classical verification of quantum circuits containing few basis changes

Tommaso F. Demarie,[1,*] Yingkai Ouyang,[1,†] and Joseph F. Fitzsimons[1,2,‡]

[1]*Singapore University of Technology and Design, 8 Somapah Road, Singapore 487372*

[2]*Centre for Quantum Technologies, National University of Singapore, 3 Science Drive 2, Singapore 117543*

We consider the task of verifying the correctness of quantum computation for a restricted class of circuits which contain at most two basis changes. This contains circuits giving rise to the second level of the Fourier Hierarchy, the lowest level for which there is an established quantum advantage. We show that, when the circuit has an outcome with probability at least the inverse of some polynomial in the circuit size, the outcome can be checked in polynomial time with bounded error by a completely classical verifier. This verification procedure is based on random sampling of computational paths and is only possible given knowledge of the likely outcome.

The paradigm of *verification of quantum computation* lies deep into the roots of quantum mechanics, raising questions about the falsifiability of the theory in regimes of high computational complexity [1]. The challenge is to certify the result of a quantum computation using devices that are themselves unable to derive that result. This is not only an issue of theoretical interest. In the last decade, the difficulty of verifying the consistency of an experiment's outcome with regards to the predictions of quantum mechanics has increased dramatically [2]. While simulating the quantum evolution of few qubits on a classical computer is possible, the difficulty of this simulation grows exponentially with the size of the quantum computer. In this sense, the question of verifiability is intimately connected to demonstrating computational advantage of quantum computers over classical machines. For instance, recent claims about the *quantumness* of certain experimental processors [3, 4] have sparked both excited reactions and strong criticisms [5–7]. This situation shows how coming up with a feasible approach for the verification of quantum computation is of practical importance.

These issues have motivated recent theoretical efforts to develop novel protocols for quantum verification, which are comprehensively reviewed in [8]. Generally, these protocols are presented as interactive games where a *verifier* with limited computational resources attempts to verify the output of a quantum computation performed by a *prover* capable of processing quantum information. Such verification protocols rely on different methods: The embedding of veracity tests [9–13] into blind quantum computing protocols [14–16], approaches based on self-testing [17–19], hybrid techniques combining these two procedures [20, 21] and methods based on the use of error correction codes [22–24]. A common thread, however, is the need for at least two parties with quantum capabilities: either a verifier with limited quantum capabilities or multiple quantum provers sharing entanglement. It remains an open question whether decision problems in **BQP** can be efficiently verified by a prover *without any* quantum power [25].

In this work, we explore the possibility of verifying a single quantum processor using purely classical means. In particular, we focus on quantum computations with likely outcomes containing only two layers of gates that do not preserve the computational basis. The choice of two layers is motivated by previous work on a hierarchy of complexity classes known collectively as the *Fourier hierarchy* **FH** [26]. Each level $\mathbf{FH}_k$ of the hierarchy corresponds to the class of problems that can be solved by polynomial quantum circuits, composed of gates that preserve the computational basis and k layers of Fourier transforms on disjoint subset of qubits. Crucially, the second level of the Fourier Hierarchy $\mathbf{FH}_2$ is the smallest level that contains quantum circuits that exhibit advantages over their classical counterparts. In the following, we use $\mathbf{FH}_2$ to denote both decision problems and the class of circuits containing at most two Fourier transforms, with the meaning clear from the context. The likelihood of the outcomes is motivated by considerations of *usefulness*: Quantum algorithms believed to offer an advantage over their classical analogs, such as quantum search [27] algorithms, allow for the efficient extraction of the desired outcome with high probability. Alternatively, models of quantum computation based on sampling are not known to have practical applications[1]. We exploit the structure of such circuits to show that a polynomial-time classical verifier can efficiently verify the outcome of quantum computations, structurally similar to $\mathbf{FH}_2$ circuits, implemented by a prover, with only a single round of communication between them.

We begin with some terminology. If $\mathbf{s} = (s_1, \dots, s_n)$ is an n-bit string, we denote by $|\mathbf{s}\rangle = |s_1\rangle \otimes \cdots \otimes |s_n\rangle$ the corresponding computational basis state. A reversible classical computation C is a bijection from n-bit strings to n-bit strings. We consider the corresponding quantum circuits $\hat{C}$ that are bijections from n-qubit computational basis states to n-qubit computational basis states, and say that such quantum circuits are *classical*. We call $\mathcal{P}_C$ the set of all circuits $\hat{C}$: This is the permutation group on the computational basis, generated by the set of generalised k-Toffoli gates, where k indicates the number of control qubits (i.e. for $k = 0$ we have a Pauli-X, for $k = 1$ a CNOT gate and so on).

