Adaptive Quantum Simulated Annealing for Bayesian Inference and Estimating Partition Functions

Aram W. Harrow[1] and Annie Y. Wei[1]

[1]Center for Theoretical Physics, Massachusetts Institute of Technology

Abstract

Markov chain Monte Carlo algorithms have important applications in counting problems and in machine learning problems, settings that involve estimating quantities that are difficult to compute exactly. How much can quantum computers speed up classical Markov chain algorithms? In this work we consider the problem of speeding up simulated annealing algorithms, where the stationary distributions of the Markov chains are Gibbs distributions at temperatures specified according to an annealing schedule.

We construct a quantum algorithm that both adaptively constructs an annealing schedule and quantum samples at each temperature. Our adaptive annealing schedule roughly matches the length of the best classical adaptive annealing schedules and improves on nonadaptive temperature schedules by roughly a quadratic factor. Our dependence on the Markov chain gap matches other quantum algorithms and is quadratically better than what classical Markov chains achieve. Our algorithm is the first to combine both of these quadratic improvements. Like other quantum walk algorithms, it also improves on classical algorithms by producing "qsamples" instead of classical samples. This means preparing quantum states whose amplitudes are the square roots of the target probability distribution.

In constructing the annealing schedule we make use of amplitude estimation, and we introduce a method for making amplitude estimation nondestructive at almost no additional cost, a result that may have independent interest. Finally we demonstrate how this quantum simulated annealing algorithm can be applied to the problems of estimating partition functions and Bayesian inference.

1 Introduction

Grover search yields a quadratic speedup over classical exhaustive search for the problem of unstructured search. A major challenge in quantum algorithms is to extend this quadratic speedup to more structured search problems. One particularly important case is Markov chain Monte Carlo algorithms, which make it possible to efficiently sample from the stationary distribution of a Markov chain. Markov chain Monte Carlo methods have applications both in Bayesian inference, where such methods are used to sample from a posterior distribution which might otherwise be difficult to compute directly, and in counting problems [11, 6] via the connection between approximate counting and sampling.

However, it is currently an open question whether there exists a completely quantum analog of the classical Markov Chain Monte Carlo algorithm. While quantum walks [26] yield quadratically faster mixing in a variety of special cases [22], there is no general quadratic speedup known for MCMC sampling. Classical Markov chains are known to mix in time $O(\delta^{-1} \log(1/\min_x \Pi(x)))$ [2], where δ is the spectral gap of the Markov chain, and $\Pi(x)$ denotes the stationary distribution, while in the most general case quantum Markov chains have been shown to mix in time $O(1/\sqrt{\delta \min_x \Pi(x)})$ [15]. Even though a recent result [3] achieved a quadratic speedup in hitting time for the problem of searching for marked elements, the technique used there, that of quantum fast-forwarding [4], will not yield a quadratic speedup for MCMC sampling as it also scales like $O(1/\min_x \Pi(x))$. In the regime where $(1/\min_x \Pi(x))$ scales with the size of the search space, the resulting quantum scaling is exponentially worse than the scaling of classical MCMC.

Indeed, there are well-known barriers to a general quantum speedup. First, directed Markov chains are general enough to encompass any randomized classical algorithm, but there are oracle problems, such

as parity, for which quantum algorithms cannot obtain more than a constant speedup, so any such speedup would need to rely on structural features of the Markov chain. Second, many natural quantum walks that produce a classical sample do so by measuring a state whose amplitudes are all nonnegative reals, which means that they could prepare such a state at no extra cost. Such a state is called a qsample [1] and is the coherent encoding of the stationary distribution of the classical Markov chain. If qsamples could be prepared even polynomially more slowly than the mixing time of classical Markov chains, let alone quadratically faster, then this would imply the unlikely conclusion that SZK $\subseteq$ BQP [1, 20].

As a result, there are several distinct approaches to the problem of qsampling and state generation, and we briefly survey these approaches in Section 1.1. The approach that we shall employ, that of quantum simulated annealing (QSA) [23, 24, 31, 32], relies on qsampling the stationary distributions of a series of intermediate Markov chains. Successive stationary distributions satisfy a "slow-varying condition" $|\langle\Pi_i|\Pi_{i+1}\rangle|^2 \geq$ const, which allows these algorithms to bound the dependence on $\min_x \Pi(x)$ while preserving the $O(1/\sqrt{\delta})$ square root scaling in the spectral gap. Such algorithms do so at the cost of also scaling with the length of the annealing schedule ℓ, and in this work we will show how to reduce the length ℓ.

Our work relies on two previous algorithmic results. First is the QSA algorithm of Wocjan and Abeyesinghe [31], who showed how to qsample from the last of a series of Markov chains. Specifically, given a series of ℓ Markov chains such that the first Markov chain is easy to qsample, all the spectral gaps are lower bounded by δ, and the stationary states have constant overlap, qsampling from the last Markov chain can be performed using $\tilde{O}(\ell/\sqrt{\delta})$ total Markov chain steps. This is important because quantum walks naturally yield reflections about the stationary state, so this gives an efficient way to turn the ability to reflect into the ability to qsample. However, it does not give us a good way to bound the length ℓ. If $Z = \sum_x e^{-H(x)}$ for some $H(x) \geq 0$ then we can naively bound $\ell \leq \max_x H(x)$. A somewhat better bound is $\ell \lessapprox F := \log(1/Z)$. We use the notation F because this quantity is called the "free energy" in statistical physics. More precisely, $\ell \leq (1+F)\log\log|\Omega|$ where Ω is the state space, and this sequence can be found knowing only a bound on F; see Lemma 3.2 of [25]. This linear scaling with F cannot be improved for such nonadaptive schedules.

However, a better sequence of Markov chains can be found if we are willing to choose them *adaptively*, i.e. based on information we extract from our samples as we run the algorithm. The second result we use is due to Štefankovič, Vempala, and Vigoda (SVV) [25], who gave a classical algorithm for finding adaptive sequences of Markov chains of length $\tilde{O}(\sqrt{F})$, an almost quadratic improvement. (Note that [8] gives a simpler classical algorithm for finding quadratically shorter sequences, but it requires that the Hamiltonian not change sign, limiting its application beyond counting problems.) At first glance, such adaptive algorithms appear difficult to quantize since extracting information from qsamples, say in order to determine the adaptive sequence, will generally damage the states. Indeed, the only quantum algorithm to use SVV was Montanaro's [18] quantum algorithm for summing partition functions, which uses the QSA algorithm of Wocjan and Abeyesinghe [31] to partially quantize a classical algorithm for summing partition functions. However, while [18] could *use* the adaptive sequence in its quantum algorithm, it had to rely on classical methods to *compute* the sequence from SVV, which limited its quantum speedup.

Our work combines the QSA algorithm of Wocjan and Abeyesinghe [31] with a fully quantized version of the work of SVV, achieving a runtime of $\tilde{O}(\sqrt{F/\delta})$. In other words we adaptively obtain a sequence matching the length from SVV (i.e. $\ell = \tilde{O}(\sqrt{F})$) while also achieving the square-root scaling with $1/\delta$ from previous QSA algorithms [23, 24, 31]. In doing so we show that amplitude estimation [5] can be made nondestructive using a state restoration scheme inspired by [28], a result we believe will be useful in its own right.

We also show that this algorithm can be applied both to the problem of estimating the partition function in counting problems and to the problem of Bayesian inference, as both problems share a general structure. In the counting problem we have a partition function of the form $Z(\beta) = \sum_{k=0}^{n} a_k e^{-\beta k}$, and we would like to estimate the quantity $Z(\infty) = a_0$, which is hard to compute, by annealing from $Z(0)$. In the Bayesian inference problem we have a prior $\Pi_0(\theta)$ and a likelihood function $L(\theta)$, and we would like to sample from the hard-to-compute posterior distribution $\Pi_1(\theta) = \Pi_0(\theta)L(\theta)/Z$ by annealing through the intermediate distributions $\Pi_\beta(\theta) = \Pi_0(\theta)\exp(\beta L(\theta))/Z_\beta$. We obtain the following theorem as our main result, which we also summarize in Table 1.

Theorem 1 (Informal statement of main results)**.**

1. Bayesian inference. *Given a prior* $\Pi_0(\theta)$ *and a likelihood function* $L(\theta)$*, define distributions* $\Pi_\beta(\theta) \propto \Pi_0(\theta)\exp(\beta L(\theta))$ *for* $\beta \in [0,1]$*. Suppose that for each* β *we can compute a Markov*

Problem	Our Result	Best Previous Result	Best Classical Result
Counting Problems	$\tilde{O}(\log \lvert\Omega\rvert/(\sqrt{\delta}\epsilon))$	$\tilde{O}(\log \lvert\Omega\rvert/(\sqrt{\delta}\epsilon) + \log \lvert\Omega\rvert/\delta)$	$\tilde{O}(\log \lvert\Omega\rvert/(\delta\epsilon^2))$
Bayesian Inference	$\tilde{O}(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)]/\delta})$	$\tilde{O}(\max_\theta L(\theta)/\sqrt{\delta})$	$O(\max_\theta L(\theta)/\delta)$

Table 1: Summary of main results. Here δ denotes the spectral gap of the Markov chain. Letting n be the maximum upper range for the counting problem (equivalently, the maximum value of the Hamiltonian), typically $|\Omega| = Z(0) \sim \exp(n)$ and $\delta \sim \text{poly}(n)$. $L(\theta)$ denotes the likelihood function for the Bayesian inference problem and likewise corresponds to values of the Hamiltonian. Our results are formalized in Theorems 10 and 14. The previous best [quantum] results for counting and Bayesian inference are due to Montanaro [18] and Wocjan-Abeyesinghe [31] respectively. The classical algorithm for counting is due to Štefankovič, Vempala and Vigoda [25] and the algorithm for Bayesian inference simply uses simulated annealing with the nonadaptive schedule in [25].

chain M_β with stationary distribution Π_β and with gap $\geq \delta$. Then we can qsample from $|\Pi_1\rangle$ using $\tilde{O}(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)]/\delta})$ steps of the quantum walks corresponding to various M_β.

2. Estimating partition functions. *Let $Z(\beta) = \sum_x e^{-\beta H(x)}$ with $H(x) \geq 0$ and suppose again that we have access to Markov chains M_β with gaps $\geq \delta$ and stationary distributions $\propto e^{-\beta H(x)}$. Then we can estimate $Z(\infty)$ to multiplicative error ϵ with high probability using $\tilde{O}(\log(Z(0))/\sqrt{\delta}\epsilon)$ steps of the quantum walks corresponding to M_β.*

These are formalized as Theorems 10 and 14. In each case we match the schedule length of SVV's adaptive algorithm and the gap dependence of Wocjan-Abeyesinghe, thus improving on all previous algorithms. An important subroutine in our results is a nondestructive version of amplitude estimation, formalized below in Theorem 6 and described in detail in Section 4.

We also consider applications of the partition function algorithm to representative problems from statistical physics and computer science, again improving on previous algorithms. Our results are summarized in Table 2 and discussed in more detail in Section 5.1.

