Faster Phase Estimation

Krysta M. Svore,[1, *] Matthew B. Hastings,[2, †] and Michael Freedman[2, ‡]

[1]*Quantum Architectures and Computation Group*
Microsoft Research, Redmond, WA 98052 USA
[2]*Station Q, Microsoft Research, Santa Barbara, CA 93106 USA*
(Dated: April 2, 2013)

We develop several algorithms for performing quantum phase estimation based on basic measurements and classical post-processing. We present a pedagogical review of quantum phase estimation and simulate the algorithm to numerically determine its scaling in circuit depth and width. We show that the use of purely random measurements requires a number of measurements that is optimal up to constant factors, albeit at the cost of exponential classical post-processing; the method can also be used to improve classical signal processing. We then develop a quantum algorithm for phase estimation that yields an asymptotic improvement in runtime, coming within a factor of log* of the minimum number of measurements required while still requiring only minimal classical post-processing. The corresponding quantum circuit requires asymptotically lower depth and width (number of qubits) than quantum phase estimation.

I. INTRODUCTION

Quantum algorithms promise computational speed-ups over their classical counterparts. Quantum phase estimation is a key technique used in quantum algorithms, including algorithms for quantum chemistry [1, 2] and quantum field theory [3], Shor's algorithm for prime factorization [4], and algorithms for quantum sampling [5, 6]. It can be used to find eigenvalues of a unitary matrix efficiently.

There are two main approaches to quantum phase estimation: (1) invoking an inverse Quantum Fourier Transform (QFT) [7–9] to extract information about the phase or (2) performing a basic measurement operation followed by classical post-processing in place of the QFT [10, 11]. An advantage to approach (2) is that it uses classical post-processing in place of quantum operations, trading off an expensive resource for an inexpensive classical computation. In particular, the QFT requires many small controlled-rotations, each of which must be approximated to precision ϵ by a sequence of basic quantum operations of length $O(\log(1/\epsilon))$ [12]. In practice, we may want to significantly reduce the circuit depth of the phase estimation algorithm in exchange for a small increase in circuit width, i.e., the number of qubits. Therefore, we focus on approach (2) and rely primarily on quantum measurements to infer information about the phase.

We begin by outlining the goal of quantum phase estimation and explaining the basic measurement operation that is used as a subroutine to do this, and contrast this problem with the classical Fourier transform. We then describe various phase estimation algorithms; these algorithms all call the same basic measurement operation, but use different parameters to do this.

We first present in Section III a technique based on random measurements to infer the phase; this technique uses the fewest number of measurements of any we know (and we prove that it is within a constant factor of optimal), but it requires impractical classical post-processing for use in, say, Shor's algorithm [4], with a complexity that is exponential in the number of bits being inferred. However, this technique may be practical in certain classical noisy signal processing and inference applications, where the number of bits being inferred is smaller. We explain these applications in this section and give some extensions of the technique that may be useful in inferring very noisy, sparse signals.

In Section IV, we review a quantum phase estimation algorithm based on the same measurement operation, but the measurements are not random and the classical post-processing can be done efficiently [10, 11]. We simulate this algorithm and determine its complexity, circuit depth, and circuit width for various sizes of input.

In Section V, we improve upon this phase estimation algorithm by considering inference across multiple qubits. We show that this technique requires asymptotically fewer measurements, and in turn has a correspondingly (asymptotically) smaller circuit width and depth, while still allowing efficient classical post-processing.

We compare the circuit constructions for Kitaev's phase estimation algorithm and the fast phase estimation algorithm in Section VI. Three models of computation are discussed: the first is a sequential model with limited parallelism, the second is a highly parallel model, and the third is a model based on a cluster of quantum computers.

II. PHASE ESTIMATION AND THE BASIC MEASUREMENT OPERATION

We begin by reviewing the goal of quantum phase estimation and the basic measurement operator, following the algorithm of Kitaev [10] (see Ref. 11 for complete details). We derive the steps slightly differently, in anticipation of our extension in the later sections.

Assume that we have a unitary operator U and we would like to estimate the eigenvalues λ_k of U given U and the eigenvectors $|\xi_k\rangle$:

$$U|\xi_k\rangle = \lambda_k|\xi_k\rangle, \tag{1}$$

where the eigenvalues take the form $\lambda_k = e^{2\pi i \cdot \varphi_k}$. The phase φ_k is a real number modulo 1, which can be represented as a unit-length circle: $\varphi_k = \frac{k}{t} \mod 1$, $\varphi_k \in \mathbb{R}/\mathbb{Z}$, $0 \le k < t < 2^m$ (while it may seem more natural at first to instead consider numbers that range between 0 and 2π, rather than choosing a number between 0 and 1 and multiplying by 2π as we do here, we choose the latter because it will be more natural when later considering an expansion of φ_k as a binary fraction). By measuring the eigenvalues of U, we can obtain an estimate of the phase φ_k; this process is called *quantum phase estimation.* [1]

The goal of all phase estimation algorithms is to take a state of the form $|\xi_k\rangle$ and determine the corresponding eigenvalue λ_k. The measurement operation described below commutes with U, so we can apply it multiple times to the same state with different parameters to improve our knowledge of the eigenvalue. There are two parameters in the measurement result: (1) the *precision δ* and (2) the *probability of error ϵ*. That is, we obtain some estimate α of φ_k where, with probability at least $1-\epsilon$, $|\alpha - \varphi_k|_{\mod 1} < \delta$, where $\mod 1$ is the distance on the unit circle. If φ_k is chosen from a discrete set of angles $\frac{k}{t}$ with fixed t and unknown k, our goal is to make the precision smaller than $\delta = \frac{1}{2t}$ so that we can determine φ_k exactly from this set. In Section III, we slightly simplify the problem by directly inferring the angle φ_k from the discrete set of possible angles, without bothering to introduce a real number precision; in this case ϵ is the probability of error in our discrete inference.

A. Basic Measurement Operation

We begin by constructing a measurement operator such that the conditional probability depends on φ_k, that is, upon measuring this operator, we learn some information about φ_k. This construction relies on the fact that if
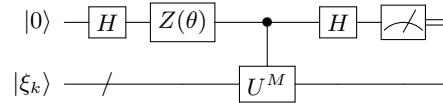
---

[1] In the context of Shor's algorithm, the corresponding eigenvector is defined as $|\xi_k\rangle = \frac{1}{\sqrt{t}} \sum_{n=0}^{t-1} e^{-2\pi i \cdot n \varphi_k} |a^n\rangle$.

FIG. 1: Circuit to perform the measurement operator.

$|\xi_k\rangle$ is an eigenvector of U, then it is also an eigenvector of powers M of U:

$$\begin{aligned} U^M |\xi_k\rangle &= \lambda_k^M |\xi_k\rangle \\ &= e^{2\pi i M \cdot \varphi_k} |\xi_k\rangle. \end{aligned} \tag{2}$$

The operator takes as input two quantum registers: one initialized to $|0\rangle$ and the other initialized to the eigenvector $|\xi_k\rangle$. The operator depends upon two parameters, a "multiple" M and an "angle" θ, where M is an integer between 1 and $t-1$ (to make it practical to implement, we restrict to positive integers M) and θ is a real number between 0 and 2π.

The measurement operator used to measure the eigenvalues is as follows:

$$\begin{aligned} &\Xi_{M,\theta}(U) \tag{3} \\ &= \sum_k \frac{1}{2} \begin{bmatrix} 1 + e^{2\pi i M \cdot \varphi_k + i\theta} & 1 - e^{2\pi i M \cdot \varphi_k + i\theta} \\ 1 - e^{2\pi i M \cdot \varphi_k + i\theta} & 1 + e^{2\pi i M \cdot \varphi_k + i\theta} \end{bmatrix} \otimes |\xi_k\rangle\langle\xi_k| \\ &= \frac{1}{2} \begin{bmatrix} 1 + U^M \exp(i\theta) & 1 - U^M \exp(i\theta) \\ 1 - U^M \exp(i\theta) & 1 + U^M \exp(i\theta) \end{bmatrix}, \end{aligned}$$

which acts on the quantum states by the following transformation:

$$\begin{aligned} &|0\rangle \otimes |\xi_k\rangle \overset{\Xi_{M,\theta}(U)}{\mapsto} \tag{4} \\ &\left(\frac{1 + e^{2\pi i M \cdot \varphi_k + i\theta}}{2} |0\rangle + \frac{1 - e^{2\pi i M \cdot \varphi_k + i\theta}}{2} |1\rangle \right) \otimes |\xi_k\rangle. \end{aligned}$$

The corresponding circuit is shown in Fig. 1. The gate $Z(\theta)$ corresponds to the unitary:

$$Z(\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}. \tag{5}$$

It follows that the measurement outcome probabilities are given by:

$$P_{M,\theta}(0|k) = \left| \frac{1 + e^{2\pi i M \cdot \varphi_k + i\theta}}{2} \right|^2 = \frac{1 + \cos(2\pi M \cdot \varphi_k + \theta)}{2}, \tag{6}$$

and

$$P_{M,\theta}(1|k) = \left| \frac{1 - e^{2\pi i M \cdot \varphi_k + i\theta}}{2} \right|^2 = \frac{1 - \cos(2\pi M \cdot \varphi_k + \theta)}{2}. \tag{7}$$

We write these probabilities as conditional probabilities to emphasize that they depend upon the unknown k.