When a gate $\hat{G}$ does not preserve the computational basis, there necessarily exists some computational basis elements $|\mathbf{i}\rangle$ and $|\mathbf{j}\rangle$ such that $0 < |\langle \mathbf{i}|\hat{G}|\mathbf{j}\rangle| < 1$. We call such gates *basis-changing* gates. The simplest example of a basis-changing gate is the Hadamard gate $\hat{H}$, which plays the role of a single-

[1] see for example the the discussion on boson sampling in [28]

qubit quantum Fourier transform [29]. In general, the quantum Fourier transform on n qubits can be implemented by an $\mathrm{O}(n \log n \log \log n)$ combination of Hadamard gates and controlled rotations about the Z-axis of the Bloch sphere [30]. Since any quantum circuit can be approximated by a sequence of Toffoli and Hadamard gates, one can think of quantum circuits as procedures that alternate between classical (Toffoli) and quantum (Hadamard) information processing.

The Fourier hierarchy captures part of the subtlety of quantum computation, and its lowest levels correspond to some common complexity classes, informally introduced hereafter [31]. A decision problem deterministically answerable by a classical computer within time polynomial in the input size belongs to the complexity class **P**. The class **NP** corresponds to decision problems for which *yes* instances can be deterministically verified in polynomial time by a classical computer, given a suitable witness string, and so trivially **P** $\subseteq$ **NP**. If a classical computer able to generate randomness can answer in polynomial time a decision problem with error probability bounded by some constant $< \frac{1}{2}$, this decision problem is contained **BPP**. Both **NP** and **BPP** are contained in the class **MA**. A decision problem belongs to **MA** if it has a witness string which can be verified by a polynomial time verifier with bounded probability of error. The class **MA** differs from **NP** because in **MA** the verifier has a bounded non-zero probability to accept a *no instance*. Moving from classical to quantum, **BQP** is the complexity class corresponding to decision problems that can be answered with bounded error probability by a *quantum* computer in polynomial time. If a *yes* instance of a decision problem can be verified with bounded error probability by a quantum polynomial time verifier with the aid of a particular quantum *proof* state, that decision problem belongs to the class **QMA**. The hierarchical relations **P** $\subseteq$ **BPP**$\subseteq$ **BQP** $\subseteq$ **QMA**, and **NP** $\subseteq$ **MA**$\subseteq$ **QMA** hold. Note that the relationship between **NP** and **BQP** is unknown, although it is conjectured that **NP** $\nsubseteq$ **BQP** and that **BQP**$\nsubseteq$ **NP** [32].

Let us allow only uniform families of quantum circuits. Then it is easy to see that $\mathbf{FH}_0 = \mathbf{P}$: Any decision problem represented as a quantum circuit composed solely of classical gates corresponds to a decision problem in **P**. It also follows that $\mathbf{FH}_1 = \mathbf{BPP}$ since, for an input state in the computational basis, a single change of basis cannot cause phase interference. Hence, for a computational basis input, the quantum output of a $\mathbf{FH}_1$ circuit is uniformly distributed on the support of the Fourier transform, and it gives access to randomness elevating **P** to **BPP**. Characterising the levels of the Fourier hierarchy becomes intriguing in terms of complexity for $k \geq 2$. Indeed, Shor's algorithm [33] for factorisation, which gives a substantial speedup when compared to the most efficient known classical algorithm for factorisation, belongs to $\mathbf{FH}_2$. One might then wonder if two layers of quantum Fourier transforms, or basis-changing gates in general, suffice to unlock the power of quantum computation. While to date an exact relationship between $\mathbf{FH}_2$ and the other complexity classes remains unknown, there exist results that assess the classical simulability of a set of $\mathbf{FH}_2$ circuits. In particular, when the final probability distribution of a $\mathbf{FH}_2$ circuit has a support at most polynomial in the input size, this can be efficiently sampled by a classical computer [34]. Note that in our analysis we deal with a larger set of quantum circuits. We highlight this difference later in the discussion.