This paper is organized as follows: in the rest of this introduction we briefly survey related work and provide a technical overview of our work. In Section 2 we show that there exists an adaptive cooling schedule by slightly modifying the arguments of SVV to also work in the Bayesian inference case. This adaptive cooling schedule then translates into a temperature schedule that is quadratically shorter than any nonadaptive schedule in both the Bayesian inference and counting problem cases. In Section 3 we describe the quantum algorithm which both constructs the adaptive cooling schedule and anneals to the quantum sample at each temperature. Applying this algorithm to Bayesian inference and the counting problem, we establish our main result Theorem 1, formalized as Theorems 10 and 14. In Section 4 we describe, in detail, how to perform state restoration following amplitude estimation at almost no additional cost. In Section 5.1 we consider applications of the partition function algorithm to representative problems from statistical physics and computer science, and in Section 5.2 we discuss warm starts for speeding up Markov chain mixing times, as well as how they have been incorporated into the algorithms of Section 3. Our conclusion is in Section 5.3.

1.1 Related Work

Here we briefly describe alternative approaches to the problem of qsampling and state generation, noting some benefits and drawbacks of each approach when compared with QSA.

- **Direct generation**: An approach due to Zalka [33], rediscovered independently by Grover and Rudolph [7] and Kaye and Mosca [13], generates the state directly via rotations, but its scope is limited as it is only efficient in the special case where the probability distribution is efficiently integrable.

- **Adiabatic state generation**: Aharanov and Ta-Shma [1] offer an approach to qsampling via adiabatic computing, but their approach scales like $O(1/\delta)$ in the spectral gap. Thus, while it produces qsamples instead of samples, it offers no speedup over the classical case.

- **Metropolis sampling**: An approach by [28] that relies on Metropolis sampling generalizes qsampling to quantum Hamiltonians, but it likewise scales like $O(1/\delta)$ in the spectral gap. [32] combines Metropolis sampling with QSA to extend the $O(\ell/\sqrt{\delta})$ scaling of QSA to quantum

Problem	Our Result	Best Previous Result	Best Classical Result
Counting k-colorings	$\tilde{O}(\|V\|^{3/2}/\epsilon)$	$\tilde{O}(\|V\|^{3/2}/\epsilon + \|V\|^2)$	$\tilde{O}(\|V\|^2/\epsilon^2)$
Ising model	$\tilde{O}(\|V\|^{3/2}/\epsilon)$	$\tilde{O}(\|V\|^{3/2}/\epsilon + \|V\|^2)$	$\tilde{O}(\|V\|^2/\epsilon^2)$
Counting matchings	$\tilde{O}(\|V\|^{3/2}\|E\|^{1/2}/\epsilon)$	$\tilde{O}(\|V\|^{3/2}\|E\|^{1/2}/\epsilon + \|V\|^2\|E\|)$	$\tilde{O}(\|V\|^2\|E\|/\epsilon^2)$
Counting independent sets	$\tilde{O}(\|V\|^{3/2}/\epsilon)$	$\tilde{O}(\|V\|^{3/2}/\epsilon + \|V\|^2)$	$\tilde{O}(\|V\|^2/\epsilon^2)$

Table 2: Summary of applications to estimating the partition function in counting problems. See the text of section 5.1 for discussion and references.

Hamiltonians, but the scaling is otherwise equivalent to that of other QSA algorithms.

- **Quantum rejection sampling**: In quantum rejection sampling [21, 14, 30], to obtain target state $|\Pi\rangle$ we instead prepare some superposition of the desired state $|\Pi\rangle$ and an undesired state $|\Pi^\perp\rangle$ and then apply amplitude amplification to obtain $|\Pi\rangle$. As [30] notes, this scheme is generally inefficient; to deal with this, [14] specializes to the case of distributions structured as a Bayesian network, while [30] employs semi-classical Bayesian updating. Even then, the algorithm of [30] still scales like $O(1/\sqrt{\epsilon Z})$ per update in ϵ, the approximation error, and Z, the partition function of the posterior distribution, whereas our algorithm's scaling is $\sim \sqrt{\delta^{-1}\log(1/Z)}\log(1/\epsilon)$. (These scalings depend on the normalization convention used for Z; see Section 1.2.1.) This scaling is generally better because δ can often be improved with a good choice of Markov chain, and when these chains are rapidly mixing $1/\delta$ will be poly $\log(1/Z)$.

1.2 Technical Overview

Here we describe adaptive annealing schedules and their application to counting problems and Bayesian inference. Then we describe our quantum algorithm for finding and annealing through such a schedule.

1.2.1 Adaptive Annealing Schedules for Counting Problems and Bayesian Inference

In both the counting problem and the Bayesian inference problem, we have a partition function of the form

$$Z(\beta) = \sum_{x\in\Omega} e^{-\beta H(x)} \tag{1}$$

at inverse temperature β, with a Hamiltonian we denote by $H(x)$ for some random variable x over state space Ω. We assume that $H(x)$ is easy to compute (say a sum of local terms) and Ω is also a simple set, such as $\{0,1\}^n$, although it may also be a non-product set such as the set of permutations. Such a partition function corresponds to the normalization of the Gibbs distribution at inverse temperature β, which is given by

$$\Pi_\beta(x) = \frac{e^{-\beta H(x)}}{Z(\beta)}. \tag{2}$$

In the counting problem of SVV [25] and Montanaro [18], the Hamiltonian takes on values $k \in \{0,\ldots,n\}$ corresponding to a discrete quantity we would like to count, such as the number of colorings of a graph, or the number of matchings. In Section 5.1 we give several examples of problems from statistical physics and computer science that can be framed in this form. In such problems we would have a partition function of the form

$$Z(\beta) = \sum_{k=0}^{n} a_k e^{-\beta k}, \tag{3}$$

where $a_k = |H^{-1}(k)|$. In general we do not need the energy function to take on only integer values but it will be convenient to assume that $0 \le H(x) \le n$ for all x.

We want to estimate the quantity $Z(\infty) = a_0$, which is often difficult to compute, while $Z(0) = \sum_k a_k = |\Omega|$, corresponding simply to the size of the parameter space, is easy to compute. The idea is to establish a schedule of $\ell+1$ inverse temperatures $\beta_0, \beta_1, \ldots, \beta_\ell$, with $\beta_0 = 0$ and $\beta_\ell = \infty$, known as a cooling schedule, that allows us to anneal from the easy case of $\beta = 0$ to the hard case of $\beta = \infty$. Once we have a cooling schedule, we can sample from the Gibbs distribution at each inverse temperature β_i, given by

$$\Pi_{\beta_i}(x) = \frac{e^{-\beta_i H(x)}}{Z(\beta_i)}. \tag{4}$$

Then, for x sampled from Π_{β_i}, the quantity

$$W_{\beta_i,\beta_{i+1}}(x) = e^{(\beta_i-\beta_{i+1})H(x)} \tag{5}$$

has expectation value

$$\mathbb{E}_{\Pi_{\beta_i}}[W_{\beta_i,\beta_{i+1}}] = \frac{Z(\beta_{i+1})}{Z(\beta_i)}, \tag{6}$$

so we can calculate $Z(\infty)$ as the telescoping product

$$Z(\infty) = Z(0)\frac{Z(\beta_1)}{Z(0)}\frac{Z(\beta_2)}{Z(\beta_1)}\cdots\frac{Z(\infty)}{Z(\beta_{\ell-1})} \tag{7}$$

by sampling $W_{\beta_i,\beta_{i+1}}$ at each successive temperature. In the SVV algorithm the temperature schedule is determined adaptively using properties of $\log Z(\beta)$ like convexity, so that, letting $|\Omega| = Z(0)$, the schedule has length $\ell = O(\sqrt{\log|\Omega|}\log n \log\log|\Omega|) = \tilde{O}(\sqrt{\log|\Omega|})$, a quadratic improvement over the best possible non-adaptive schedule length of $O(\log|\Omega|\log n) = \tilde{O}(\log|\Omega|)$. Recall that $n = \max_x H(x)$. We write $\tilde{O}(f)$ to suppress terms that are polylog in f, and in doing so, we assume that $\log|\Omega|$ and n are polynomially related. Our results do not otherwise assume any relation between $|\Omega|$ and n.

Such techniques could also be applied to the problem of Bayesian inference. Bayesian inference refers to an important paradigm in machine learning where values are assigned to model parameters according to a probability distribution that is updated using the observed data; this then allows us to quantify our uncertainty in the model parameters, as well as to update this uncertainty. Given model parameters θ that we wish to learn, we generally start with a prior distribution $\Pi_0(\theta)$ over the possible values that θ can take, and then given data points $\{x_i\}$ we update our prior distribution to obtain a posterior distribution over θ according to Bayes' rule:

$$p(\theta|\{x_i\}) = \frac{\Pi_0(\theta)\prod_i p(x_i|\theta)}{\sum_\theta \Pi_0(\theta)\prod_i p(x_i|\theta)}. \tag{8}$$

Here the normalization, the partition function $Z = \sum_\theta \Pi_0(\theta)\prod_i p(x_i|\theta)$, is often difficult to compute directly due to the sheer size of the parameter space. In analogy to the counting problem, where β parametrizes the partition function from the easy case of $Z(0)$ to the hard case of $Z(\infty)$, for Bayesian inference we will define the partition function

$$Z(\beta) = \sum_\theta \Pi_0(\theta)e^{-\beta L(\theta)}, \tag{9}$$

where the Hamiltonian corresponds to $L(\theta)$, the negative log-likelihood function, defined as

$$L(\theta) = -\log\left(\prod_i p(x_i|\theta)\right). \tag{10}$$

Then, in analogy to the counting problem, $Z(0)$ is easy to calculate as it just corresponds to $\sum_\theta \Pi_0(\theta) = 1$, while $Z(1)$, corresponding to the full posterior distribution, is hard to compute. As in the counting problem, we can imagine establishing a temperature schedule $\beta_0, \beta_1, \ldots, \beta_\ell$ with $\beta_0 = 0$ and $\beta_\ell = 1$. Then the Gibbs distribution at each temperature is given by

$$\Pi_{\beta_i}(\theta) = \frac{\Pi_0(\theta)e^{-\beta_i L(\theta)}}{Z(\beta_i)}. \tag{11}$$

Note, however, that in the case of Bayesian inference we don't need to compute the actual value of the partition function $Z(1)$ since we're ultimately interested in sampling from the posterior distribution. That is, it's enough to just return a sample from the last Markov chain. Thus we can in fact think of our Bayesian inference algorithm as performing simulated annealing using the adaptive cooling schedule as an annealing schedule. Because of the similarities between the counting problem and the Bayesian inference problem, we claim that we can modify the arguments of SVV to show that there exists a temperature schedule for Bayesian inference of length $\ell = \tilde{O}(\sqrt{\log(1/Z(1))}) = \tilde{O}(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)]})$. This schedule is quadratically shorter than the non-adaptive annealing schedule obtained in QSA papers such as those of [23, 24, 31], where the best result due to [31] uses inverse temperatures separated by a constant $\Delta\beta = O(1/\|H\|)$ so that $\ell = O(\|H\|) = O(\max_\theta L(\theta))$. Additionally, the dependence on $(1/Z(1))$ is exponentially better than the $O(1/\sqrt{Z(1)})$ dependence per Bayesian update in algorithms based on quantum rejection sampling, like that of [30]; however, this advantage is partially offset by a new dependence on the gap δ.