B. Relation To Classical Fourier Transform and Generalizations

With Eqs. (6,7) in hand, we can see that if we apply a large number of measurements using the same M at both $\theta = 0$ and $\theta = \pi/2$, we can accurately estimate $\cos(2\pi M \cdot \varphi_k)$ and $\sin(2\pi M \cdot \varphi_k)$. Using a sufficiently accurate estimate of these cosines and sines at two different values of M allows us to determine φ_k accurately. This is the problem of reconstructing a sparse signal (in this case, composed of a single Fourier mode) from its value at a small number of different "times" (i.e., different values of M). However, the accurate determination of $\cos(2\pi M \cdot \varphi_k)$ would require a very large number of measurements, polynomial in t, while other methods require many fewer measurements. The reason is that the large number of measurements at a fixed value of M means that each measurement imparts little additional information. By varying M, we are able to obtain accurate results from a much smaller number of measurements.

This relates to a problem of reconstructing the Fourier transform of a signal from very noisy measurements. The quantum phase estimation problem involves a signal with a single Fourier mode. However, this gives rise to a natural generalization of reconstructing a problem with a small number of Fourier modes from very noisy measurements. We consider this problem at the end of the next section.

III. "INFORMATION THEORY" PHASE ESTIMATION

One procedure for estimating the phase (or angle) is to perform a series of random measurements and then solve a hard classical reconstruction problem. We measure the operator at a set of randomly chosen multiples M_i and angles θ_i and classically reconstruct the angle $2\pi\varphi_k$. In this section, we show that we can determine φ_k with only $O(\log(t))$ measurements; we also show that this result is tight.

We randomly select M_i for each measurement i between 1 and $t-1$, and also assume a small randomized offset noise $\theta_i = 2\pi r$, where r is a random double. The conditional measurement probabilities for this measurement operator on the ith measurement are given by:

$$P_i(0|k) = \frac{1 + \cos(2\pi M_i \cdot \varphi_k + \theta_i)}{2}, \qquad (8)$$

and

$$P_i(1|k) = \frac{1 - \cos(2\pi M_i \cdot \varphi_k + \theta_i)}{2} = 1 - P_i(0|k). \quad (9)$$

Let v_i be the outcome of the ith measurement. Since different measurements are independent events, the prob-

ability of getting a given sequence of measurement outcomes is

$$P(v_1, \ldots, v_s|k) = \prod_{i=1}^{s} P_i(v_i|k). \qquad (10)$$

Assuming a flat a priori distribution of k, the probability distribution of k given the measurement sequence is proportional to $P(v_1, \ldots, v_s|k)$. The algorithm then to compute k given a sequence of s measurements is simple: for each k compute the probability $P(v_1, \ldots, v_s|k)$, outputting the k which maximizes this. The post-processing time required is of order st, which is exponentially large in the number of bits inferred since the value of k that it outputs can be written with $\lceil \log_2(t) \rceil$ bits.

The information theory phase estimation algorithm is given in Algorithm 1.

Algorithm 1 Information Theory Phase Estimation

1: **for** $i = 1$ to s **do**
2:     Choose random M_i. Choose random θ_i.
3:     Perform basic measurement operation with multiple M_i and angle θ_i.
4: **end for**
5: Maximize

$$P(v_1, \ldots, v_s|k) = \prod_{i=1}^{s} P_i(v_i|k)$$

    over all choices of k.
6: **return** k/t, the estimate of the phase.

To illustrate, we simulated the probability of inferring the given angle $2\pi\varphi_k$ among $t = 10^4$ equally distributed possible angles. Figures 2a–2e plot the inferred probability distribution as a function of angle after s measurements, where $s = \{10, 20, 30, 40, 50\}$. The black diamond on each plot indicates the peak at the correct angle. From the plots, we see that after 10–20 measurements, the inference is very noisy, while after 40 or more measurements it has inferred *some* information about the correct angle, and after 50 measurements it is very precise.

In Figure 3, we plot simulation results for inferring the given angle $2\pi\varphi_k$ among $t = \{10^1, 10^2, 10^3, 10^4, 10^5\}$ equally distributed possible angles. The x-axis is the number of random measurements s and the y-axis is the probability that the k which maximizes Eq. (10) after s measurements is the correct angle. Clearly, as t increases, the number of measurements increases, following an $O(\log(t))$ behavior.
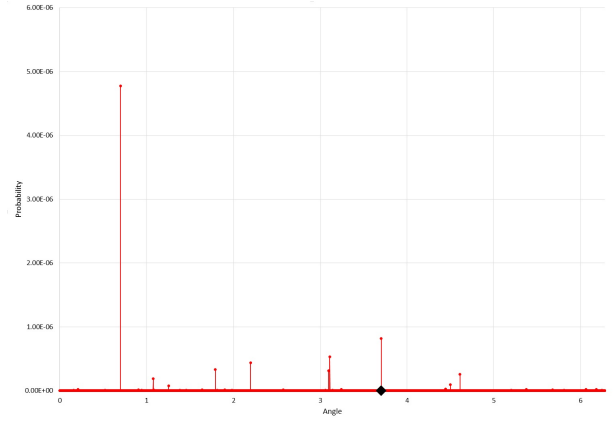
A. Bounds on s

We now show that $O(\log(t))$ measurements suffice to estimate the angle with high probability. This number of measurements required is asymptotically optimal (up to
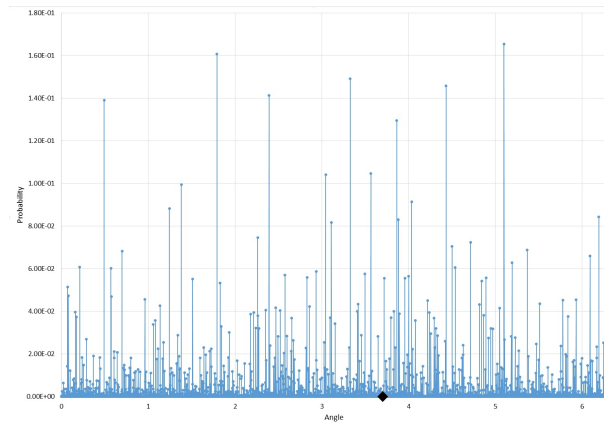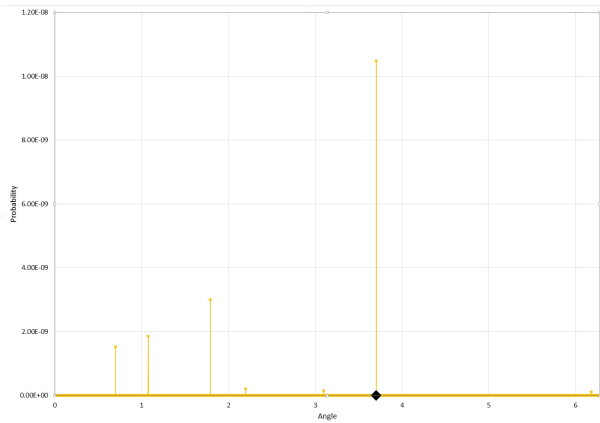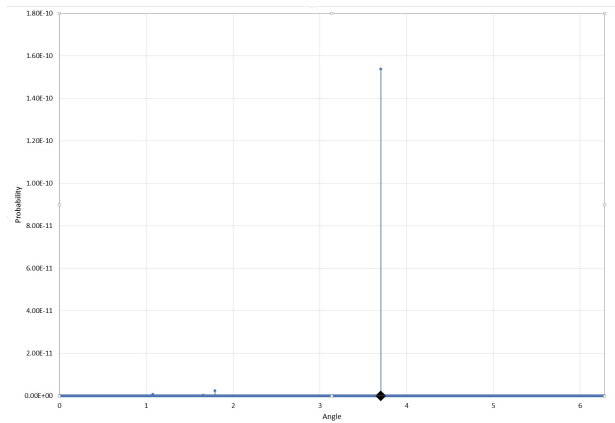
(a) 10 random measurements.
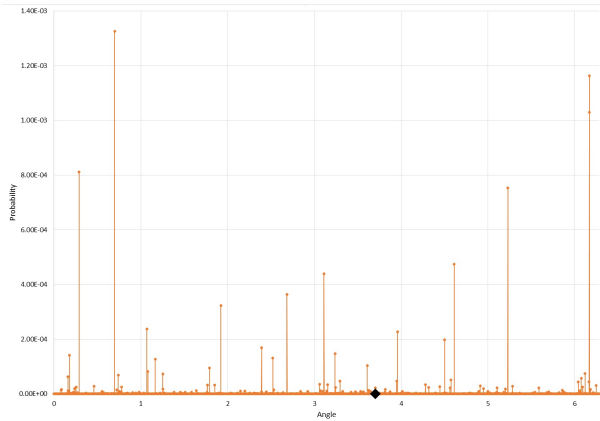
(b) 20 random measurements.