Our main result deals with the verification of circuits with two layers of basis-changing gates preceded, interspaced, and followed by classical computation from $\mathcal{P}_C$. These circuits are composed of a number of gates polynomial in the input size. Importantly, the basis-changing gates are classically samplable, a property defined rigorously in the next paragraph. Consider a prover performing the circuit just described on a generic input in the computational basis. The prover claims that the classical outcome of the computation, after measuring the resulting quantum state in the computational basis, is the n-bit string $\mathbf{s} = (s_1, \ldots, s_n)$. The verification problem we consider is to decide whether the probability of obtaining **s** is large or alternatively small, *under the promise* that *exactly one* of these two instances holds and that their separation is at least some inverse polynomial in n. We prove that the verification process can be performed by a randomized polynomial time classical verifier with access to the classical description of the input state, the quantum circuit and the string **s**.

We begin by defining the class of basis-changing gates used in the quantum circuits that we consider. We say that an n-qubit unitary operator $\hat{T}$ is a *classically samplable transform* if it satisfies the following set of conditions:

1. $\hat{T}$ can be implemented by a number of Hadamard, CNOT, and $\frac{\pi}{8}$ gates polynomial in the input size n.

2. For all $\mathbf{s}_1 \in \{0,1\}^n$, there exists a polynomial time randomised classical algorithm which randomly samples a distribution over n bit strings such that the probability of outputting $\mathbf{s}_2 \in \{0,1\}^n$ is

$$p_{\mathbf{s}_2}^{\mathbf{s}_1} = \frac{|\langle \mathbf{s}_2 | \hat{T} | \mathbf{s}_1 \rangle|}{\sum_{\mathbf{s} \in \{0,1\}^n} |\langle \mathbf{s} | \hat{T} | \mathbf{s}_1 \rangle|}. \tag{1}$$

3. For every $\mathbf{s}_1$ and $\mathbf{s}_2$, the complex phase of $\langle \mathbf{s}_2 | \hat{T} | \mathbf{s}_1 \rangle$, can be computed classically in polynomial time.

Any tensor product of the identity operator, Hadamard, Fourier, or inverse Fourier transforms on disjoint systems satisfies the above definition. However the full set of operations that satisfy these criteria is larger, and we do not limit the subsequent analysis to the gates listed above. This extends the class of circuits we allow with respect to the classically simulable circuits analysed in [34]. There, in contrast to our case, the second classically samplable transform is either exactly a Fourier or inverse Fourier transform applied to any subset of $k \leq n$ qubits, or alternatively an arbitrary tensor product of n single-qubit unitary operations.

We say that the subset $S_{\hat{T}} \subseteq \{1, ..., n\}$ is the support of $\hat{T}$ if $\hat{T}$ acts non-trivially on the qubits labelled by the elements of $S_{\hat{T}}$. Given an input state $|\mathbf{s}\rangle$ we use $\mathcal{B}(\hat{T}, |\mathbf{s}\rangle)$ to denote the set of all n-bit strings where the i-th component is equal to s_i for all $i \notin S_{\hat{T}}$. For simplicity, in our analysis we restrict our attention to classically samplable transforms for which $p^{\mathbf{s}_1}_{\mathbf{s}_2} = 2^{-m}$, where m is the cardinality of $S_{\hat{T}}$. We thereby define a *k-transform circuit*, which is a quantum circuit $\mathcal{C}$ that has the following properties.

1. The input to $\mathcal{C}$ is a computational basis state.
2. The quantum circuit $\mathcal{C}$ comprises of a polynomial number of Toffoli gates (basis preserving) and k classically samplable transforms (basis changing), followed by measurement of all qubits in the computational basis.
3. The output of $\mathcal{C}$ is the bit string that corresponds to the measured computational basis state.

Having defined the circuits under examination, we cast the corresponding verification task as a decision problem with the promise that the input satisfies the requirements for either a *yes* instance or a *no* instance as we now describe. A k-transform circuit is δ-*deterministic* with output $\mathbf{s}$ if the measurement outcome after running the circuit is $\mathbf{s}$ with probability at least δ. In the k-transform verification problem, an instance consists of a k-transform circuit $\mathcal{C}$ and a string $\mathbf{s}$, with the promise that exactly one of the following instances is true.