Merely the existence of such a short temperature schedule is not quite enough. In the next section we will demonstrate a quantum algorithm for efficiently finding temperature schedules of this length.

1.2.2 Quantizing Adaptive Annealing

So far our claims, that Bayesian inference can be treated as a simulated annealing problem analogous to counting problems, and that the annealing schedule can be made quadratically shorter, have been claims that would apply equally to both classical and quantum settings. We additionally claim that the computation of cooling schedules can be fully quantized. Combined with the QSA algorithm of [31], which performs quantum annealing given a cooling schedule, this means that it is possible to fully quantize both the algorithm for computing partition functions in the counting problem, and the algorithm for qsampling from the posterior distribution in Bayesian inference. To prove this claim, we combine techniques from [18] with a nondestructive version of amplitude estimation.

Montanaro's [18] algorithm for summing partition functions partially quantizes the SVV algorithm; the adaptive temperature schedule itself is still computed classically according to the SVV algorithm, but once given an inverse temperature, the algorithm specifies how to quantum sample at that temperature, as well as how to efficiently compute expectation values using those samples. Qsampling is performed according to the QSA algorithm of Wocjan and Abeyesinghe [31], who showed that given a sequence of ℓ slow-varying Markov chains (i.e., the overlap between successive stationary distributions is lower-bounded by some constant), each with spectral gap at least δ, an approximation to the stationary distribution of the final Markov chain can be obtained with $\tilde{O}(\ell/\sqrt{\delta})$ Markov chain steps, whereas classically the dependence on ℓ and δ would be $O(\ell/\delta)$. Given these quantum samples, Montanaro's algorithm then estimates expectation values using an amplitude estimation based algorithm that requires quadratically fewer samples than would be necessary classically. Overall Montanaro shows that it is possible to estimate the partition function with up to ϵ multiplicative error using $\tilde{O}(\log|\Omega|/(\sqrt{\delta}\epsilon) + \log|\Omega|/\delta)$ Markov chain steps, and notes that this complexity could be improved to $\tilde{O}(\log|\Omega|/(\sqrt{\delta}\epsilon))$ were it were possible to compute the cooling schedule itself via quantum means. The fact that the SVV algorithm uses a nonadaptive temperature schedule as a "warm start" for the adaptive schedule (which allows for a faster mixing time) is cited as an obstacle to quantizing the computation of the cooling schedule. We claim that these obstacles can be overcome.

As in Montanaro's algorithm, we can use the algorithm of Wocjan and Abeyesinghe to sample from the Gibbs distribution at each temperature. Additionally, we will also quantize the actual process of computing the cooling schedule itself. Our algorithm works as follows: since we are guaranteed the existence of the adaptive cooling schedule, we can binary search to find the next temperature. For each binary search candidate we can use amplitude estimation to calculate the overlap between the candidate state and the current state, which allows us to check whether the slow-varying condition is satisfied. Note that amplitude estimation only requires that we be able to reflect over the candidate state, and that the quantum walk operator provides such a reflection operator. We also observe that all quantum measurements occur only during the amplitude estimation step, and that amplitude estimation can be made non-destructive so that it's possible to restore the post-measurement state to the pre-measurement state at almost no additional cost. Finally we also claim that in the quantum case, the slow-varying condition itself is enough to ensure warm-start mixing times, which ends up simplifying one of the steps in the SVV algorithm.

Putting these claims together yields the results of Theorem 1 and Table 1.

2 Existence of Cooling Schedule

In this section we slightly modify an argument of SVV [25] to show that there exists a cooling schedule of bounded length for a partition function of the form (1), which encompasses both (3), corresponding to the counting problem, and (9), corresponding to Bayesian inference. Furthermore, this cooling schedule satisfies the Chebyshev condition in the case of counting problems, and the slow-varying condition in the case of Bayesian inference.

As noted in the previous section, the SVV algorithm generates a sequence of inverse temperatures $\beta_0, \beta_1, \ldots, \beta_\ell$ with $\beta_0 = 0$ and $\beta_\ell = \infty$; then, given such a schedule, the idea is to sample from the Gibbs distribution at each temperature in order to compute the quantities $W_{\beta_i,\beta_{i+1}}$, whose expectation value is the ratio of Z at successive temperatures. Taking the telescoping product of these ratios according to equation (7) then allows us to estimate $Z(\infty)$ starting from $Z(0)$.

In a B-Chebyshev cooling schedule such as that generated by SVV, we have the additional requirement that the variance of $W_{\beta_i,\beta_{i+1}}$ is bounded; that is, that

$$\frac{\mathbb{E}\left(W_{\beta_i,\beta_{i+1}}^2\right)}{\mathbb{E}\left(W_{\beta_i,\beta_{i+1}}\right)^2} = \frac{Z(2\beta_{i+1}-\beta_i)Z(\beta_i)}{Z(\beta_{i+1})^2} \leq B \qquad (12)$$

for a constant B. This additional bounded variance requirement then guarantees that the product of expectation values $\mathbb{E}[W_{\beta_0,\beta_1}]\mathbb{E}[W_{\beta_1,\beta_2}]\ldots\mathbb{E}[W_{\beta_{\ell-1},\beta_\ell}]$ will be a good approximation to the product $W_{\beta_0,\beta_1}W_{\beta_1,\beta_2}\cdots W_{\beta_{\ell-1},\beta_\ell}$ within a bounded number of samples.

In the case of Bayesian inference we're not actually trying to calculate the partition function $Z(1)$ (instead we want to sample from the Gibbs distribution at $\beta = 1$), so it might seem like we don't need the additional bounded variance condition. However, the *slow-varying condition* is another property, closely related to bounded variance, which we will need. The slow-varying condition states that $|\langle\pi_{\beta_i}|\pi_{\beta_{i+1}}\rangle|^2 \geq$

$1/B$, since

$$\langle \pi_{\beta_i} | \pi_{\beta_{i+1}} \rangle = \frac{Z\left(\frac{\beta_i+\beta_{i+1}}{2}\right)}{\sqrt{Z(\beta_i)}\sqrt{Z(\beta_{i+1})}}.$$

Then the slow-varying condition can be rewritten as

$$\frac{Z(\beta_i)Z(\beta_{i+1})}{Z\left(\frac{\beta_i+\beta_{i+1}}{2}\right)^2} \le B. \tag{13}$$

We define $f(\beta) = \log Z(\beta)$ to help understand the slow-varying and Chebyshev conditions. Note that f is convex. Observe that when we set $B = e^2$, both the slow-varying condition (13) and the Chebyshev condition (12) can be rewritten in the form

$$f\left(\frac{\gamma_i+\gamma_{i+1}}{2}\right) \ge \frac{f(\gamma_i)+f(\gamma_{i+1})}{2} - 1, \tag{14}$$

where for the Chebyshev condition $\gamma_i = \beta_i$ and $\gamma_{i+1} = 2\beta_{i+1} - \beta_i$, while for the slow-varying condition $\gamma_i = \beta_i$ and $\gamma_{i+1} = \beta_{i+1}$. Equation (14) should be compared with the inequality $f(\frac{\gamma_i+\gamma_{i+1}}{2}) \le \frac{f(\gamma_i)+f(\gamma_{i+1})}{2}$ resulting from convexity of f.

The existence of Chebyshev and slow-varying sequences is then expressed by the following lemma, which guarantees the existence of a sequence of inverse temperatures satisfying equation (14). We will slightly modify the original bound that appears in SVV, from $\ell \le \sqrt{(f(0)-f(1))\log(f'(0)/f'(\gamma))}$ to $\ell \le \sqrt{(f(0)-f(1))\log(f'(0)/(f'(\gamma)+1))}$, in order for this bound to work in the case of Bayesian inference. The full proof of the lemma appears in Appendix A.

Lemma 2. *(Modified from SVV [25] Lemma 4.3, Appendix A) For f a convex function over domain $[0,\gamma]$, there exists a sequence $\gamma_0 < \gamma_1 < \ldots < \gamma_\ell$ with $\gamma_0 = 0$ and $\gamma_\ell = \gamma$ satisfying*

$$f\left(\frac{\gamma_i+\gamma_{i+1}}{2}\right) \ge \frac{f(\gamma_i)+f(\gamma_{i+1})}{2} - 1 \tag{15}$$

with length

$$\ell \le \sqrt{(f(0)-f(\gamma))\log\left(\frac{f'(0)}{f'(\gamma)+1}\right)}. \tag{16}$$

This suggests that we can construct a Chebyshev cooling schedule greedily; given left endpoint γ_i, choose the next endpoint by finding the largest possible right endpoint γ_{i+1} so that the midpoint satisfies equation (14), and Lemma 2 then guarantees an upper bound on the length of a schedule constructed in this manner.

In the next section we will describe a quantum algorithm for efficiently carrying out a version of this procedure. In the remainder of this section we show that the length of the schedule generated by this algorithm, both in the case of the counting problem and in the case of Bayesian inference, is quadratically shorter than the length of the corresponding nonadaptive schedule.

In the case of the counting problem, SVV show that the schedule derived from (16) ends up being $\tilde{O}(\sqrt{\log|\Omega|})$, where typically $\log|\Omega| \sim \mathrm{poly}(n)$ for $n = \max_x H(x)$:

Theorem 3. *(SVV [25] Theorem 4.1) For $Z(\beta)$ a partition function of the form given by equation (3), letting $|\Omega| = Z(0)$ and assuming $Z(\infty) \ge 1$, there exists a B-Chebyshev cooling schedule with $B = e^2$, $\beta_0 = 0$, and $\beta_\ell = \infty$, of length*

$$O(\log\log|\Omega|\sqrt{\log|\Omega|\log(n)}) = \tilde{O}(\sqrt{\log|\Omega|}).$$

The full proof of Theorem (3) can be found in [25], but the idea is the following. For counting problems, where we need to anneal all the way to $\beta_\ell = \infty$, it's enough to take $\beta_{\ell-1} = \gamma$ with γ the inverse temperature satisfying $f(\gamma) = 1$. This choice of γ guarantees that eq. (14) is satisfied between $\beta_\ell = \infty$ and $\beta_{\ell-1} = \gamma$. Next we use Lemma (2) to note that there exists a sequence of $\gamma_{0'}, \gamma_{1'}, \ldots, \gamma_{\ell'}$ with $\gamma_{0'} = \gamma_0 = 0$ and $\gamma_{\ell'} = \gamma$ that satisfy (14) with

$$\ell' \le \sqrt{\log|\Omega|\log(n)}. \tag{17}$$

We can see that the expression for ℓ' comes from (16) with $\log|\Omega|$ corresponding to the $f(0)-f(\gamma)$ term and $\log n$ corresponding to the $\log(f'(0)/(f'(\gamma)+1))$ term. Next we need to extract the $\beta_0, \ldots, \beta_{\ell-1}$ from the $\gamma_{0'}, \ldots, \gamma_{\ell'}$, where $\beta_0 = \gamma_0 = 0$ and $\beta_{\ell-1} = \gamma_{\ell'} = \gamma$. SVV show that it suffices to insert additional inverse temperatures in each interval $[\gamma_i, \gamma_{i+1}]$ in the following way:

$$\begin{aligned} &\gamma_i, \gamma_i + (1/2)(\gamma_{i+1}-\gamma_i), \gamma_i + (3/4)(\gamma_{i+1}-\gamma_i), \\ &\quad \gamma_i + (7/8)(\gamma_{i+1}-\gamma_i), \ldots, \\ &\quad \gamma_i + (1-2^{-\lceil \log\log|\Omega| \rceil})(\gamma_{i+1}-\gamma_i), \gamma_{i+1}, \end{aligned} \tag{18}$$

which ensures that each pair of adjacent temperatures satisfies the Chebyshev condition (12). This adds an additional factor of $\log\log|\Omega|$, so the dominant term is still $\sqrt{\log|\Omega|}$. SVV also show that any nonadaptive schedule must be $\tilde{\Omega}(\log|\Omega|)$, so the adaptive schedule is quadratically shorter.