(c) 30 random measurements.

(d) 40 random measurements.

(e) 50 random measurements.

FIG. 2: Results of simulating the probability of inferring a given angle among $t = 10000$ equally distributed possible angles. Plots are of the inferred probability distribution as a function of angle after 10–50 random measurements. The correct angle is marked by a black diamond.

constant factors), as clearly $\lfloor \log_2(t) \rfloor$ measurements are required to have an error probability greater than $1/2$: after s measurements, there are at most 2^s possible outcomes for the sequence of measurements, so to select an angle from a set of t choices with probability greater than $1/2$, we need $2^s > t/2$. A more sophisticated entropic argument would likely be able to improve the constant in front of this lower bound.

The next theorem implies that the number of measurements to obtain error probability at most ϵ is $\log_{1/c}(t/\epsilon)$
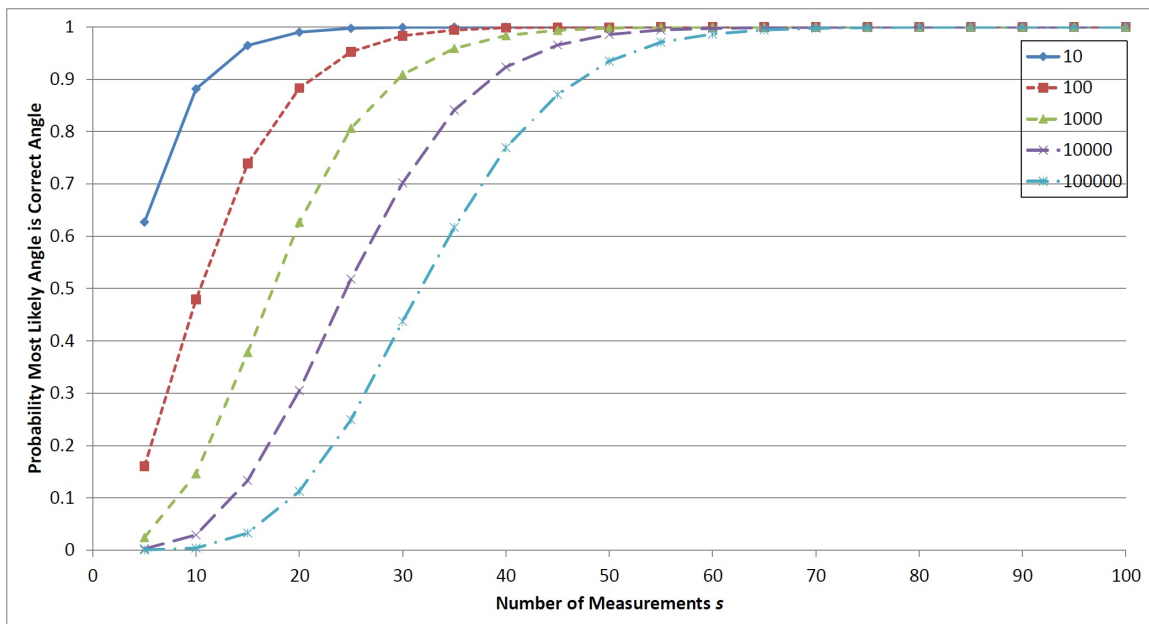
FIG. 3: Results of simulating "information theory" phase estimation for $t = \{10^1, 10^2, 10^3, 10^4, 10^5\}$ equally distributed possible angles. The x-axis is the number of random measurements s; the y-axis is the probability that the most likely angle after s measurements is the correct angle.

for some constant $c < 1$.

Theorem 1. *Suppose we choose the multiples M_i and angles θ_i at random as above. Suppose the measurement outcomes are chosen with probabilities given in Eqs. (8,9) for $k = k_0$. Then, the probability ϵ that the algorithm described above chooses a $k' \neq k_0$ as the choice with maximal likelihood is bounded by*

$$tc^s \qquad (11)$$

for some numerical constant c strictly less than 1 (c does not depend upon t).

Proof. We first consider a given $k' \neq k_0$ and estimate the probability that after s measurements, the probability $P(v_1, \ldots, v_s|k') = \prod_{i=1}^s P_i(v_i|k')$ is greater than or equal to $P(v_1, \ldots v_s|k_0)$. Consider the expectation value

$$E\left[\frac{P(v_1, \ldots, v_s|k')^{1/2}}{P(v_1, \ldots, v_s|k_0)^{1/2}}\right], \qquad (12)$$

where the expectation value is over measurement outcomes and choices of M_i and θ_i. This equals

$$E_{\{M_i, \theta_i\}}\left[\sum_{\{v_i\}} \frac{P(v_1, \ldots, v_s|k')^{1/2}}{P(v_1, \ldots, v_s|k_0)^{1/2}} P(v_1 \ldots, v_s|k_0)\right]$$

$$= E_{\{M_i, \theta_i\}}\left[\sum_{\{v_i\}} P(v_1, \ldots, v_s|k')^{1/2} P(v_1, \ldots, v_s|k_0)^{1/2}\right],$$

where the sum is over all 2^s possible sequences $v_1, \ldots, v_s$ of measurement outcomes and the expectation value is

now over all choices of θ_i, M_i. This equals

$$\left(E_{M,\theta}\left[\sum_v P_{M,\theta}(v|k')^{1/2} P_{M,\theta}(v|k_0)^{1/2}\right]\right)^s, \qquad (13)$$

where $E_{M,\theta}[...]$ is the expectation value over M, θ. A direct calculation shows that for all $k' \neq k$, the term in parenthesis $E_{M,\theta}[\sum_v P_{M,\theta}(v|k')^{1/2} P_{M,\theta}(v|k_0)^{1/2}]$ is bounded by some constant $c < 1$ for all t. Thus, the expectation value (12) is bounded by c^s. Thus, for a given k', the probability that $P(v_1, \ldots, v_s|k') \geq P(v_1, \ldots, v_s|k_0)$ is bounded by c^s, as can be shown by applying Markov's inequality to $\frac{P(v_1, \ldots, v_s|k')^{1/2}}{P(v_1, \ldots, v_s|k_0)^{1/2}}$.

Thus, the probability that there is a k' such that $P(v_1, \ldots, v_s|k') \geq P(v_1, \ldots, v_s|k_0)$ is bounded by tc^s. $\square$

We have not bothered to optimize the estimate in the above theorem: it is possible that a tighter bound could be considered by estimating the expectation value $E[\left(\frac{P(v_1, \ldots, v_s|k')}{P(v_1, \ldots, v_s|k_0)}\right)^a]$ for some constant $0 < a < 1$ and optimizing the choice of a in the spirit of the Chernoff bound.

Finally, we remark that while we have selected θ randomly between 0 and 2π in the above algorithm and in the above theorem, in fact it would suffice to pick θ randomly from the set of angles $\{0, \pi/2\}$, or indeed from any set of a pair of angles that do not differ by exactly π (for example, the set $\{0, \pi\}$ would not work). The proof of the theorem would be essentially the same in this case, and restricting to such a smaller set of angles may be more convenient for implementation on a quantum computer.

B. Classical Inference of Multiple Fourier Modes

The results above suggest a natural generalization of the problem. Define a classical channel $E(x)$ which maps from a real number x between -1 and 1 to an output consisting of a single bit. We fix the output probabilities of this channel:

$$P(0|x) = \frac{1+x}{2}, \qquad (14)$$

$$P(1|x) = \frac{1-x}{2}. \qquad (15)$$

Then Eqs. (8,9) can be interpreted as follows: for $\theta_i = 0$, for any M_i, we take the number $\cos(2\pi M_i \cdot \varphi_k)$ and input this number into the channel and the output of the channel is the measurement outcome, while for $\theta_i = 1$, we instead input $\sin(2\pi M_i \cdot \varphi_k)$.