1. The *yes* instance: $\mathcal{C}$ is δ-deterministic with output $\mathbf{s}$.
2. The *no* instance: $\mathcal{C}$ is not ϵ-deterministic for any output.

The task is to decide if either the *yes* instance or the *no* instance holds for the circuit $\mathcal{C}$, where the thresholds δ and ϵ are positive real numbers in the interval $[0,1]$ such that $\epsilon < \delta/2$, and $\gamma = \sqrt{\frac{\delta}{2}} - \sqrt{\epsilon}$ satisfies $\gamma = \Omega(\text{poly}^{-1}(n))$. This last constraint ensures that the probabilities are sufficiently distinct so that the difference can be resolved with a polynomial number of samples.

Our main result is that the k-transform verification promise problem is in **BPP** for $k \leq 2$. It suffices to show that if $\mathcal{C}$ is δ-deterministic then there exists a proof of this fact that can be verified by a classical prover in polynomial time with bounded error of $\frac{1}{3}$, and that this verification procedure rejects any proof with bounded error of $\frac{1}{3}$ if $\mathcal{C}$ is not ϵ-deterministic. When $k = 0$, the circuit is completely classical, and hence it can be verified by direct evaluation. When $k = 1$, let us call each layer of classical computation $\hat{C}_i$, where the index i indicates the temporal order of the layer in the circuit. Then the output state of $\mathcal{C}$ before the final measurement is $\hat{C}_2\hat{T}_1\hat{C}_1|\mathbf{s}_{\text{in}}\rangle$ with an n-qubit computational basis input state $|\mathbf{s}_{\text{in}}\rangle$. Here $\hat{C}_1$ and $\hat{C}_2$ are polynomial-sized Toffoli circuits in $\mathcal{P}_C$, and $\hat{T}_1$ is a classically samplable transform. Note that $C_1(\mathbf{s}_{\text{in}}) = \mathbf{r}$ for some n-bit sting $\mathbf{r}$ and hence $\hat{C}_1|\mathbf{s}_{\text{in}}\rangle = |C_1(\mathbf{s}_{\text{in}})\rangle = |\mathbf{r}\rangle$. Because of the reversible classical property of $\hat{C}_2$, the verifier can efficiently derive $|C_2^{-1}(\mathbf{s})\rangle$, where $\hat{C}_2|C_2^{-1}(\mathbf{s})\rangle = |\mathbf{s}\rangle$. Finally the complex phase $\langle C_2^{-1}(\mathbf{s})|\hat{T}_1|\mathbf{r}\rangle$ can be trivially computed by definition. This answers the verification problem for $k = 1$.

We now evaluate the probability that a fixed output string $\mathbf{s}$ is obtained from any 2-transform circuit evaluated on the n-qubit computational basis state $|\mathbf{s}_{\text{in}}\rangle$. The output of a 2-transform circuit $\mathcal{C}$ before the measurement can be written as $\hat{C}_3\hat{T}_2\hat{C}_2\hat{T}_1\hat{C}_1|\mathbf{s}_{\text{in}}\rangle$ where the transforms $\hat{T}_1, \hat{T}_2$ act non-trivially on $a \leq n$ and $b \leq n$ qubits respectively. Then

$$\hat{T}_1|\mathbf{r}\rangle = 2^{-\frac{a}{2}} \sum_{\mathbf{j}\in\mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)} e^{i\alpha_{\mathbf{r},\mathbf{j}}}|\mathbf{j}\rangle\,, \tag{2}$$

where $\alpha_{\mathbf{r},\mathbf{j}}$ is the phase for the complex amplitude of the state $|\mathbf{j}\rangle$ produced by the samplable transform given the fixed input $|\mathbf{r}\rangle$. Then

$$\hat{C}_2\hat{T}_1|\mathbf{r}\rangle = 2^{-\frac{a}{2}} \sum_{\mathbf{j}\in\mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)} e^{i\alpha_{\mathbf{r},\mathbf{j}}}|C_2(\mathbf{j})\rangle\,, \tag{3}$$