In the case of Bayesian inference, we claim that we have a similar result, where the adaptive schedule has length $\ell = \tilde{O}(\sqrt{\log(1/Z(1))}) = \tilde{O}(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)]})$.

Here the argument is more straightforward because we can directly take the $\gamma_0, \ldots, \gamma_\ell$ from Lemma (2) to be the inverse temperatures $\beta_0, \ldots, \beta_\ell$.

Theorem 4. *For partition function $Z(\beta)$ of the form given by equation (9), there exists a temperature schedule with $B = e^2$, $\beta_0 = 0$, and $\beta_\ell = 1$, satisfying $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle|^2 \geq 1/B$, of length*

$$O\left(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)] \log(\mathbb{E}_{\Pi_0}[L(\theta)])}\right) = \tilde{O}\left(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)]}\right). \tag{19}$$

Proof of Theorem 4. We use the result of Lemma 2 with $\gamma_\ell = 1$. Plugging into the expression for the length of the cooling schedule from equation (16), we note that $f(0) = 0$ and $f(1) = \log Z(1)$ so that $f(0) - f(1) = \log(1/Z(1)) = -\log Z(1)$. Note that this can be rewritten as

$$\begin{aligned} -\log Z(1) &= -\log\left(\sum_\theta \Pi_0(\theta) e^{-L(\theta)}\right) \\ &= -\log\left(\mathbb{E}_{\Pi_0(\theta)}\left[e^{-L(\theta)}\right]\right). \end{aligned} \tag{20}$$

By Jensen's inequality, $-\log(\mathbb{E}[X]) \leq -\mathbb{E}[\log(X)]$, so

$$\log(1/Z(1)) \leq \mathbb{E}_{\Pi_0}[L(\theta)]. \tag{21}$$

We also note that $f'(0) = \mathbb{E}_{\Pi_0(\theta)}[L(\theta)]$ and $f'(1) = \mathbb{E}_{\Pi_1(\theta)}[L(\theta)]$, where $\Pi_0(\theta)$ denotes the prior distribution and $\Pi_1(\theta)$ denotes the posterior distribution, so that $\log(f'(0)/(f'(1)+1)) \leq \log(\mathbb{E}_{\Pi_0}[L(\theta)])$. Putting everything together,

$$\begin{aligned} \ell &= O\left(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)] \log(\mathbb{E}_{\Pi_0}[L(\theta)])}\right) \\ &= \tilde{O}\left(\sqrt{\mathbb{E}_{\Pi_0(\theta)}[L(\theta)]}\right). \end{aligned} \tag{22}$$

□

The length of the adaptive cooling schedule is quadratically shorter than the length of nonadaptive annealing schedules currently employed by QSA algorithms such as those [23, 24, 31]. For example, in the best result due to [31], which employs slow-varying Markov chains to perform QSA on a sequence of Markov chains with stationary distributions given by

$$\Pi_\beta(x) = \frac{e^{-\beta H(x)}}{Z(\beta)}, \tag{23}$$

taking the inverse temperatures to be separated by a constant $\Delta\beta = 1/\|H\|$ ensures that the slow-varying condition is preserved. Applying this to Bayesian inference, where we have $x = \theta$ with $\theta \sim \Pi_0(\theta)$, $H(\theta) = L(\theta)$, and β that we anneal between 0 and 1, we end up with a nonadaptive schedule of length $O(\max_\theta L(\theta))$.

3 Construction of Cooling Schedule and Quantum Algorithm Details

We now give a quantum algorithm that adaptively constructs the cooling schedule from the previous section. As it does so, it simultaneously produces the quantum state corresponding to the Gibbs distribution at the current inverse temperature in the schedule construction process. In the case of Bayesian inference, obtaining the state at the final inverse temperature corresponds to qsampling from the posterior distribution. For the counting problem, sampling at each inverse temperature allows us to estimate the telescoping product (see equation (7)) corresponding to the partition function $Z(\infty)$.

To do so we will need the following result of Wocjan and Abeyesinghe [31], as restated by Montanaro [18], which shows that it is possible to quantum sample given access to a sequence of slow-varying Markov chains:

Theorem 5. *(Wocjan and Abeyesinghe [31], restated as Montanaro [18] Theorem 9) Assume that we have classical Markov chains $M_0, \ldots, M_\ell$ with stationary distributions $\Pi_0, \ldots, \Pi_\ell$ that are slow-varying; that is to say, they satisfy $|\langle \Pi_i | \Pi_{i+1} \rangle|^2 \geq p$ for all $i = 0, .., \ell - 1$. Let δ lower bound the spectral gaps of the Markov chains, and assume that we can prepare the starting state $|\Pi_0\rangle$. Then, for any $\epsilon > 0$, there is a quantum algorithm that produces a quantum state that is ϵ-close to $|\Pi_\ell\rangle$ and uses*

$$O\left(\ell \sqrt{\delta^{-1}} \log^2(\ell/\epsilon)(1/p)\log(1/p)\right)$$

total steps of the quantum walk operators W_i corresponding to the Markov chains M_i.

As we stated in the previous section, satisfying the slow-varying condition takes the same form as satisfying the Chebyshev condition for a cooling schedule. We can also easily prepare the starting state $|\Pi_0\rangle$ using a result of [33, 7, 13], who showed that it is possible to efficiently create the coherent encoding

$$\sum_i \sqrt{p_i}\,|i\rangle$$

of the discretized version $\{p_i\}$ of a probability distribution $p(x)$, provided that $p(x)$ can be efficiently integrated classically (for example, by Monte Carlo methods). For counting problems, $|\Pi_0\rangle$ is just the uniform distribution, which can be easily integrated. For Bayesian inference we make a choice of prior that

can be integrated classically, allowing us to easily prepare $|\Pi_0\rangle$. A recent review of other state-preparation methods can be found in [27]; see also [1].

Now we describe how to proceed to the next state $|\Pi_{\beta_{i+1}}\rangle$ assuming that we already have the state $|\Pi_{\beta_i}\rangle$. According to the procedure described in the previous section, we'd like to find the largest β_{i+1} so that $|\langle\Pi_{\beta_{i+1}}|\Pi_{\beta_i}\rangle|^2 \geq p$ in the case of Bayesian inference, and $|\langle\Pi_{2\beta_{i+1}-\beta_i}|\Pi_{\beta_i}\rangle|^2 \geq p$ for counting problems. To do so we will binary search for β_{i+1} in the Bayesian case, and $2\beta_{i+1}-\beta_i$ in the counting problem case, computing the overlap for the state $|\Pi_{\beta'}\rangle$ corresponding to each candidate inverse temperature β' to see if $|\langle\Pi_{\beta'}|\Pi_{\beta_i}\rangle|^2 \geq p$ is satisfied. Note that we can't actually produce each state $|\Pi_{\beta'}\rangle$ since being able to anneal to this state would require that it already satisfy the slow-varying condition. Luckily, being able to reflect about $|\Pi_{\beta'}\rangle$ suffices, and quantum walks will give us the ability to perform this reflection. When we estimate the overlap we also need to make sure that the state $|\Pi_{\beta_i}\rangle$ is not destroyed, and we ensure this by computing the overlap between $|\Pi_{\beta_i}\rangle$ and $|\Pi_{\beta'}\rangle$ using a form of amplitude estimation that has been made nondestructive. This will be doubly useful in the case of counting problems, where to calculate $Z(\infty)$ we will need to estimate expectation values $\mathbb{E}[W_{\beta_i,\beta_{i+1}}]$ at intermediate temperatures without destroying the corresponding state, which we then continue annealing to the next temperature. In Section 4 we describe the amplitude estimation algorithm of Brassard, Hoyer, Mosca, and Tapp (BHMT) [5] and demonstrate how the starting state can be restored at almost no additional cost in the number of Markov chain steps required. The nondestructive amplitude estimation algorithm can be summarized as follows:

Theorem 6 (Nondestructive amplitude estimation)**.** *Given state $|\psi\rangle$ and reflections $R_\psi = 2|\psi\rangle\langle\psi| - I$ and $R = 2P - I$, and any $\eta > 0$, there exists a quantum algorithm that outputs $\tilde{a}$, an approximation to $a = \langle\psi|P|\psi\rangle$, so that*

$$|\tilde{a} - a| \leq 2\pi\frac{a(1-a)}{M} + \frac{\pi^2}{M^2}$$

with probability at least $1-\eta$ and $O(\log(1/\eta)M)$ uses of R_ψ and R. Moreover the algorithm restores the state $|\psi\rangle$ with probability at least $1-\eta$.

This is proved in Section 4.

To perform amplitude estimation we will need to be able to perform the reflections $R_\psi = 2|\Pi_{\beta_i}\rangle\langle\Pi_{\beta_i}| - I$ and $R = 2P - I = 2|\Pi_{\beta'}\rangle\langle\Pi_{\beta'}| - I$. The following theorem due to Magniez, Nayak, Roland, and Santha (MNRS) [15] allows us to approximate these reflections.

Theorem 7. *(MNRS [15] Theorem 6) Suppose that we wish to approximate the reflection $R = 2|\Pi\rangle\langle\Pi| - I$ about $|\Pi\rangle$, where $|\Pi\rangle$ is the coherent encoding of Π, the stationary distribution of Markov chain M with spectral gap δ. Then there is a quantum circuit $\tilde{R}$ so that for $|\Psi\rangle$ orthogonal to $|\Pi\rangle$, $\|(\tilde{R} + I)|\Psi\rangle\| \leq 2^{1-k}$, and $\tilde{R}$ uses $O(k/\sqrt{\delta})$ steps of the quantum walk operator W corresponding to M.*

In the amplitude estimation algorithm we need to be able to perform $O(\log(1/\eta)M)$ applications of the Grover search operator $Q = -R_\psi R$ (see Section 4 for more details), but instead we have access to an approximation $\tilde{Q} = -\tilde{R}_\psi\tilde{R}$. We claim that this error can be bounded using the following observation.