This then suggests a natural generalization. Consider a classical signal written as a sum of Fourier modes:

$$f(M) = \sum_k a(k) \exp(2\pi i M \cdot \varphi_k). \qquad (16)$$

Here, M is an integer and the function is periodic with period t.

Then, we have the natural classical problem:

Problem 1. *Assume that $f(M)$ is K-sparse, meaning that at most K of the coefficients $a(k)$ are non-zero. Assume that the non-zero $a(k)$ are chosen from a discrete set S of possible values (typically we will be interested in $|S|$ being small), with $\min_{a \neq b, a \in S, b \in S} |a - b| \geq d_{min}$ for some d_{min}. The $a(k)$ may be complex.*

Let A_{max} be the maximum of $|f(M)|$ over all such K-sparse $a(k)$ and over all M.

Assume that we have some channel $C(x)$ which maps from a real number in the range $[-A_{max}, A_{max}]$ to an output chosen from a discrete set (the channel $C(x)$ need not be the same as that given above in Eqs. (14,15)). For this channel to be useful in inferring x from measurements of the output, we will require that different input numbers lead to different output probabilities, and we will quantify this more precisely below in Eq. (17).

Pick several different M_i, and for each M_i measure $C(\mathrm{Re}(f(M_i)))$ or $C(\mathrm{Im}(f(M_i)))$. Infer the coefficients $a(k)$.

This problem can be interpreted as inferring a classical sparse signal from noisy measurements at several different "times" (interpreting each M_i as a time at which to infer the signal). We now show, given suitable assumptions on $C(x)$, that this problem can be solved using a number of measurements that is $O(\log(N_{choices}))$, where $N_{choices}$ is the number of possible choices of K-sparse $f(M)$. As before, this number of measurements

is asymptotically optimal. Note that for $K << t$, $\log(N_{choices}) \approx K \log(t|S|)$.

The procedure we describe is similar to that previously: we select random M_i and randomly choose whether to measure $C(\mathrm{Re}(f(M_i)))$ or $C(\mathrm{Im}(f(M_i)))$ at each time. After s measurements, we select the choice of $a(k)$ which has the maximal a posteriori likelihood, assuming a flat initial distribution. Interestingly, since the number of measurements we need is asymptotically much smaller than $\sqrt{t}$ (indeed we only need $O(\log(t))$ measurements if $K = O(1)$), this means that this random procedure typically does not ever pick $M_i = M_j$ for $i \neq j$. That is, interpreting the M_i as "times", this means that we do not ever measure the signal twice at the same time.

Note that we have assumed that the non-zero coefficients are chosen from a small set S of possible values. As the number of possible values of S increases, the number of measurements increases for two reasons. First $N_{choices}$ increases. Second, the values in the set become more closely spaced (d_{min} becomes smaller compared to A_{max}), and the measurement outcomes probabilities hence become less sensitive to the particular value of $a(k)$. This second problem is actually the more serious one. Suppose that we have a signal that is 1-sparse, and we even know that the only non-zero a_k is at $k = 0$. The question is to infer the magnitude of a_0. Every measurement then consists of sending a_0 into the channel $C(x)$. Using the channel $C(x)$ before, it takes $1/\epsilon^2$ measurements to infer a_0 to precision ϵ. This number of measurements is exponential in the number of bits of precision in a_0. That is, it takes many more measurements to infer the amplitude of a Fourier coefficient than it does to infer its frequency.

Theorem 2. *Suppose we choose the multiples M_i at random as above and randomly choose whether to measure $C(\mathrm{Re}(f(M_i)))$ or $C(\mathrm{Im}(f(M_i)))$ at each time. Suppose also that $C(x)$ has the probability that for any $x, y \in [-A_{max}, A_{max}]$ we have*

$$\sum_v P(v|x)^{1/2} P(v|y)^{1/2} \leq 1 - c_0 |x - y|^2 \qquad (17)$$

for some constant c_0, where the probabilities $P(v|x)$ are the probability that the channel C gives output v given input x. Then, the probability ϵ that the algorithm described above chooses a $k' \neq k_0$ as the choice with maximal likelihood is bounded by

$$N_{choices} c^s \qquad (18)$$

where

$$c \leq 1 - c_0 \left(\frac{d_{min}}{2}\right)^2 \frac{d_{min}^2}{16 A_{max}^2}. \qquad (19)$$

Proof. Assume the correct choice of $a(k)$ is given by $a_0(k)$. We consider a given sequence $a'(k)$ (such

that for at least one k, $a'(k) \neq a_0(k)$) and estimate the probability that after s measurements, the probability $P(v_1, \ldots, v_s|a'(k))$ is greater than or equal to $P(v_1, \ldots v_s|a_0(k))$, where $v_1, \ldots, v_s$ are the measurement outcomes of the channel.

Let

$$f_0(M) = \sum_k a_0(k) \exp(2\pi i M \cdot \varphi_k) \qquad (20)$$

and

$$f'(M) = \sum_k a'(k) \exp(2\pi i M \cdot \varphi_k). \qquad (21)$$

Consider the expectation value

$$E\left[\frac{P(v_1, \ldots, v_s|a'(k))^{1/2}}{P(v_1, \ldots, v_s|a_0(k))^{1/2}}\right], \qquad (22)$$

where the expectation value is over measurement outcomes and choices of M_i and choices of real or imaginary part. This equals

$$E_{\{M_i, R_i\}}\left[\sum_{\{v_i\}} \frac{P(v_1, \ldots, v_s|a'(k))^{1/2}}{P(v_1, \ldots, v_s|a_0(k))^{1/2}} P(v_1 \ldots, v_s|a_0(k))\right]$$

$$= E_{\{M_i, R_i\}}\left[\sum_{\{v_i\}} P(v_1, \ldots, v_s|a'(k))^{1/2} P(v_1, \ldots, v_s|a_0(k))^{1/2}\right],$$

where the sum is over all possible sequences $v_1, \ldots, v_s$ of measurement outcomes and the expectation value is now over all choices of θ_i and of real or imaginary part ($R_i = 0, 1$ is used to denote a measurement of real or imaginary part). This equals

$$\left\{\frac{1}{t} \sum_{M=0}^{t-1} \left(\frac{P(v|\mathrm{Re}(f_0(M)))^{1/2} P(v|\mathrm{Re}(f'(M)))^{1/2}}{2} \right.\right.$$

$$\left.\left. + \frac{P(v|\mathrm{Im}(f_0(M)))^{1/2} P(v|\mathrm{Im}(f'(M)))^{1/2}}{2}\right)\right\}^s. (23)$$

Below, we will use the assumptions on $C(x)$ to show that the term in parenthesis in Eq. (23) is bounded by some constant $c < 1$ for all t. Using this bound, the expectation value (22) is bounded by c^s. Thus, for given $a'(k)$, the probability that $P(v_1, \ldots, v_s|a'(k)) \geq P(v_1, \ldots, v_s|a_0(k))$ is bounded by c^s. Thus, the probability that there is an $a'(k)$ such that $P(v_1, \ldots, v_s|a'(k)) \geq P(v_1, \ldots, v_s|a_0(k))$ is bounded by $N_{choices}c^s$, as claimed.

We now bound the term in parenthesis in Eq. (23). Consider $\frac{1}{t} \sum_M |f'(M) - f_0(M)|^2$. This is greater than for d_{min}^2. Also, for every M, $|f'(M) - f_0(M)|^2 \leq 4A_{max}^2$. So, for randomly chosen M, the probability that $|f'(M) - f_0(M)|^2$ is greater than or equal to $d_{min}^2/2$ is at least $d_{min}^2/8A_{max}^2$. So, the probability that if we randomly choose M and randomly choose whether to measure real or imaginary part, that the corresponding part (i.e., either real or imaginary) of $f'(M) - f_0(M)$ is greater than $d_{min}/2$ in absolute value is at least $d_{min}^2/16A_{max}^2$.

Hence, by the assumption (17) on $C(x)$, we have that the term in parenthesis in Eq. (23) is bounded by

$$c \leq 1 - c_0\left(\frac{d_{min}}{2}\right)^2 \frac{d_{min}^2}{16A_{max}^2}. \qquad (24)$$

$\square$

IV. KITAEV'S PHASE ESTIMATION ALGORITHM

Recall from Section II B that if we apply a large number of measurements using two different values of M at both $\theta = 0$ and $\theta = \pi/2$, we can accurately estimate $\cos(2\pi M \cdot \varphi_k)$ and $\sin(2\pi M \cdot \varphi_k)$, and therefore determine φ_k. In this section, we review Kitaev's phase estimation algorithm to determine φ_k with exponential precision [10] (for complete details, we refer the reader to Sec. 13.5 in Ref. 11). This algorithm relies on obtaining accurate measurements at multiples of φ_k. We begin by reviewing how to accurately measure a given multiple of φ_k with constant precision, building up to estimating the phase with exponential precision. We also simulate the algorithm to determine how many measurements are required in practice.