and

$$\hat{T}_2\hat{C}_2\hat{T}_1|\mathbf{r}\rangle = 2^{-\frac{a+b}{2}} \sum_{\substack{\mathbf{j}\in\mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)\\ \mathbf{k}\in\mathcal{B}(\hat{T}_2,|C_2(\mathbf{j})\rangle)}} e^{i\alpha_{\mathbf{r},\mathbf{j}}}e^{i\beta_{C_2(\mathbf{j}),\mathbf{k}}}|\mathbf{k}\rangle\,, \tag{4}$$

where each $\beta_{C_2(\mathbf{j}),\mathbf{k}}$ is the phase associated to the complex amplitude of each state $|\mathbf{k}\rangle$ induced by the action of $\hat{T}_2$ on the state $|C_2(\mathbf{j})\rangle$. The combined action $\hat{T}_2\hat{C}_2\hat{T}_1$, equivalent to the core of Shor's algorithm, is unlikely to be simulated efficiently by a classical circuit because the gate $\hat{C}_2$ is performed on a superposition of computational basis vectors [35]. Indeed, such circuits allow for the preparation and measurement in the XY-plane and Z-basis of arbitrary graph states, and hence can be used to implement uncorrected measurement-based computation [36]. Under post-selection this becomes universal, and hence by standard arguments [37–39] sampling the output of 2-transform circuits within bounded multiplicative error is computationally hard classically. However, with knowledge of $\mathbf{s}$, Born's rule $P_{\mathbf{s}} = |\langle C_3^{-1}(\mathbf{s})|\hat{T}_2\hat{C}_2\hat{T}_1|\mathbf{r}\rangle|^2$ gives the probability of obtaining the output $\mathbf{s}$, which can be estimated using a sampling technique as follows.

A randomised classical sampling algorithm that runs in a time polynomial in n is used to answer the verification problem for any 2-transform circuit on n qubits. To show this, we start with the amplitude $\xi_{\mathbf{s}} = \langle C_3^{-1}(\mathbf{s})|\hat{T}_2\hat{C}_2\hat{T}_1|\mathbf{r}\rangle$ associated to the state $|\mathbf{s}\rangle$. One needs to distinguish between the $b \geq a$ and $a > b$ cases. We consider only the former case, since the same analysis can be performed for the latter case by first taking the complex conjugate of the amplitude $\xi_{\mathbf{s}}$ and expanding over paths through $\hat{T}_2$ rather than $\hat{T}_1$, as is done next. We expand the amplitude as

$$\begin{aligned}\xi_{\mathbf{s}} &= 2^{-\frac{a}{2}} \sum_{\mathbf{j}\in\mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)} e^{i\alpha_{\mathbf{r},\mathbf{j}}}\langle C_3^{-1}(\mathbf{s})|\hat{T}_2|C_2(\mathbf{j})\rangle \\ &= 2^{-\frac{a+b}{2}} \sum_{\mathbf{j}\in\mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)} \theta_{C_2(\mathbf{j}),C_3^{-1}(\mathbf{s})}e^{i\alpha_{\mathbf{r},\mathbf{j}}+i\beta_{C_2(\mathbf{j}),C_3^{-1}(\mathbf{s})}}\,,\end{aligned}$$

where $\theta_{C_2(\mathbf{j}),C_3^{-1}(\mathbf{s})} \in \{0,1\}$ depending on whether $\langle C_3^{-1}(\mathbf{s})|\hat{T}_2|C_2(\mathbf{j})\rangle$ is non-zero. To simplify notation, we define

$$u_{\mathbf{j}} = 2^{-a}\mathrm{Re}\left(\theta_{C_2(\mathbf{j}),C_3^{-1}(\mathbf{s})} e^{i\alpha_{\mathbf{r},\mathbf{j}}+i\beta_{C_2(\mathbf{j}),C_3^{-1}(\mathbf{s})}}\right)\text{ , and}$$
$$v_{\mathbf{j}} = 2^{-a}\mathrm{Im}\left(\theta_{C_2(\mathbf{j}),C_3^{-1}(\mathbf{s})} e^{i\alpha_{\mathbf{r},\mathbf{j}}+i\beta_{C_2(\mathbf{j}),C_3^{-1}(\mathbf{s})}}\right)\text{ ,}$$