Lemma 8. *Let $\tilde{R}_\psi$ and $\tilde{R}$ be the respective approximations to $R_\psi = 2|\psi\rangle\langle\psi| - I$ and $R = 2P - I$ given by the algorithm of Theorem 7. Then, letting $Q = -R_\psi R$ with approximation $\tilde{Q} = -\tilde{R}_\psi\tilde{R}$, and letting state $|\psi'\rangle \in \mathrm{span}\{|\psi\rangle, \mathrm{Im}(P)\}$, the error in using approximate reflections can be bounded as $\|Q^i|\psi'\rangle - \tilde{Q}^i|\psi'\rangle\| \leq i2^{2-k}$.*

Proof of Lemma 8. By induction. The $i = 1$ case follows from Theorem 7. Assume $\|Q^{i-1}|\psi'\rangle - \tilde{Q}^{i-1}|\psi'\rangle\| \leq (i-1)2^{2-k}$. Then $\|Q^i|\psi'\rangle - \tilde{Q}^i|\psi'\rangle\| \leq \|Q^{i-1}|\psi'\rangle - \tilde{Q}^{i-1}|\psi'\rangle\| + \|(Q - \tilde{Q})Q^i|\psi'\rangle\| \leq i2^{2-k}$. □

Using Lemma 8, we can then restate the result on nondestructive amplitude estimation using approximate reflections.

Theorem 9. *(Nondestructive amplitude estimation using approximate reflections) Given state $|\psi\rangle$, an approximation $\tilde{R}_\psi$ to reflection $R_\psi = 2|\psi\rangle\langle\psi| - I$, an approximation $\tilde{R}$ to reflection $R = 2P - I$, and any $\eta > 0$, where all approximate reflections are given by Theorem 7, there exists a quantum algorithm that outputs $\tilde{a}$, an approximation to $a = \langle\psi|P|\psi\rangle$, so that*

$$|\tilde{a} - a| \leq 2\pi a(1-a)\epsilon + \pi^2\epsilon^2$$

with probability at least $1-\eta$. The algorithm restores the state $|\psi\rangle$ with probability at least $1-\eta$ and requires $O(1/(\epsilon\sqrt{\delta})\log(1/\epsilon)\log(1/\eta))$ steps of the quantum walk operators corresponding to $\tilde{R}_\psi$ and $\tilde{R}$, where δ lower bounds the spectral gaps of the corresponding Markov chains.

Proof of Theorem 9. The algorithm for nondestructive amplitude estimation (see Theorem 6 and Section 4) requires the ability to generate the state $Q^M|\psi\rangle$. By Lemma 8 we know that we can generate an approximation $\tilde{Q}^M|\psi\rangle$ with $\|Q^M|\psi\rangle - \tilde{Q}^M|\psi\rangle\| \leq M2^{2-k}$. Taking $k = \log M + c$ then ensures that

this error is bounded by a constant. Finally, calling $\epsilon = 1/M$, we note that amplitude estimation occurs with error $O(\epsilon)$ if we require $O(1/\epsilon \log(1/\eta))$ uses of $\tilde{R}_\psi$ and $\tilde{R}$. With our choice of k, each use of $\tilde{R}_\psi$ and $\tilde{R}$ requires $O(\log(1/\epsilon)/\sqrt{\delta})$ Markov chain steps, so the algorithm requires $O(1/(\epsilon\sqrt{\delta}) \log(1/\epsilon) \log(1/\eta))$ total Markov chain steps. □

Having described everything we need—the QSA algorithm, the binary search, and nondestructive amplitude estimation—we will now put everything together. The result will be two fully quantum algorithms, one for constructing an adaptive schedule and qsampling from the posterior distribution for Bayesian inference, and another for constructing an adaptive schedule and calculating $Z(\infty)$ for counting problems.

3.1 QSA for Bayesian Inference

The quantum algorithm for Bayesian inference is given by the following.

Algorithm 1 QSA for Bayesian inference.

Input: State $|\Pi_0\rangle = \sum_x \sqrt{\Pi_0}\,|x\rangle$, the coherent encoding of the prior distribution, constant $p > 0$, and constant $\eta > 0$.

Output: State $|\widetilde{\Pi_1}\rangle$, an approximation to the coherent encoding of the posterior distribution, and temperature schedule $\beta_0, \beta_1, \ldots, \beta_\ell$ with $\beta_0 = 0$ and $\beta_\ell = 1$ so that $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle| \geq p$.

1: **for** i:=1 to $\ell = O\left(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)] \log(\mathbb{E}_{\Pi_0}[L(\theta)])}\right)$ **do**
2: At current inverse temperature β_i with state $|\Pi_{\beta_i}\rangle$,
3: **repeat**
4: Binary search on $\beta' \in [\beta_i, 1]$ with precision $1/(\max_\theta L(\theta))$.
5: Perform nondestructive amplitude estimation to calculate $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2$ with error $\epsilon_e = p/10$ and failure probability $\eta/(\ell \max_\theta L(\theta))$.
6: **until** $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2 \geq p$.
7: Anneal from $|\Pi_{\beta_i}\rangle$ to $|\Pi_{\beta_{i+1}}\rangle$ at inverse temperature $\beta_{i+1} = \beta'$.
8: **end for**
9: Return $|\Pi_{\beta_\ell}\rangle$.

For simplicity the above algorithm refers in each case to the ideal state, e.g. we write "Return $|\Pi_{\beta_\ell}\rangle$" to mean that we return the state which approximates $|\Pi_{\beta_\ell}\rangle$.

Theorem 10 (Quantum adaptive annealing algorithm for Bayesian inference)**.** *Assume that we are given a prior distribution $\Pi_0(\theta)$ and a likelihood function $L(\theta)$, so that we can parametrize the partition function $Z(\beta) = \sum_\theta \Pi_0(\theta) e^{-\beta L(\theta)}$ at each inverse temperature $\beta \in [0, 1]$. Assume that we can generate the state $|\Pi_0\rangle$ corresponding to the coherent encoding of the prior, and assume that for every inverse temperature β we have a Markov chain M_β with stationary distribution Π_β and spectral gap lower-bounded by δ. Then, for any $\epsilon > 0$, $\eta > 0$, there is a quantum algorithm that, with probability at least $1 - \eta$, produces state $|\widetilde{\Pi_1}\rangle$ so that $\| |\widetilde{\Pi_1}\rangle - |\Pi_1\rangle \| \leq \epsilon$ for $|\Pi_1\rangle$ the coherent encoding of the posterior distribution $\Pi_1(\theta) = \Pi_0(\theta) e^{-L(\theta)}/Z(1)$. The algorithm uses*

$$O\Big(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)] \log(\mathbb{E}_{\Pi_0}[L(\theta)])} \log^2(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)] \log(\mathbb{E}_{\Pi_0}[L(\theta)])}/(\epsilon\sqrt{\delta})) \log(\max_\theta L(\theta)) \log(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)] \log(\mathbb{E}_{\Pi_0}[L(\theta)])} \max_\theta L(\theta)/\eta)\Big) = \tilde{O}(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)]/\delta})$$

total steps of the quantum walk operators corresponding to the Markov chains M_β.

Proof of Theorem 10. From Theorem 4 we know that the annealing schedule has length

$$\ell = O\left(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)] \log(\mathbb{E}_{\Pi_0}[L(\theta)])}\right).$$

From Theorem 5 we know that, given a sequence of ℓ inverse temperatures $\{\beta_i\}$ with stationary distributions that satisfy $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle|^2 \geq p$ for a constant $p > 0$, quantum annealing to the state at the final temperature β_ℓ takes

$$O(\ell \delta^{-1/2} \log^2(\ell/\epsilon)(1/p) \log(1/p))$$

total steps of the quantum walk operators corresponding to the M_{β_i}.

Since we simultaneously construct the schedule and anneal our state on the fly, we also need to account for the cost of constructing the schedule. At each inverse temperature β_i we perform binary search to find inverse temperature β_{i+1} satisfying $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle|^2 \geq p$ in the interval $[\beta_i, 1]$. We choose binary search precision $1/(\max_\theta L(\theta))$ since $\Pi_\beta \propto e^{-\beta L(\theta)}$, which means that the binary search procedure contributes a factor of $\log(\max_\theta L(\theta))$ to the complexity. For each candidate inverse temperature β' in the binary search, we perform nondestructive amplitude estimation to calculate $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2$. We set the failure probability of nondestructive amplitude estimation

to $\eta/(\ell \max_\theta L(\theta))$. From Theorem 9, we can estimate $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2 \geq p$ with error that is $O(\epsilon_e)$ using $O(1/(\epsilon_e \sqrt{\delta}) \log(1/\epsilon_e) \log(\ell \max_\theta L(\theta)/\eta))$ Markov chain steps. Since we take $\epsilon_e = p/10$, our binary search then guarantees that we can find a sequence of ℓ inverse temperatures satisfying $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle|^2 \geq 9p/10$ with a total cost of

$$O(\ell \delta^{-1/2} \log(\max_\theta L(\theta))(1/p) \log(1/p) \log(\ell(\max_\theta L(\theta))/\eta))$$

total Markov chain steps. Adding the two contributions from constructing the schedule and annealing the state, we get a total cost of

$$O(\ell \delta^{-1/2} \log(\max_\theta L(\theta)) \log^2(\ell/\epsilon)(1/p) \log(1/p)$$
$$\log(\ell(\max_\theta L(\theta))/\eta)) = \tilde{O}(\sqrt{\mathbb{E}_{\Pi_0}[L(\theta)]/\delta})$$

Markov chain steps. □

3.2 QSA for Counting Problems

In counting problems, we'd like to calculate $Z(\infty)$ according to the telescoping product given by (7), which means that we need to sample and estimate an expectation value $\mathbb{E}[W_{\beta_i,\beta_{i+1}}]$ at each inverse temperature β_i, where $W_{\beta_i,\beta_{i+1}}$ is given by equation (6). Computing the expectation value can be done using the amplitude estimation based algorithm of Montanaro, and moreover it can be made nondestructive using nondestructive amplitude estimation. This is Algorithm 4 of [18], which estimates an expectation value $\mathbb{E}(X)$ assuming bounded variance $\mathrm{Var}(X)/(\mathbb{E}(X))^2 \leq B$. Note that the bounded variance condition of Algorithm 4 is satisfied using the Chebyshev condition of Equation 12.

Thus we would expect a cost both in terms of the number of Markov chain steps required and in terms of the number of samples required, where the sample cost is incurred by the calculation of the expectation values, while the Markov chain cost is incurred both in the computation of the temperature schedule itself, and in the calculation of expectation values given the inverse temperatures.

The quantum algorithm for approximating $Z(\infty)$ is as follows. Note that in Line 5 we perform nondestructive amplitude estimation to determine the next temperature in the schedule, while in Lines 10, 13, and 16 we perform nondestructive amplitude estimation using Algorithm 4 of [18] to estimate the $\mathbb{E}[W_{\beta_i,\beta_{i+1}}]$, which we then multiply together at the end to obtain an estimate for the partition function. The proof of correctness of this algorithm is in Theorem 14.

Algorithm 2 QSA for computing partition functions for counting problems.

Input: Descriptions of state space Ω and energy function $H : \Omega \mapsto \mathbb{R}_+$. Constant $B > 0$, bound $n \geq \max_x H(x)$, error $\epsilon = O(1/\sqrt{\log\log|\Omega|})$, failure probability $\eta > 0$, and $\tilde{O}(B\sqrt{\log|\Omega|}/\epsilon)$ copies of state $|\Pi_0\rangle := |\Omega|^{-1/2} \sum_{x\in\Omega} |x\rangle$.

Output: $\tilde{Z}$, an ϵ-approximation to $Z(\infty)$, and B-Chebyshev cooling schedule $\beta_0 = 0, \beta_1, \ldots, \beta_\ell = \infty$ satisfying $\log Z(\beta_{\ell-1}) = 1$.