A. Estimating φ_k with Constant Precision

Recall that $\varphi_k = \frac{k}{t} \mod 1$, where $\varphi_k \in \mathbb{R}/\mathbb{Z}$ and $0 \leq k < t < 2^m$. Let $\theta_i = \{0, \pi/2\}$ at random. Using the measurement operator given in Section IV A and Eqs. (6,7), the conditional probability when measuring multiple $M = 1$ is given by:

$$P(0|k) = \frac{1 + \cos(2\pi \cdot \varphi_k + \theta_i)}{2} \qquad (25)$$

We now solve for the conditional probability $P(0|k)$:

$$2P(0|k) - 1 \qquad (26)$$
$$= \cos(2\pi \cdot \varphi_k + \theta_i)$$
$$= \cos(2\pi \cdot \varphi_k)\cos\theta_i - \sin(2\pi \cdot \varphi_k)\sin\theta_i.$$

We make s measurements, choosing $\theta_i \in \{0, \pi/2\}$ randomly, to obtain approximations $P_{\cos}^*$ and $P_{\sin}^*$ close to $\cos(2\pi \cdot \varphi_k)$ and $\sin(2\pi \cdot \varphi_k)$, respectively. Let there be N_c measurements with $\theta_i = 0$. Let $N_c(0)$ denote the number of these measurements having outcome 0 and let $N_c(1)$ denote the number having outcome 1. Then, let

$$P_{\cos}^* = \frac{N_c(0) - N_c(1)}{N_c}. \qquad (27)$$

If there are N_s measurements with $\theta_i = \pi/2$, with $N_s(0)$ of them having outcome 0 and $N_s(1)$ having outcome 1, then let

$$P_{\sin}^* = \frac{N_s(1) - N_s(0)}{N_s}. \qquad (28)$$

Given $P^*_{\cos}, P^*_{\sin}$, our best estimate of φ_k is obtained by taking an arctangent of $P^*_{\sin}/P^*_{\cos}$, choosing the appropriate quadrant.

Equivalently, we can determine multiples M_i of φ_k in the same manner by measuring and obtaining the probability

$$P(0|k) = \frac{1 + \cos(2\pi M_i \cdot \varphi_k + \theta_i)}{2}, \qquad (29)$$

and computing similar estimates P^* and again taking an arctangent.

In practice, how many measurements s are needed to accurately determine $M_i \cdot \varphi_k$? This is analyzed in the next two sections.

B. Estimating φ_k with Exponential Precision

To efficiently achieve exponential precision in our estimate of φ_k, we measure multiples M_i of φ_k. Then we use the measurement results in a classical inference technique to enhance the precision of the estimate. We begin by measuring multiple $M_0 = 2^{m-1}$, then $M_1 = 2^{m-2}$, increasing the precision as we move to $M_{m-1} = 2^0$. Each measurement gives us an estimate of $M_i \varphi_k \mod 1$.

To achieve the desired precision and probability of error, we measure each multiple s times, where in this section, s refers to the number of measurements *per* multiple for both cosine and sine, so that the total number of measurements required is $2ms$. The estimate of $2^{j-1} \cdot \varphi_k$, using methods of Sec. IV A, is denoted as ρ_j.

We introduce binary fraction notation, where $\overline{.\alpha_1 \ldots \alpha_j} = \sum_{p=1}^{j} 2^{-p} \alpha_p$, $\alpha_p \in \{0,1\}$. The output of the algorithm is $\alpha = \overline{.\alpha_1 \ldots \alpha_{m+2}}$, which is an exponentially precise estimate of φ_k:

$$|\alpha - \varphi_k| < \frac{1}{2^{m+2}}. \qquad (30)$$

Kitaev's phase estimation algorithm [10] is given in Algorithm 2.

Algorithm 2 Kitaev's Phase Estimation [10]

1: **for** $j = m - 1$ to 1 **do**
2:      Set ρ_j to the estimate of $2^{j-1} \cdot \varphi_k$ using $O(s)$ measurements per j.
3: **end for**
4: Set $\overline{.\alpha_m \alpha_{m+1} \alpha_{m+2}} = \beta_m$, where β_m is the octant value $\{\frac{0}{8}, \frac{1}{8}, ..., \frac{7}{8}\}$ closest to ρ_m.
5: **for** $j = m - 1$ to 1 **do**
6:      Infer α_j:

$$\alpha_j = \begin{cases} 0 & \text{if } |\overline{.0\alpha_{j+1}\alpha_{j+2}} - \rho_j| \mod 1 < 1/4. \\ 1 & \text{if } |\overline{.1\alpha_{j+1}\alpha_{j+2}} - \rho_j| \mod 1 < 1/4. \end{cases}$$

7: **end for**
8: **return** α, the estimate of the phase.

Note that in Algorithm 2, we modify the inference step in line 6 to use ρ_j, as opposed to using β_j as done in Ref. 11.

C. Simulation Results

How large does s need to be to estimate φ_k to exponential precision? The probability that a given estimate of $2^{j-1}\varphi_k$ differs by more than a given amount from the true value is exponentially small in s, as shown in Ref. 11 using a Chernoff bound. This implies that to accurately compute the word (the entire sequence of bits α), we need s to scale logarithmically with m.

We ran 10000 independent simulations of Algorithm 2, for words of length $m = \{1000, 10000\}$. These word lengths are of particular interest since Shor's algorithm promises computational speed-ups over its classical counterpart for word lengths around 2048–4096. We considered performance of the algorithm as we varied the number of measurements s of each multiple. Figure 4 shows the numerical results. The x-axis is the number of measurements s. The y-axis is the probability, where we plot both the probability of a given bit being wrong (blue) across all bits and simulation runs, and the probability of a given word being wrong (red) across all simulation runs (i.e., the probability that at least one bit in the word is wrong). For both word lengths, we see that s scales logarithmically in m, and does not exceed 64. The number of measurements required thus scales as $O(m \log(m))$. The corresponding classical post-processing circuit scales as $O(m)$ size and $O(\log(m))$ depth.

V. FAST PHASE ESTIMATION

In this section, we extend Kitaev's algorithm for phase estimation by considering inference across multiple bits simultaneously. We begin by describing an algorithm that improves the number of measurements $O(m \log(m))$ in the previous section to $O(m \log(\log(m)))$. This algorithm consists of two "rounds", where the first round is similar to Kitaev's algorithm, but the second round infers multiple bits simultaneously. Having described this algorithm, we then describe how to further improve it by considering more rounds, requiring $O(m \log(\log(\log(m))))$ measurements for three rounds, and so on, ultimately describing an algorithm that requires $O(m \log^*(m))$ measurements, where $\log^*(m)$ is the iterated logarithm and is bounded for all practical purposes by 5. These algorithms all require only an amount of computational time for classical post-processing that is $O(m \log(m))$ as discussed at the end of the section.

The algorithms in this section can be motivated as follows: the limitation of Kitaev's algorithm is that it infers single bits at a time, and requires logarithmically many
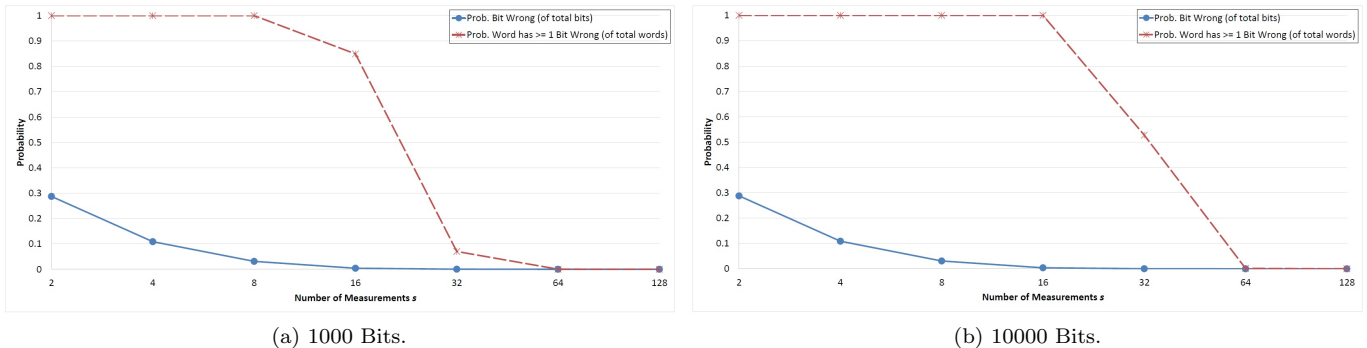
(a) 1000 Bits.