so that $\xi_{\mathbf{s}} = 2^{-\frac{(b-a)}{2}}\left(\sum_{\mathbf{j}} u_{\mathbf{j}} + iv_{\mathbf{j}}\right)$. The triangle inequality implies that $2^{-\frac{b-a}{2}} \geq |\xi_{\mathbf{s}}|$. Hence all the cases where $b - a = \Omega(\mathrm{poly}(n))$ are trivial to analyse, since they cannot be $\mathrm{poly}^{-1}(n)$-deterministic for any $\mathbf{s}$. In the following we use the rescaled values $\delta' = 2^{b-a}\delta$ and $\epsilon' = 2^{b-a}\epsilon$ such that $\gamma' = \sqrt{\frac{\delta'}{2}} - \sqrt{\epsilon'}$. Let $A = 2^{-a}\sum_{\mathbf{j}\in\mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)} u_{\mathbf{j}}$ and $B = 2^{-a}\sum_{\mathbf{j}\in\mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)} v_{\mathbf{j}}$. It follows that when $|\xi_{\mathbf{s}}|^2 \geq \delta$ we have $|A+iB| \geq \sqrt{\delta'}$, then either $|A| \geq \sqrt{\frac{\delta'}{2}}$ or $|B| \geq \sqrt{\frac{\delta'}{2}}$ is true. When $|\xi_{\mathbf{s}}|^2 \leq \epsilon$, from the triangle inequality, the inequality $|A+iB| \leq \sqrt{\epsilon'}$ implies that both $|A| \leq \sqrt{\epsilon'}$ and $|B| \leq \sqrt{\epsilon'}$ are true.

Using the variables $u_{\mathbf{j}}$ and $v_{\mathbf{j}}$ we define the independently and identically distributed random variables $\hat{X}_i$ for $i = 1,\ldots,N$ where N is polynomial in n and $\Pr(\hat{X} = u_{\mathbf{j}} + iv_{\mathbf{j}}) = 2^{-a}$ for all $\mathbf{j} \in \mathcal{B}(\hat{T}_1,|\mathbf{r}\rangle)$. The definition of $\hat{T}$ ensures that there exists a polynomial time randomised classical algorithm for sampling the set $\{\hat{X}_i\}_{i=1}^N$. Let $\hat{A}$ and $\hat{B}$ be the real and imaginary parts of $\frac{1}{N}\sum_i \hat{X}_i$ respectively. Let $\theta = \sqrt{\epsilon'} + \gamma'/2$. Without loss of generality assume that at the end of the sampling $|\hat{A}| \geq |\hat{B}|$. If this is the case, when $|\hat{A}| < \theta$, the verifier concludes that $|A+iB| \leq \sqrt{\epsilon'}$, and if $|\hat{A}| \geq \theta$, the verifier concludes that $|A+iB| \geq \sqrt{\delta'}$ since the promise of the problem excludes the possibility that $\sqrt{\frac{\delta'}{2}} \leq |A+iB| < \sqrt{\delta'}$. If $|\hat{A}| \leq |\hat{B}|$ the same conclusions apply when substituting $|\hat{A}|$ with $|\hat{B}|$. In the following paragraphs we prove that the conclusion of the verifier is incorrect with probability exponentially small in N.

Here we utilise the Hoeffding bound [40] and the reverse triangle inequality applied to probabilities. Hoeffding's bound states that $\Pr\left[|\hat{A} - A| \geq \frac{\gamma'}{2}\right] \leq 2e^{-\gamma'^2 N/8}$. The reverse triangle inequality implies that $|\hat{A} - A| \geq ||\hat{A}| - |A||$, and hence

$$\Pr\left[||\hat{A}| - |A|| \geq \frac{\gamma'}{2}\right] \leq \Pr\left[|\hat{A} - A| \geq \frac{\gamma'}{2}\right]. \qquad (5)$$

Note that when $|A| \geq \sqrt{\delta'/2}$,

$$\Pr\left[|\hat{A}| \leq \theta\right] \leq \Pr\left[|A| - |\hat{A}| \geq \frac{\gamma'}{2}\right]. \qquad (6)$$

Combining the inequalities in Eq. 5 and Eq. 6 with the Hoeffding bound results in $\Pr[|\hat{A}| \leq \theta] \leq 2e^{-\gamma'^2 N/8}$. When $|A| \leq \sqrt{\epsilon'}$,

$$\Pr\left[|\hat{A}| \geq \theta\right] \leq \Pr\left[|\hat{A}| - |A| \geq \frac{\gamma'}{2}\right]. \qquad (7)$$

By similar reasoning to the previous case, this yields $\Pr[|\hat{A}| \geq \theta] \leq 2e^{-\gamma'^2 N/8}$.