1: **for** i:=1 to $O(\sqrt{\log|\Omega| \log(n)})$ **do**
2: At current inverse temperature β_i with states $|\Pi_{\beta_i}\rangle$,
3: **repeat**
4: Binary search on $\beta' \in [\beta_i, \gamma]$ with precision $1/n$.
5: Perform nondestructive amplitude estimation to estimate $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2$ with error $\epsilon_e = p/10$ and failure probability $\eta/(n \log\log n \sqrt{\log|\Omega| \log(n)})$.
6: **until** our estimate satisfies $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2 \geq 1/B$.
7: Set $\beta_{i+m+1} = (\beta_i + \beta')/2$ for $m = \lceil \log\log|\Omega| \rceil$.
8: **for** j:=1 to $m = \lceil \log\log|\Omega| \rceil$ **do**
9: Set $\beta_{i+j} = \beta_i + (1 - 2^{-j})(\beta_{i+m+1} - \beta_i)$
10: Perform Algorithm 4 of [18] on states $|\Pi_{\beta_{i+j-1}}\rangle$ using nondestructive amplitude estimation with error $\epsilon_e = \epsilon$ and failure probability $\eta/(n \log\log n \sqrt{\log|\Omega| \log(n)})$ to estimate $\mathbb{E}[W_{\beta_{i+j-1},\beta_{i+j}}]$.
11: Anneal from states $|\Pi_{\beta_{i+j-1}}\rangle$ to states $|\Pi_{\beta_{i+j}}\rangle$.
12: **end for**
13: Perform Algorithm 4 of [18] on states $|\Pi_{\beta_{i+m}}\rangle$ using nondestructive amplitude estimation with error $\epsilon_e = \epsilon$ and failure probability $\eta/(n \log\log n \sqrt{\log|\Omega| \log(n)})$ to estimate $\mathbb{E}[W_{\beta_{i+m},\beta_{i+m+1}}]$.
14: Anneal from states $|\Pi_{\beta_{i+m}}\rangle$ to states $|\Pi_{\beta_{i+m+1}}\rangle$.
15: **end for**
16: Perform Algorithm 4 of [18] on states $|\Pi_\gamma\rangle$ using nondestructive amplitude estimation with error $\epsilon_e = \epsilon$ and failure probability $\eta/(n \log\log n \sqrt{\log|\Omega| \log(n)})$ to estimate $\mathbb{E}[W_{\gamma,\infty}]$.
17: Return $\tilde{Z} = \prod_{i=0}^{\ell-1} \mathbb{E}[W_{\beta_i,\beta_{i+1}}]$

As in Algorithm 1 we use notation that ignores the errors in our estimates. Specifically, in the last line we write $\mathbb{E}[W_{\beta_i,\beta_{i+1}}]$ to mean our estimates of this that we have computed in lines 10, 13, and 16. Likewise

we refer to various states $|\Pi_\beta\rangle$ while our algorithm actually has access to approximate versions of those states.

The following theorem due to Montanaro [18] specifies how many total qsamples are needed to calculate $Z(\infty)$.

Theorem 11 (Montanaro [18] Theorem 8)**.** *Given a counting problem partition function $Z(\beta)$ and a B-Chebyshev cooling schedule $\beta_0, \beta_1, \ldots, \beta_\ell$ with $\beta_0 = 0$ and $\beta_\ell = \infty$, and assuming the ability to qsample from each Gibbs distribution Π_{β_i}, there is a quantum algorithm which outputs an estimate $\tilde{Z}$ of $Z(\infty)$ such that*

$$\Pr\left[(1-\epsilon)Z(\infty) \leq \tilde{Z} \leq (1+\epsilon)Z(\infty)\right] \leq 3/4$$

using

$$O\left(\frac{B\ell \log \ell}{\epsilon} \log^{3/2}\left(\frac{B\ell}{\epsilon}\right) \log\log\left(\frac{B\ell}{\epsilon}\right)\right)$$

qsamples at each Π_{β_i}, which corresponds to

$$O\left(\frac{B\ell^2 \log \ell}{\epsilon} \log^{3/2}\left(\frac{B\ell}{\epsilon}\right) \log\log\left(\frac{B\ell}{\epsilon}\right)\right) = \tilde{O}(B\ell^2/\epsilon)$$

qsamples in total.

The cost in terms of quantum walk steps needed can be split up into two parts: the cost of computing the schedule itself (that is, determining the inverse temperatures and annealing through them), and the cost of computing the expectation values in order to estimate $Z(\infty)$ (that is, given each inverse temperature). The following theorem due to Montanaro [18] specifies the total Markov chain steps needed to estimate the expectation values given a temperature schedule.

Theorem 12. *(Montanaro [18] Theorem 11) Given a counting problem partition function $Z(\beta)$, a B-Chebyshev cooling schedule $\beta_0, \beta_1, \ldots, \beta_\ell$ with $\beta_0 = 0$ and $\beta_\ell = \infty$, and a series of Markov chains with stationary distributions Π_{β_i} and spectral gap lower bounded by δ, and assuming the ability to qsample from Π_0, for any $\eta > 0$ and $\epsilon = O(1/\sqrt{\log \ell})$ there exists a quantum algorithm which uses*

$$O((\ell^2/\sqrt{\delta}\epsilon) \log^{5/2}(\ell/\epsilon) \log(\ell/\eta) \log\log(\ell/\epsilon)) = \tilde{O}(\ell^2/\sqrt{\delta}\epsilon)$$

steps of the quantum walk operators corresponding to the Markov chains and outputs $\tilde{Z}$, an estimate of $Z(\infty)$ such that

$$\Pr\left[(1-\epsilon)Z(\infty) \leq \tilde{Z} \leq (1+\epsilon)Z(\infty)\right] \geq 1 - \eta.$$

We claim that analogous to the case of Bayesian inference, the construction of the schedule itself can be completed with $\tilde{O}(\sqrt{\ell/\delta})$ Markov chain steps:

Theorem 13. *Given the counting problem partition function $Z(\beta) = \sum_{k=0}^{n} a_k e^{-\beta k}$ with $|\Omega| = \sum_{k=0}^{n} a_k$ and $n = \max_x H(x)$, assume that we can generate state $|\Pi_0\rangle$ corresponding to the uniform distribution over Ω. Letting γ be the temperature at which $\log Z(\gamma) = 1$, assume also that for every inverse temperature $\beta \in [0, \gamma]$ we have a Markov chain M_β with stationary distribution Π_β and spectral gap lower-bounded by δ. Then, for any $\epsilon > 0$, $\eta > 0$, there is a quantum algorithm (Algorithm 2, lines 1–15) which anneals through the sequence of states $|\widetilde{\Pi_{\beta_i}}\rangle$ so that $\| |\widetilde{\Pi_{\beta_i}}\rangle - |\Pi_{\beta_i}\rangle \| \leq \epsilon$ for $|\Pi_{\beta_i}\rangle$ the coherent encoding of the Gibbs distribution at inverse temperatures β_i. The algorithm uses*

$$O\Big(\log\log|\Omega| \sqrt{\log|\Omega| \log(n)} \delta^{-1/2} \log^2(\sqrt{\log|\Omega| \log(n)}/\epsilon) \log n \log(n \log\log n \sqrt{\log|\Omega| \log(n)}/\eta)\Big) = \tilde{O}(\sqrt{(\log|\Omega|)/\delta})$$

total steps of the quantum walk operators corresponding to the Markov chains M_β.

Proof of Theorem 13. According to Theorem 3, the B-Chebyshev cooling schedule has length

$$\ell = O\left(\log\log|\Omega| \sqrt{\log|\Omega| \log(n)}\right).$$

From Theorem 5 we know that, given a sequence of ℓ inverse temperatures $\{\beta_i\}$ with stationary distributions that satisfy $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle|^2 \geq p$ for a constant $p > 0$, quantum annealing through the sequence of states $|\Pi_{\beta_i}\rangle$ corresponding to the Gibbs distributions Π_{β_i} at each inverse temperature β_i takes

$$O(\ell \delta^{-1/2} \log^2(\ell/\epsilon)(1/p) \log(1/p))$$

total steps of the quantum walk operators corresponding to the M_{β_i}. Note that here, unlike in the case of Bayesian inference, we need to show that $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle|^2 \geq p$ is satisfied as the B-Chebyshev condition instead guarantees that $|\langle \Pi_{\beta_i} | \Pi_{2\beta_{i+1}-\beta} \rangle|^2 \geq 1/B$ is satisfied. But we claim that satisfying the latter is enough to satisfy the former. To see this, note that $|\langle \Pi_{\beta_i} | \Pi_{2\beta_{i+1}-\beta} \rangle|^2 \geq p$ is equivalent to the B-Chebyshev condition with $B = 1/p$, and that the B-Chebyshev condition can be rewritten as

$$\frac{Z(\beta_i) Z(2\beta_{i+1} - \beta_i)}{Z(\beta_{i+1})^2} = \sum_{x \in \Omega} \frac{\Pi_{\beta_{i+1}}(x)^2}{\Pi_{\beta_i}(x)} \geq \frac{1}{p}. \quad (24)$$

The overlap $\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle$ which appears in the slow-varying condition can be rewritten as

$$\begin{aligned}\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle &= \sum_{x \in \Omega} \Pi_{\beta_{i+1}}(x) \sqrt{\frac{\Pi_{\beta_i}(x)}{\Pi_{\beta_{i+1}}(x)}} \\ &\geq \frac{1}{\sqrt{\sum_{x \in \Omega} \Pi_{\beta_{i+1}}(x) \frac{\Pi_{\beta_{i+1}}(x)}{\Pi_{\beta_i}(x)}}} = \sqrt{p}\end{aligned} \tag{25}$$

where we obtain the inequality from Jensen's inequality in the form $1/\sqrt{\mathbb{E}[X]} \leq \mathbb{E}[1/\sqrt{X}]$.