(b) 10000 Bits.

FIG. 4: The number of measurements s versus the probability of a given bit being wrong (blue) and a given word being wrong (red), meaning probability that a given word has at least one bit wrong. (a) Simulation results for words of length 1000 bits. (b) Simulation results for words of length 10000 bits.

measurements per bit. So, a natural generalization is to consider multiples M that are not powers of two, so that we can infer multiple bits at a time. The information theory method does this, by using random M, but requires lots of post-processing. So, in this section we consider "sparse" M, in that the M will be a sum of a small number of powers of two. There is a tradeoff, in that as the "density" (defined to be the number of powers of two) increases, the number of measurements required is reduced, but the postprocessing becomes more complicated. So to make the inference efficient, we use a bootstrapping procedure with multiple rounds, with the density increasing from one round to the next. The early rounds yield only imperfect inferences, but they give enough information to simplify the inference in later rounds.

A. Two Round Algorithm

The measurements that we use in the first round of the two-round algorithm are equivalent to those of Kitaev's algorithm, except that the parameter s will be chosen differently. We set $s = s_1$ for some s_1 chosen later (we call this quantity s_1 as in the second round we will have an s_2, and so on). Using a Chernoff bound estimate as in Ref. 11, we can bound the probability that the difference between $2^{j-1} \cdot \varphi_k$ and our best estimate of $2^{j-1} \cdot \varphi_k$ is greater than $1/16$ by $\exp(-cs_1)$ for some constant $c > 0$. For notational simplicity, we will use one piece of notation that was used in Kitaev's original algorithm: for each j, we will let β_j be the closest approximation in the set $\{\frac{0}{8}, ..., \frac{7}{8}\}$ to the estimate of $2^{j-1} \cdot \varphi_k$. So, the probability of an error larger than $1/8$ in β_j is bounded by $\exp(-cs_1)$.

In the original Kitaev algorithm, we then combine these β_j to estimate φ_k. Instead, our goal in the first round of the two-round algorithm is to give, for almost every j, a quantity called ρ_j that will be an estimate of $2^{j-1} \cdot \varphi_k$ to a precision δ_1, where the subscript 1 is to in-

dicate that this is the precision on the first round. This quantity δ_1 will be much larger than the final precision δ of our two-round algorithm, but will be much smaller than 1. We say "almost every" j because, as we will see, we will only be able to give this precise estimate ρ_j for $0 \le j < m - \log(1/\delta_1)$; however, since $\log(\delta_1)$ will be much smaller than m, this will indeed be most of the j. To compute ρ_j, we use β_{j+l} for $l = 0, ..., \log(1/\delta_1)$ in a Kitaev-style inference procedure to compute $\log(1/\delta_1)+2$ bits in the binary expansion of ρ_j. That is, we obtain the three lowest order bits in the binary expansion from $\beta_{j+\log(1/\delta_1)}$. We then sharpen the estimate, obtaining the lth bit in the binary expansion from β_{j+l-1} and from the $l + 1^{\text{th}}$ and $l + 2^{\text{th}}$ bits, proceeding iteratively. We can bound the probability of error in ρ_j by

$$\Pr\left[\left| \rho_j - 2^{j-1} \cdot \varphi_k \right|_{\mod 1} \ge \delta_1 \right] \le \log(1/\delta_1) \exp(-cs_1).$$
(31)

The factor of $\log(\delta_1)$ occurs because to obtain an error less than δ_1 requires $\log(1/\delta_1)$ bits of precision. This estimate of the probability of error is essentially the same as the estimate of the probability of having an error in Kitaev's original algorithm, except that instead of having m bits in the expansion, we have $\log(1/\delta_1)$ bits. The event of having large error for some given j is uncorrelated with the event of having large error for bits j' if $|j' - j|$ is large enough compared to $\log(1/\delta_1)$. This will play an important role in analyzing the algorithm later, allowing us to neglect certain correlations (we will explain this below, although we will not give a mathematical proof of this). The fact that we have only obtained the accurate estimate of ρ_j for $j \le m - \log(1/\delta_1)$ will not pose a difficulty in what follows; this will be only a minor technical detail. For one thing, most "sets of measurements" (as defined below) do not "contain" (also defined below) the j for which we do not have an accurate estimate. Alternatively, we can simply on the first round infer all ρ_j for $j \le m$ accurately by running the first round on

$m + \log(1/\delta_1)$ bits.

The second round uses $s_2 m$ "sets" of measurements, for some parameter s_2 chosen later, where each set of measurements will consist of repeating the same measurement a total of C times, for some constant C. We also introduce a parameter S, called the "density" in this round. For the ith set of measurements, we pick S different random values of j in the range $1 \le j \le m$, calling these values $j_1^i, ..., j_S^i$. We will require that these values $j_1^i, ..., j_S^i$ all be distinct from each other in a given measurement (if any two are equal, we simply generate another S-tuple of values; we will have $S << \sqrt{m}$ so a random tuple will have distinct entries with probability close to 1). Then we estimate

$$\left(2^{j_1^i - 1} + 2^{j_2^i - 1} + ... + 2^{j_S^i - 1} \right) \varphi_k, \qquad (32)$$

calling this estimate σ_i. We do this estimate using C applications of the basic measurement operation, with $M_i = 2^{j_1^i - 1} + ... + 2^{j_S^i - 1}$ for each measurement and θ_i being chosen randomly in $\{0, \pi/2\}$. The constant C will be chosen so that

$$\mathbf{Pr}\left[\left| M_i \cdot \varphi_k - \sigma_i \right|_{\mod 1} > 1/32 \right] \le \frac{1}{8}. \qquad (33)$$

The constant C is of order unity and does not depend upon m.

This completes the description of the measurements in the two-round algorithm. We now describe the classical post-processing phase. We will explain below how to estimating a quantity β_j' for each j. This quantity will be an approximation to $2^{j-1}\varphi_k$, chosen from the set $\{\frac{0}{8}, ..., \frac{7}{8}\}$. The goal of the algorithm is to obtain an estimate such that for all j we have

$$\mathbf{Pr}\left[\left| 2^{j-1} \cdot \varphi_k - \beta_j' \right|_{\mod 1} > 1/16 \right] \le \frac{\epsilon}{m}, \qquad (34)$$

for some constant ϵ. Thus, by a union bound, the probability of an error greater than $1/8$ in any of the β_j' will be bounded by ϵ. We then use the β_j' to determine the α_j using a procedure similar to Kitaev's algorithm. This procedure is given in Algorithm 3, steps $13 - 17$.

Algorithm 3 Fast Phase Estimation

1: First Round:
2: **for** $j = m - 1$ to 1 **do**
3:     Estimate $2^{j-1} \cdot \varphi_k$ using $O(1)$ measurements per j.
4: **end for**
5: Later Rounds:
6: **for** $r = 2$ to Number of Rounds **do**
7:     Set density, S, and number of measurements per bit, s_r, for given round.
8:     **for** $i = 1$ to $s_r m$ **do**
9:         Set M_i to a sum of S different powers of two, choosing these powers of two at random or with a pseudo-random distribution. Perform $O(1)$ measurements with given M_i and random or pseudo-random θ.
10:     **end for**
11: **end for**
12: Perform multi-bit inference to determine estimate of $\beta_j' = 2^{j-1} \cdot \varphi_k$ for all j. Use estimates from previous round to give starting point for inference in next round. See text for details.
13: Set $\overline{.\alpha_m \alpha_{m+1} \alpha_{m+2}} = \beta_m'$;
14: **for** $j = m - 1$ to 1 **do**
15:     Infer α_j:

$$\alpha_j = \begin{cases} 0 & \text{if } |\overline{.0\alpha_{j+1}\alpha_{j+2}} - \beta_j'|_{\mod 1} < 1/4. \\ 1 & \text{if } |\overline{.1\alpha_{j+1}\alpha_{j+2}} - \beta_j'|_{\mod 1} < 1/4. \end{cases}$$

16: **end for**
17: **return** α, the estimate of the phase.

If the error is bounded by $1/8$ for all β_j', then the estimate of the phase will be accurate to $2^{-(2n+2)}$.