We have hence shown that a randomised classical algorithm can distinguish between the *yes* and the *no* instance with probability at least $1 - 2e^{-\gamma'^2 N/8}$. This classical test assesses if the string $\mathbf{s}$ is a likely outcome of the quantum computation and gives a protocol for the classical verification of a 2-transform circuit $\mathcal{C}$:

1. The prover performs $\mathcal{C}$. It generates a classical output string $\mathbf{s}$ and sends it to the verifier.

2. The verifier uses the string $\mathbf{s}$ to identify the amplitude $\langle C_3^{-1}(\mathbf{s})|\hat{T}_2\hat{C}_2\hat{T}_1|\mathbf{r}\rangle$. It then classically samples N complex phases $\{\hat{X}_j\}$, with $\hat{X}_j = \hat{A}_j + i\hat{B}_j$.

3. If $|\hat{A}| > \theta$ and $|\hat{B}| > \theta$ the verifier accepts the result $\mathbf{s}$, and it rejects otherwise.

If the circuit $\mathcal{C}$ is δ-deterministic with outcome $\mathbf{s}$, the verifier accepts with probability at least p if $N > 8\gamma^{-2}\log\frac{2}{1-p}$, and rejects with at least the same probability otherwise. The most general case of non-uniformly distributed amplitudes can be derived by using results from past work on classical simulability of quantum circuits by Van den Nest [41]. There, it is proved that there exists an efficient classical algorithm to approximate the element $\langle C_3^{-1}(\mathbf{s})|\hat{T}_2\hat{C}_2\hat{T}_1|\mathbf{r}\rangle$ with polynomial accuracy, without imposing any additional assumption. This can be used to extend our approach to a most general setting.

The fact that the k-transform verification problem is in **BPP** for $k \leq 2$ bears relevant consequences. We can modify the question by asking whether there exists any $\mathbf{s}'$ for which $\mathcal{C}$ is δ-deterministic, given the promise as before that either such an $\mathbf{s}'$ exists, or the circuit is not ϵ-deterministic for any output. Since $\mathbf{s}$ acts as a witness for this, using the previous algorithm, it follows that this problem is in **MA** for $k \leq 2$. Furthermore, this witness can be efficiently found by sampling $\mathcal{C}$ with high probability, which can be accomplished by a prover limited to efficient quantum computation.

We conclude with a remark. The results from [34] state that circuits in $\mathbf{FH}_2$ with sufficiently sparse output distribution can be simulated efficiently by a classical computer. It remains an open problem to prove whether the circuits we consider in this work can be simulated classically. These circuits can be used to define a novel hierarchy of circuits with respect to the number of classically samplable transforms, analogous to the Fourier hierarchy, such that each $\mathbf{FH}_k$ is necessarily contained within the k-th level of this hierarchy. Studying the complexity of classically simulating this hierarchy of circuits promises then to improve our understanding on the relationship between the structure of quantum computations and quantum supremacy.

ACKNOWLEDGEMENTS

The authors acknowledge support from Singapore's National Research Foundation and Ministry of Education. TFD thanks Atul Mantri and Michal Hajdušek for interesting and stimulating discussions. The authors thank Marteen Van den Nest, Martin Schwarz, and Ashley Montanaro for their precious comments and criticisms. JFF acknowledges support from the Air Force Office of Scientific Research under AOARD grant FA2386-15-1-4082. This material is based on research funded in part by the Singapore National Research Foundation under NRF Award NRF-NRFF2013-01.

* Electronic address: `tommaso_demarie@sutd.edu.sg`
† Electronic address: `yingkai_ouyang@sutd.edu.sg`
‡ Electronic address: `joseph_fitzsimons@sutd.edu.sg`