Since we simultaneously construct the schedule and anneal our state on the fly, we also need to account for the cost of constructing the schedule. At each inverse temperature β_i we perform binary search to find temperature $2\beta_{i+1} - \beta_i$ satisfying $|\langle \Pi_{\beta_i} | \Pi_{2\beta_{i+1}-\beta_i} \rangle|^2 \geq p$ in the interval $[\beta_i, \gamma]$. We choose binary search precision $1/n$ since $\Pi_\beta \propto e^{-\beta k}$ for $k \in \{0, n\}$, which means that the binary search procedure contributes a factor of $\log n$ to the complexity. For each candidate inverse temperature β' in the binary search, we perform nondestructive amplitude estimation to calculate $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2$. We set the failure probability of amplitude estimation to be $\eta/(n\ell)$. From Theorem 9, we can estimate $|\langle \Pi_{\beta_i} | \Pi_{\beta'} \rangle|^2 \geq p$ with error that is $O(\epsilon_e)$ using $O(1/\epsilon_e \log(1/\epsilon_e) \log(n\ell/\eta)/\sqrt{\delta})$ Markov chain steps. Since we take $\epsilon_e = p/10$, our binary search then guarantees that we can find a sequence of ℓ inverse temperatures satisfying both $|\langle \Pi_{\beta_i} | \Pi_{2\beta_{i+1}-\beta_i} \rangle|^2 \geq 9p/10$ and $|\langle \Pi_{\beta_i} | \Pi_{\beta_{i+1}} \rangle|^2 \geq 9p/10$ with a total cost of

$$O(\ell \delta^{-1/2} \log n (1/p) \log(1/p) \log(n\ell/\eta))$$

total Markov chain steps. Adding the two contributions from constructing the schedule and annealing the state, we get a total cost of

$$\begin{aligned}&O(\ell \delta^{-1/2} \log n \log^2(\ell/\epsilon)(1/p) \log(1/p) \log(n\ell/\eta)) \\ &\quad = \tilde{O}(\sqrt{(\log |\Omega|)/\delta})\end{aligned}$$

Markov chain steps. □

Adding these two contributions to the total number of Markov chain steps required (and noting that $O(B\ell/\epsilon)$ samples are needed at each of the ℓ inverse temperatures), we get a total complexity of $\tilde{O}((\log |\Omega|)/\sqrt{\delta}\epsilon)$, where ϵ is the error in computing the approximation to $Z(\infty)$. Thus we can finally state the following for the counting problem:

Theorem 14 (Quantum adaptive annealing algorithm for computing partition functions for counting problems)**.** *Given the counting problem partition function* $Z(\beta) = \sum_{k=0}^{n} a_k e^{-\beta k}$ *with* $|\Omega| = \sum_{k=0}^{n} a_k$ *and* $n = \max_x H(x)$*, assume that we can generate state* $|\Pi_0\rangle$ *corresponding to the uniform distribution over* Ω*. Letting* γ *be the temperature at which* $\log Z(\gamma) = 1$*, assume also that for every inverse temperature* $\beta \in [0, \gamma]$ *we have a Markov chain* M_β *with stationary distribution* Π_β *and spectral gap lower-bounded by* δ*. Then, for any* $\epsilon = O(1/\sqrt{\log\log|\Omega|})$ *and any* $\eta > 0$*, there is a quantum algorithm (Algorithm 2) that uses*

$$\begin{aligned}&O((\ell^2/\sqrt{\delta}\epsilon) \log^{5/2}(\ell/\epsilon) \log(\ell/\eta) \log\log(\ell/\epsilon)) \\ &\quad + O((\ell/\epsilon)\ell\delta^{-1/2} \log n \log^2(\ell/\epsilon) \log(n\ell/\eta)) \\ &\quad = \tilde{O}((\log|\Omega|)/\sqrt{\delta}\epsilon)\end{aligned}$$

steps of the Markov chains and outputs $\tilde{Z}$*, an approximation to* $Z(\infty)$ *such that*

$$\Pr\left[(1-\epsilon)Z(\infty) \leq \tilde{Z} \leq (1+\epsilon)Z(\infty)\right] \geq 1 - \eta.$$

In Section 5.1 we give several examples of partition function problems, and we evaluate the runtime of our algorithm on these examples.

4 Nondestructive Amplitude Estimation

In this section we first describe the amplitude estimation algorithm of Brassard, Hoyer, Mosca, and Tapp (BHMT) [5], and then we show how it can be made nondestructive. The result of BHMT can be stated as follows:

Theorem 15. *(BHMT [5] Theorem 12) Given state* $|\psi\rangle$ *and reflections* $R_\psi = 2|\psi\rangle\langle\psi| - I$ *and* $R = 2P - I$*, there exists a quantum algorithm that outputs* $\tilde{a}$*, an approximation to* $a = \langle\psi|P|\psi\rangle$*, so that*

$$|\tilde{a} - a| \leq 2\pi \frac{a(1-a)}{M} + \frac{\pi^2}{M^2}$$

with probability at least $8/\pi^2$ *and* M *uses of* R_ψ *and* R*.*

In amplitude estimation we are interested in the eigenspectrum of the Grover search operator, given by

$$Q = -R_\psi R. \tag{26}$$

We can decompose our original Hilbert space into $\mathcal{H}_1 = \mathrm{Im}(P)$ and its complement $\mathcal{H}_0$. Writing $|\psi\rangle$ as

$$|\psi\rangle = \sin\theta\,|\psi_1\rangle + \cos\theta\,|\psi_0\rangle \tag{27}$$

for $|\psi_1\rangle \in \mathcal{H}_1$ and $|\psi_0\rangle \in \mathcal{H}_0$, we note that on the space spanned by $\{|\psi_1\rangle, |\psi_0\rangle\}$, Q acts as

$$Q = \begin{pmatrix} \cos(2\theta) & \sin(2\theta) \\ -\sin(2\theta) & \cos(2\theta) \end{pmatrix}. \tag{28}$$

This matrix has eigenvalues $e^{\pm 2i\theta}$ with corresponding eigenvectors

$$|\psi_\pm\rangle = \frac{1}{\sqrt{2}}(|\psi_1\rangle \pm i\,|\psi_0\rangle). \tag{29}$$

Since $a = \langle\psi|P|\psi\rangle = \sin^2\theta$, estimating the eigenvalues of Q allows us to estimate a. To estimate the eigenvalues of Q, BHMT define the Fourier transform

$$F_M : |x\rangle \mapsto \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} e^{2\pi i x y/M} |y\rangle \tag{30}$$

and the state

$$|S_M(\omega)\rangle = \frac{1}{\sqrt{M}} \sum_{y=0}^{M-1} e^{2\pi i \omega y} |y\rangle. \tag{31}$$

Then performing $F_M^{-1}|S_M(\omega)\rangle$ and measuring in the computational basis allows us to perform phase estimation. Explicitly, according to BHMT Theorem 11,

Theorem 16. *(Phase estimation, BHMT Theorem 11) Let y be the random variable corresponding to the result of measuring $F_M^{-1}|S_M(\omega)\rangle$. If $M\omega$ is an integer, then* $\Pr[y = M\omega] = 1$. *Otherwise,*

$$P\left(\left|\frac{y}{M} - \omega\right| \le \frac{1}{M}\right) \ge \frac{8}{\pi^2}. \tag{32}$$

Finally, we will also need to define the operator

$$\Lambda_M(U) : |j\rangle|y\rangle \mapsto |j\rangle U^j |y\rangle. \tag{33}$$

Now we can state the amplitude estimation algorithm:

Algorithm 3 Amplitude estimation algorithm.

Input: State $|\psi\rangle$ and operators $R_\psi = 2|\psi\rangle\langle\psi| - I$ and $R = 2P - I$.
Output: $\tilde{a}$, an estimate of $\langle\psi|P|\psi\rangle$.

1: Start with state $|0\rangle|\psi\rangle$.
2: Apply operator $(F_M^{-1} \otimes I)\Lambda_M(Q)(F_M \otimes I)$.
3: Measure first register to obtain either state $|y\rangle|\psi_+\rangle$ or $|y\rangle|\psi_-\rangle$.
4: Return $\tilde{a} = \sin^2(\pi y/M)$.

We can boost the success probability of amplitude estimation using the powering lemma [12], which improves the amplitude estimation success probability of $8/\pi^2$ to $1-\eta$ for any $\eta > 0$ at the cost of an extra $O(\log 1/\eta)$ factor.

Lemma 17. *(Powering lemma [12]) Suppose we have an algorithm that produces an estimate $\tilde{\mu}$ of μ so that $|\mu - \tilde{\mu}| < \epsilon$ with some fixed probability $p > 1/2$. Then for any $\eta > 0$, repeating the algorithm $O(\log 1/\eta)$ times and taking the median suffices to produce $\tilde{\mu}$ with $|\mu - \tilde{\mu}| < \epsilon$ with probability at least $1 - \eta$.*

This allows us to state the following version of amplitude estimation with powering:

Algorithm 4 Amplitude estimation with powering.

Input: State $|\psi\rangle$, operators $R_\psi = 2|\psi\rangle\langle\psi| - I$ and $R = 2P - I$, and $\eta > 0$.
Output: $\tilde{a}$, an estimate of $\langle\psi|P|\psi\rangle$.

1: Start with state $|\psi\rangle$.
2: **for** i:=1 to $q = O(\log(1/\eta))$ **do**
3: Add a new register $|0\rangle_i$.
4: Apply operator $(F_M^{-1} \otimes I)\Lambda_M(Q)(F_M \otimes I)$ on subsystem $|0\rangle_i|\psi\rangle$.
5: **end for**
6: Add register $|0\rangle_{q+1}$ and apply the function that maps the median of the first q registers to this register.
7: Uncompute the first q registers.
8: Measure $(q+1)$-st register to obtain median y_m.
9: Return $\tilde{a} = \sin^2(\pi y_m/M)$.

After performing amplitude estimation, we'd like to restore our state to the initial starting state. To do so, we start by observing that we can rewrite the state $|\psi\rangle$ as

$$|\psi\rangle = \frac{1}{\sqrt{2}}(e^{-i\theta}|\psi_+\rangle + e^{i\theta}|\psi_-\rangle). \tag{34}$$

Then applying the operator of step 2 of Algorithm 3

yields the following sequence of states:

$$
\begin{aligned}
&((F_M^{-1}\otimes I)\Lambda_M(Q)(F_M\otimes I))\,|0\rangle\,|\psi\rangle \\
&= ((F_M^{-1}\otimes I)\Lambda_M(Q)(F_M\otimes I))\left(\frac{1}{\sqrt{2}}\,|0\rangle\,(e^{-i\theta}\,|\psi_+\rangle \right. \\
&\qquad \left. +e^{i\theta}\,|\psi_-\rangle)\right) \\
&= ((F_M^{-1}\otimes I)\Lambda_M(Q))\left(\frac{1}{\sqrt{2M}}\sum_{j=0}^{M-1}|j\rangle\,(e^{-i\theta}\,|\psi_+\rangle \right. \\
&\qquad \left. +e^{i\theta}\,|\psi_-\rangle)\right) \\
&= (F_M^{-1}\otimes I)\left(\frac{e^{-i\theta}}{\sqrt{2M}}\sum_{j=0}^{M-1}e^{2ij\theta}\,|j\rangle\,|\psi_+\rangle \right. \\
&\qquad \left. +\frac{e^{i\theta}}{\sqrt{2M}}\sum_{j=0}^{M-1}e^{-2ij\theta}\,|j\rangle\,|\psi_-\rangle\right) \\
&= \frac{e^{-i\theta}}{\sqrt{2}}(F_M^{-1}\,|S_M(\theta/\pi)\rangle)\,|\psi_+\rangle \\
&\qquad + \frac{e^{i\theta}}{\sqrt{2}}(F_M^{-1}\,|S_M(1-\theta/\pi)\rangle)\,|\psi_-\rangle
\end{aligned}
$$

Thus after the measurement in step 3, the algorithm will always end in either of the two states $|j\rangle\,|\psi_\pm\rangle$.