To estimate $2^{j-1} \cdot \varphi_k$ for a given j, consider all sets of measurements such that one that one of the random values of j_a was equal to j; we say that such a set of measurements "contains j". On average, there will be $s_2 S$ such sets of measurements. Let us first proceed by assuming that there are indeed exactly $s_2 S$ sets of measurements and then later deal with the fluctuations in the number of sets of measurements. On the ith set of measurements, we obtain some estimate of σ_i. Suppose this set contains j. Without loss of generality, let us suppose that $j_1 = j$. Then, given only σ_i and $\rho_{j_2}, ..., \rho_{j_S}$, our best estimate of $2^{j-1} \cdot \varphi_k$ is:

$$\sigma_i - \rho_{j_2} - \rho_{j_3} - ... - \rho_{j_S} \qquad (35)$$

We now bound the probability that the estimate is off by more than $1/16$. We do this by bounding the probability that our value of σ_i differs by more than $1/32$ from the correct value using Eq. (33) and also bounding the probability that our estimate of $\sum_{l=2}^{S} \rho_{j_l}$ differs by more than $1/32$ from the correct value. To bound that probability, we have

$$\Pr\left[\left| \left(\sum_{l=2}^{S} \rho_{j_l} - 2^{j_l - 1} \cdot \varphi \right) \right|_{\mod 1} \ge \frac{1}{32} \right] \le S \log(32S) \exp(-cs_1), \qquad (36)$$

where we have taken $\delta_1 = 1/32S$ in Eq. (31) so that if each quantity $\rho_{j_l} - 2^{j_l-1} \cdot \varphi$ is accurate to within δ_1 then the sum is accurate to within $1/32$. We then use a union bound: if the probability that any given measurement is inaccurate is bounded by $\log(1/\delta_1)\exp(-cs_1)$, then the probability that at least one measurement is inaccurate is bounded by S times that quantity.

We choose $s_1 \sim \log(\log(m))$ and $S \sim \log(m)$ so that the right-hand side of Eq. (36) is bounded by $1/32$. Then, using Eqs. (33,36), the probability that the quantity in Eq. (35) differs by at least $1/16$ from $2^{j-1} \cdot \varphi_k$ is bounded by $1/4$. We get roughly $s_2 S$ different estimates of $2^{j-1} \cdot \varphi_k$, one for each set of measurements involving the given j. Let us assume the independence of certain events between different sets of measurements, namely the event that the quantity in Eq. (35) differs by more than $1/32$ from $2^{j-1} \cdot \varphi_k$ (we discuss this further below). Then, we can combine these measurements to obtain an estimate of β'_j by picking the value of β'_j which is most frequently within $1/16$ of $\sum_{l=2}^{S} \left(\rho_{j_l} - 2^{j_l-1} \cdot \varphi \right)$; i.e., it is within $1/16$ of that value for the greatest number of sets of measurements containing j.

The probability of error in β'_j by more than $1/16$ is then bounded by $\exp(-c's_2 S)$ for some constant $c' > 0$. Picking $s_2 \sim 1$, we find that the probability of error is $1/\mathrm{poly}(m)$ for any desired polynomial, with the power depending upon the ratio between $S/\log(m)$, so we can ensure that this probability is small compared to ϵ/m. The number of measurements required by this procedure is $O(m\log(\log(m)))$.

We now discuss several issues of correlations and fluctuations that were left open in the above analysis. First, consider the fluctuation in the number of sets of measurements that contain j, for any given j. On average this quantity is $s_2 S$, but there may be some fluctuations. However, the probability that there are fewer than $s_2 S/2$ different such sets of measurements is exponentially small in $s_2 S$, and hence for the given choice of $s_2 S$, the quantity is bounded by $1/\mathrm{poly}(m)$ and so can be made negligible (in fact, this probability, being exponentially small in $s_2 S$, has a similar scaling as the probability that we incorrectly infer a given $2^{j-1} \cdot \varphi_k$ given $s_2 S$ sets of that contain j, as that probability is also exponentially small in $s_2 S$). So, with high probability all j are contained in at least $s_2 S/2$ measurements, and so we can double S and apply the analysis above.

It is possible that a better way to deal with fluctuations in the number of measurements is to change the distribution of choices of j_a^i, and anti-correlate the choices in different sets of measurements to reduce the fluctuations in the number of sets of measurements containing a given j. This will at best lead to a constant factor improvement.

Another kind of correlation that we must deal with is correlation between the events that the quantity in Eq. (35) differs by more than $1/16$ from $2^{j-1} \cdot \varphi_k$. For a given j, let us assume for a given set of measurements we have $j_1^i = j$. Let us refer to $j_2^i, ..., j_S^i$ as the "partners" of j. For a given j, the different sets of measurements involving that j will typically have wildly different partners of j; that is, for two different sets of measurements, m, n, we will typically have $|j_a^m - j_b^n| \gtrsim m/S^2 >> \log(S)$ for $a, b \neq 1$. So, for most sets of measurements, these will be independent. Similarly, in a given measurement we will typically have $|j_a^m - j_b^m| >> \log(S)$ so we can ignore correlations between errors in different ρ_{j_a}.

Of course, the above is not a rigorous proof, but we expect that such a proof can be provided without any significant difficulty. Note that if for a given j we have a large number of (roughly) independent sets of measurements containing that j, then adding a small number of correlated sets of measurements will not prevent the inference from working.

B. Multiple Round Algorithm

We can improve this procedure by increasing the number of rounds. In the first and second rounds we proceed as before, though the constants s_1, s_2, S will be changed. Let us write $S = S_2$ for the second round. The third round of the procedure is the same as the second round, except that we do s_3 sets of measurements, and in each measurement we pick S_3 different random values of j. On the third round, as in the second, we repeat each set of measurements C times; it is only the first round where the quantity C does not appear, for the reason that in that round, each measurement is already being repeated s_1 times. We can increase the number of rounds indefinitely. In each round, we can exponentially increase the density compared to the previous round, while keeping all constants s_a of order unity. The number of measurements required is then proportional to the number of rounds. Since S increases exponentially in each round and we need $S \sim \log(m)$ in the last round, the number of rounds required is $\sim \log^*(m)$ and the total number of measurements is $\sim m\log^*(m)$.

C. Classical Post-processing Time Required

The simplest implementation of the algorithm above requires a time $O(m\log^2(m))$. We discuss this first and then discuss how to improve to $O(m\log(m))$. Each bit is contained in $\sim \log(m)$ sets of measurements (indeed, the fact that it is contained in this many sets of measurements is the whole point of the algorithm). To compute the quantity in Eq. (35), the sum on the right-hand side require summing over S different quantities, and for $S \sim \log(m)$, this means that it takes time $\sim \log(m)$ to do the computation for each bit for each set of measurements

containing that bit. So, with m bits, each contained in $\log(m)$ sets of measurements, the time is $O(m\log^2(m))$.

However, we can slightly improve this by noting that Eq. (35) can be written as

$$\sigma_i - \left(\rho_{j_1} + \rho_{j_2} + ... + \rho_{j_S}\right) + \rho_{j_1}. \quad (37)$$

The quantity in parentheses can be computed once for each set of measurements, and re-used in inferring each of the ρ_{j_i} for $i \in \{1,...,S\}$, and then it only requires $O(1)$ time to do the arithmetic for each of these i. This improves the total time to $O(m\log(m))$.

VI. ANALYSIS OF QUANTUM CIRCUIT DEPTH AND WIDTH

The fast phase estimation algorithm offers an asymptotic improvement in the number of measurements required to estimate the phase with exponential precision. How does the corresponding quantum circuit scale, in terms of depth, width and size? We define the *depth* of a quantum circuit as the number of timesteps, where gates on disjoint qubits can occur in parallel in a given timestep. Here we assume that a given n-qubit gate takes one timestep. The *width* of a quantum circuit is the number of qubits. The *size* of a quantum circuit is the total number of non-identity quantum gates. We analyze the circuits given three different computing settings, to emphasize tradeoffs in depth and width depending on resource availability. Table I contains a summary of the circuit resources required for each algorithm given the setting.

First, consider the setting where each measurement operator is performed sequentially. That is, the circuit is given by the sequence of gates (shown in Fig. 5)

$$H^{\otimes ms} Z(\theta_{M_1})\Lambda^1(U^{M_1})[q_1, A] Z(\theta_{M_2})\Lambda^1(U^{M_2})[q_2, A] \ldots$$
$$Z(\theta_{M_{ms}})\Lambda^1(U^{M_{ms}})[q_{ms}, A] H^{\otimes ms}, \quad (38)$$

where $\Lambda^n(U)[q_1, q_2]$ denotes n-qubits in register q_1 controlling the application of gate U to register q_2. The quantum register containing the eigenvector state is denoted by $|A\rangle$ and consists say of a qubits. Each phase estimation algorithm performs $O(ms)$ measurements, resulting in a circuit of depth and size $O(ms)$. The circuit requires $O(ms)$ ancilla qubits, one per measurement, plus a additional qubits. For Kitaev's phase estimation and fast phase estimation, s equals $O(\log(m))$ and $O(\log^*(m))$, respectively. Thus in the sequential setting, fast phase estimation offers an asymptotic improvement in circuit depth and size, as well as in the number of ancilla qubits.