[1] D. Aharonov and U. Vazirani, in *Computability:Turing, Gödel, Church, and Beyond*, B. J. Copeland, C. J. Posy, and O. Shagrir, eds., (MIT Press, 2013), pp. 376–.
[2] P. Hauke, F. M. Cucchietti, L. Tagliacozzo, I. Deutsch, and M. Lewenstein, Reports on Progress in Physics **75**, 082401 (2012).
[3] S. Boixo, T. Albash, F. Spedalieri, N. Chancellor, and D. Lidar, Nat. Comm. 4 (2013).
[4] S. Boixo *et al.*, Nature Phys. **10**, 218 (2014).
[5] J. Smolin and G. Smith, arXiv:1305.4904 (2013).
[6] S. Shin, G. Smith, J. Smolin, and U. Vazirani, arXiv:1404.6499 (2014).
[7] S. Shin, G. Smith, J. Smolin, and U. Vazirani, arXiv:1401.7087 (2014).
[8] A. Gheorghiu, T. Kapourniotis, and E. Kashefi, arXiv:1709.06984 (2017).
[9] J. Fitzsimons and E. Kashefi, arXiv:1203.5217 (2012).
[10] S. Barz, J. Fitzsimons, E. Kashefi, and P. Walther, Nat. Phys. **9**, 727 (2013).
[11] T. Morimae, Phys. Rev. A **89**, 060302(R) (2014).
[12] M. Hayashi and T. Morimae, arXiv:1505.07535 (2015).
[13] A. Broadbent, arXiv:1509.09180v1 (2015).
[14] A. Broadbent, J. Fitzsimons, and E. Kashefi, in *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, (2009).
[15] S. Barz *et al.*, Science **335**, 303 (2012).
[16] T. Morimae and K. Fujii, Phys. Rev. A 87 (2013).
[17] M. McKague, arXiv:1309.5675 [quant-ph] (2013).
[18] B. Reichardt, F. Unger, and U. Vazirani, Nature **496**, 7446 (2013).
[19] B. Reichardt, F. Unger, and U. Vazirani, in *Proceedings of the 4th conference on Innovation in Theoretical Computer Science*, pp. 321–322 (ACM, 2013).
[20] A. Gheorghiu, E. Kashefi, and P. Wallden, arXiv:1502.02571 (2015).
[21] M. Hajdušek and C. Perez-Delgado and J. Fitzsimons, arXiv:1502.02563v1 (2015).
[22] D. Aharonov, M. Ben-Or, and E. Eban, in *Proceeding of Innovations in Computer Science 2010 (ICS 2010)*, pp. 453–469 (2010).
[23] J. Fitzsimons and M. Hajdušek, arXiv:1512.04375 (2015).
[24] T. Morimae and J. Fitzsimons, arXiv:1603.06046 (2016).
[25] A. Mantri, T. Demarie, N. Menicucci, and J. Fitzsimons, Phys. Rev. X **7**, 031004 (2017).
[26] Y. Shi, Theor. Comput. Sci. 344 (2005).
[27] L. K. Grover, in *28th ACM Symposium on Theory of Computation*, p. 212 (Association for Computing Machinery, New York, 1996).
[28] B. Gard, K. Motes, J. Olson, P. Rohde, and J. Dowling, in *From Atomic to Mesoscale: The Role of Quantum Coherence in Systems of Various Complexities: An introduction to boson sampling* (World Scientific Publishing, 2015), Chap. 8.
[29] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, 2000).
[30] R. Cleve and J. Watrous, in *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pp. 526–536 (2000).
[31] J. Watrous, arXiv:0804.3401 (2008).
[32] S. Aaronson, in *Proceedings of the forty-second ACM symposium on Theory of computing*, pp. 141–150 (2010).
[33] P. W. Shor, in *Proceedings, 35th Annual Symposium on Fundamentals of Computer Science*, pp. 124–134 (IEEE Press, Los Alamitos, 1994).
[34] M. Schwarz and M. V. den Nest, arXiv:1310.6749 (2013).
[35] D. Aharonov, Z. Landau, and J. Makowsky, arXiv:0611156 [quant-ph] (2007).
[36] R. Raussendorf, D. E. Browne, and H. J. Briegel, Phys. Rev. A **68**, 022312 (2003).
[37] M. J. Bremner, R. Jozsa, and D. J. Shepherd, in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, p. rspa20100301 (2010).
[38] T. Morimae, K. Fujii, and J. F. Fitzsimons, Physical review letters **112**, 130502 (2014).
[39] P. P. Rohde *et al.*, Physical Review A **91**, 012342 (2015).
[40] W. Hoeffding, Journal of the American statistical association 58 (1963).
[41] M. V. den Nest, Quant. Inf. Comp. **11**, 784 (2011).