Second, consider a more parallel setting obtained by decreasing circuit depth at the cost of increasing circuit width. We can *parallelize* quantum phase estimation using techniques presented in Refs. 11 and 13. The idea is
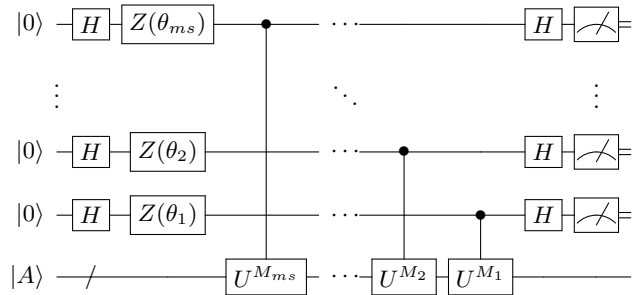

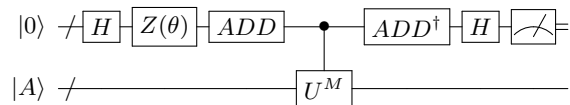FIG. 5: Quantum circuit for sequential phase estimation.

FIG. 6: Quantum circuit for parallel phase estimation. Each wire represents a register of qubits.

to apply one multi-controlled gate instead of the sequence in Eq. (38), by evolving as

$$|M\rangle \otimes |A\rangle \to |M\rangle \otimes U^M |A\rangle, \quad (39)$$

where M is given by the sum of the multiples:

$$M = \sum_{j=1}^{ms} M_j q_j. \quad (40)$$

The sum can be computed using a quantum addition circuit based on a 3-2 quantum adder (also called a carry-save adder) [11, 14, 15], which reduces the sum of three m-bit numbers to a sum of two encoded numbers in $O(1)$ depth, with width $O(m)$ and size $O(m)$. Consider M to be a sum of s m-bit integers. The circuit first uses a $\log(s)$-depth tree of 3-2 adders to produce two encoded numbers, and then adds these two numbers in place using a quantum carry-lookahead adder [16] with $O(m)$ ancillae, $O(m)$ size, and $O(\log(m))$ depth. In total, the addition requires a quantum circuit of $O(ms)$ ancillae, $O(\log(s) + \log(m))$ depth, and $O(ms)$ size.

The circuit for performing parallel phase estimation is shown in Figure 6. The circuit begins with a quantum register containing qubits initialized to $|0\rangle$. Each qubit q_i undergoes a Hadamard operation, followed by a phase rotation by angle θ_i about the z-axis. An addition circuit is applied to determine $|M\rangle$. A controlled U^M operation is applied, followed by an addition circuit to uncompute $|M\rangle$. Finally, $O(ms)$ Hadamard operations and measurements are applied, which can be done in depth $O(1)$. The complete circuit for parallel phase estimation requires

TABLE I: Table of circuit depth, width, and size for Kitaev's quantum phase estimation and fast phase estimation.

Type	Kitaev's Phase Estimation [11]			Fast Phase Estimation		
	Depth	Width	Size	Depth	Width	Size
Sequential	$O(m \log(m))$	$O(m \log(m))$	$O(m \log(m))$	$O(m \log^*(m))$	$O(m \log^*(m))$	$O(m \log^*(m))$
Parallel	$O(\log(m))$	$O(m \log(m))$	$O(m \log(m))$	$O(\log(m))$	$O(m \log^*(m))$	$O(m \log^*(m))$
Cluster	$O(\log(m))$	$O(m^2)$	$O(m \log(m))$	$O(\log^*(m))$	$O(m^2)$	$O(m \log^*(m))$
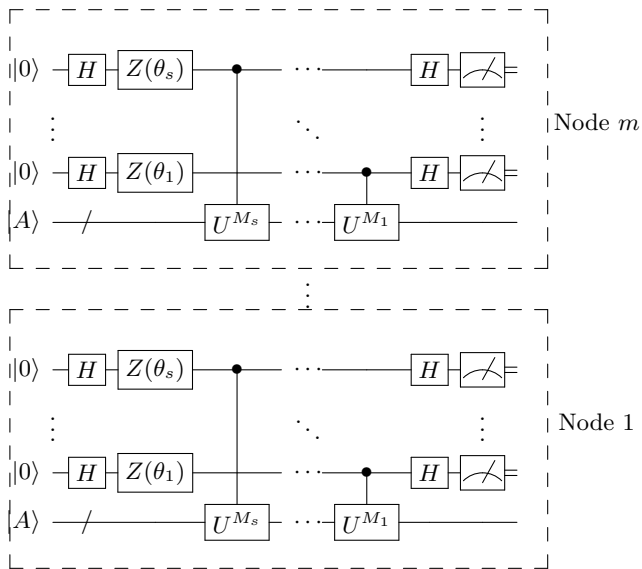
FIG. 7: Quantum circuit for parallel phase estimation across a cluster consisting of m nodes.

$O(ms)$ size and $O(\log(s) + \log(m))$ depth, up to the implementation of the multi-controlled U^M gate. Again, s equals $O(\log(m))$ for Kitaev's phase estimation and $O(\log^*(m))$ for fast phase estimation yielding a significant reduction in circuit size and width to $O(m \log^*(m))$.

Third, consider access to a cluster of quantum computers containing m nodes. Each node performs s measurements, resulting in a depth of $O(s)$, with a size and width *per* node of $O(s)$ gates and $O(s + a)$ qubits, respectively. The cumulative cost across all m nodes is $O(s)$ depth, $O(ms)$ size, and $O(ms + ma)$ qubits. Again, fast phase estimation yields asymptotic improvements in all dimensions, and results, for all practical purposes, in a constant-depth phase estimation circuit. One potential advantage of the cluster model is that errors do not accumulate on the eigenvector state $|A\rangle$, since subsets of measurements are done on separate nodes. This could be advantageous when designing a fault-tolerant phase estimation algorithm.

Table I summarizes the circuit size, depth, and width for the various settings of the two algorithms. Fast phase estimation yields asymptotic improvements in each dimension.

VII. CONCLUSIONS AND FUTURE WORK

We have presented several algorithms for quantum phase estimation based on a basic measurement operation and classical post-processing. Both our "information theory" algorithm and our fast phase estimation algorithm depend upon a randomized construction of which measurements to take, and have applications to classical signal processing and quantum phase estimation. Our fast phase estimation algorithm achieves asymptotic improvements in circuit depth, width, and size over Kitaev's phase estimation, resulting in significant reductions in resource requirements including circuit depth and size, and the number of ancilla qubits. Remarkably, when using an m-node cluster of quantum computers, our algorithm requires essentially constant time. It is an interesting question for future work to de-randomize these algorithms.

[*] ksvore@microsoft.com

[†] mahastin@microsoft.com

[‡] michaelf@microsoft.com

[1] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Science **309**, 1704 (2005).

[2] B. P. Lanyon, J. D. Whitfield, G. G. Gillet, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, A. Aspuru-Guzik, and A. G. White, Nature Chemistry **2**, 106 (2009), 0905.0887.

[3] S. P. Jordan, K. S. M. Lee, and J. Preskill, "Quantum computation of scattering in scalar quantum field theories," (2011), 1112.4833.

[4] P. Shor, SIAM Journal of Computing **26**, 1484 (1997).

[5] K. Temme, T. Osborne, K. Vollbrecht, D. Poulin, and F. Verstraete, Nature **471** (2011), 10.1038/nature09770, 0911.3635.

[6] M. Ozols, M. Roetteler, and J. Roland, in *3rd Innovations in Theoretical Computer Science Conference (ITCS)* (ACM, 2012) pp. 290–308, 1103.2774.

[7] D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **83**, 5162 (1999).

[8] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Proceedings of the Royal Society of London, Series A **454** (1998).

[9] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, U.K., 2000).

[10] A. Y. Kitaev, Electronic Colloquium on Computational Complexity (ECCC) **3** (1996).

[11] A. Y. Kitaev, A. Shen, and M. Vyalyi, *Classical and Quantum Computation* (American Mathematical Society, Providence, Rhode Island, 2002).

[12] P. Selinger, "Efficient clifford+T approximation of single-qubit operators," (2012), 1212.6253.

[13] R. Cleve and J. Watrous, in *FOCS '41* (2000) pp. 526–536, quant-ph/0006004.

[14] P. Pham and K. M. Svore, in *Reversible Computing '12* (2012) 1207.6655.

[15] P. Gossett, "Quantum Carry-Save Arithmetic," (1998).

[16] T. G. Draper, S. A. Kutin, E. M. Rains, and K. M. Svore, Quantum Information and Computaton **6**, 351 (2006), quant-ph/0406142.