Note that we'd like to restore this to the starting state $|0\rangle\,|\psi\rangle$, and that $|\langle\psi|\psi_\pm\rangle|^2 = 1/2$ is a constant. Since this overlap is constant, and since we are working with two-dimensional subspaces, we can restore the state using a scheme similar to that of Temme et. al. [28], which was in turn inspired by a scheme of Marriott and Watrous [16].[1] That is, given $|\psi_\pm\rangle$, we first apply the projection operator $|\psi\rangle\langle\psi| = (R_\psi + I)/2$. We either obtain $|\psi\rangle$, in which case we are done, or we obtain some $|\psi^\perp\rangle$ so that $\langle\psi|\psi^\perp\rangle = 0$. Since $|\psi^\perp\rangle$ can also be expressed in the basis $\{|\psi_+\rangle, |\psi_-\rangle\}$, we can again apply amplitude estimation to collapse the last register onto either $|\psi_+\rangle$ or $|\psi_-\rangle$. Then we repeat the projection onto $|\psi\rangle$. Since the overlap between $|\psi\rangle$ and $|\psi_\pm\rangle$ is constant, the expected numbers of times we need to perform the series of projections before attaining our desired state $|\psi\rangle$ is constant as well.

This suggests the following algorithm for state restoration:

[1] We thank Fernando Brandão for discussions related to this point.

Algorithm 5 State restoration following amplitude estimation.

Input: $\eta > 0$; either state $|\psi_+\rangle$ or $|\psi_-\rangle$; and operators $R_\psi = 2|\psi\rangle\langle\psi| - I$ and $R = 2P - I$, where $|\psi_\pm\rangle$ are the eigenstates of $Q = -R_\psi R$ with eigenvalues $e^{\pm 2i\theta}$.
Output: State $|\psi\rangle$.

1: **while** current state is not $|\psi\rangle$ **do**
2: Apply $(R_\psi + I)/2$.
3: **if** current state is $|\psi\rangle$ **then**
4: Return $|\psi\rangle$.
5: **end if**
6: **for** i:=1 to $q = O(\log(1/\eta))$ **do**
7: Add a new register $|0\rangle_i$.
8: Apply operator $(F_M^{-1}\otimes I)\Lambda_M(Q)(F_M\otimes I)$ on subsystem $|0\rangle_i\,|\psi\rangle$.
9: **end for**
10: Add register $|0\rangle_{q+1}$ and apply the function that maps the median of the first q registers to this register.
11: Uncompute the first q registers.
12: Measure $(q+1)$-st register to obtain either $|\psi_+\rangle$ or $|\psi_-\rangle$.
13: **end while**

Performing amplitude estimation according to Algorithm 4 with failure probability less than $\eta/2$, and then performing state restoration according to Algorithm 5 with failure probability less than $\eta/2$, gives us an algorithm for nondestructive amplitude estimation with probability of success at least $1-\eta$:

Theorem 18. *(Nondestructive amplitude estimation) Given state $|\psi\rangle$ and reflections $R_\psi = 2|\psi\rangle\langle\psi| - I$ and $R = 2P - I$, and any $\eta > 0$, there exists a quantum algorithm that outputs $\tilde{a}$, an approximation to $a = \langle\psi|P|\psi\rangle$, so that*

$$|\tilde{a} - a| \leq 2\pi\frac{a(1-a)}{M} + \frac{\pi^2}{M^2}$$

with probability at least $1-\eta$ and $O(\log(1/\eta)M)$ uses of R_ψ and R. Moreover the algorithm restores the state $|\psi\rangle$ with probability at least $1-\eta$.

5 Discussion and Applications

5.1 Applications to Partition Function Problems

In this section, following the treatment of [18] and [25], we give several examples of problems from statistical physics and computer science that can be

framed as partition function problems. We then show how our algorithm can be applied to obtain a speedup. We obtain a quadratic improvement in the scaling with ϵ due to Montanaro's algorithm for computing expectation values [18], and we obtain an improvement in the scaling with graph parameters due to the adaptive schedule of [25] and the QSA algorithm of [31]. The results are summarized in Table 2 and elaborated below.

Counting k-colorings In the k-coloring problem, we are given a graph $G = (V, E)$ with maximum degree Δ, and we'd like to count the number of ways to color the vertices with k colors such that no two adjacent vertices share the same color (in statistical physics, this problem is also known as the antiferromagnetic Potts model at zero temperature). Here Ω is the set of colorings of G, and for each $\sigma \in \Omega$, $H(\sigma)$ is the number of monochromatic edges in σ. Thus we have the partition function

$$Z(\beta) = \sum_{\sigma \in \Omega} e^{-\beta H(\sigma)}.$$

We know that $|\Omega| = Z(0) = k^{|V|}$, and we'd like to calculate $Z(\infty)$, corresponding to the number of valid k-colorings. Jerrum [9] showed that using Glauber dynamics, a single site update Markov chain, it is possible to obtain mixing time $O(|V| \log |V|)$ whenever $k > 2\Delta$. Thus our quantum algorithm can obtain an approximation for the k-coloring problem in time $\tilde{O}(|V|^{3/2}/\epsilon)$, whereas the classical algorithm of SVV scales like $\tilde{O}(|V|^2/\epsilon^2)$, and the partially quantum algorithm of Montanaro scales like $\tilde{O}(|V|^{3/2}/\epsilon + |V|^2)$.

Ising Model The Ising model on a graph $G = (V, E)$ is a model from statistical physics where we place a spin at each vertex and assign each spin a value of $+1$ or -1. The Hamiltonian counts the number of edges whose endpoints have different spins. Here the space of possible assignments is given by $\Omega = \{\pm 1\}^{|V|}$, so $|\Omega| = Z(0) = 2^{|V|}$. The Ising model has been extensively studied, and results such as [17, 19] show that in certain regimes, Glauber dynamics mixes rapidly, in time $O(|V| \log |V|)$. Thus our quantum algorithm scales like $\tilde{O}(|V|^{3/2}/\epsilon)$, while the classical algorithm of SVV [25] scales like $\tilde{O}(|V|^2/\epsilon^2)$, and the partially quantum algorithm of Montanaro [18] scales like $\tilde{O}(|V|^{3/2}/\epsilon + |V|^2)$.

Counting Matchings A matching over a graph $G = (V, E)$ is a subset of edges that share no vertex in common. Letting Ω denote the set of all matchings over G, we then have a partition function of the form

$$Z(\beta) = \sum_{M \in \Omega} e^{-\beta |M|}.$$

Then we know that $Z(\infty) = 1$, and we seek to calculate $Z(0) = |\Omega|$. Here we would need to anneal backwards in temperature; that is, if we had inverse temperatures $\beta_0 = 0 < \beta_1 < \ldots < \beta_\ell = \infty$, we would want to anneal in the reverse order,

$$Z(0) = Z(\infty) \frac{Z(\beta_{\ell-1})}{Z(\infty)} \frac{Z(\beta_{\ell-2})}{Z(\beta_{\ell-1})} \cdots \frac{Z(0)}{Z(\beta_1)}.$$

We would want to satisfy the Chebyshev condition in reverse as well; that is, we'd like to have

$$\frac{Z(2\beta_i - \beta_{i+1}) Z(\beta_{i+1})}{Z(\beta_i)^2} \leq B$$

Note that as in the case of the non-reversed schedule, we take $\beta_{\ell-1} = \gamma_0$ so that $Z(\gamma_0) = e$ in order to satisfy the Chebyshev condition between $\beta_{\ell-1}$ and $\beta_\ell = \infty$. Next we need to anneal backwards from $\beta = \gamma_0$ to $\beta = 0$. To do this we will modify the partition function to

$$Z(\beta') = \sum_{x \in \Omega} e^{(\beta' - \gamma_0) H(x)}$$

and anneal forwards from $\beta' = 0$, corresponding to $Z(\beta' = 0) = Z(\beta = \gamma_0) = e$, to $\beta' = \gamma_0$, corresponding to $Z(\beta' = \gamma_0) = Z(\beta = 0) = |\Omega|$. Since $Z(\beta')$ is still a convex function, the results from Appendix A and Section 2 guaranteeing the existence of a quadratically shorter schedule satisfying the Chebyshev condition still apply. (Note that the original paper by SVV [25] showed the existence of this cooling schedule for $\log Z(\beta)$ a decreasing function, but the argument in Appendix A applies equally well to increasing convex functions.)

Jerrum and Sinclair [10] showed that the Markov chain for computing matchings has mixing time $O(|V||E|)$. Since $|\Omega| = O(|V|! \cdot 2^{|V|})$, our quantum algorithm has complexity $\tilde{O}(|V|^{3/2}|E|^{1/2}/\epsilon)$, compared to the $\tilde{O}(|V|^2|E|/\epsilon^2)$ complexity of SVV [25] and the $\tilde{O}(|V|^{3/2}|E|^{1/2}/\epsilon + |V|^2|E|)$ complexity of Montanaro [18].

Counting Independent Sets An independent set on a graph $G = (V, E)$ with maximum degree Δ is a set of vertices that share no edge. Letting Ω denote the set of independent sets on G, and given a fugacity $\lambda > 0$, we define

$$Z(\beta) = \sum_{\sigma \in \Omega} \lambda^{|\sigma|}.$$

Again we know that $Z(\infty) = 1$, and we seek to calculate $Z(0) = |\Omega|$. As with the case of counting matchings, we can anneal backwards by modifying the partition function.

Vigoda [29] showed that Glauber dynamics results in a mixing time of $O(|V| \log |V|)$ whenever $\lambda < 2/(\Delta - 2)$. Since $|\Omega| = O(2^{|V|})$, our quantum algorithm has complexity $\tilde{O}(|V|^{3/2}/\epsilon)$, while the classical algorithm of SVV [25] scales like $\tilde{O}(|V|^2/\epsilon^2)$, and the algorithm of Montanaro [18] scales like $\tilde{O}(|V|^{3/2}/\epsilon + |V|^2)$.

5.2 Warm Starts and Nonadaptive Schedules

Montanaro's quantum algorithm [18] is already a sort of quantum version of SVV [25]. So why doesn't it already achieve what we do? Montanaro cites two related obstacles: warm starts and nonadaptive schedules. In this section we will explain how warm starts are used by SVV, and why SVV use nonadaptive schedules to construct a schedule with warm starts. For SVV this choice was not strictly necessary, but rather due to the fact that they consider applications to counting problems, where there is almost no additional cost to using nonadaptive schedules to ensure warm starts. In the quantum case warm starts are still desirable, but achieving them using nonadaptive schedules is too costly, especially without the nondestructive amplitude estimation that we introduced in Section 4. This led Montanaro to develop an algorithm that still relied on SVV's classical algorithm to construct a schedule with warm starts, and then used this schedule as input to the quantum walks.

We now explain these points in more detail.

Warm starts. The idea behind warm starts for classical random walks is that the spectral gap (directly) controls convergence in the 2-norm while applications usually require bounds in the 1-norm. This norm conversion introduces some cost which is greatly reduced by starting the random walk in a distribution that is close to the target distribution, aka a "warm start."

To make this more concrete, we define two notions of distance between probability distributions. The total variation distance is

$$\|\Pi_1 - \Pi_2\|_{TV} = \frac{1}{2} \sum_{x \in \Omega} |\Pi_1(x) - \Pi_2(x)|$$

and the L^2 distance, which is also a variance, is

$$\left\| \frac{\Pi_1}{\Pi_2} - 1 \right\|_{2,\Pi_2}^2 = \mathrm{Var}_{\Pi_2}(\Pi_1/\Pi_2) = \sum_{x \in \Omega} \Pi_2(x) \left(\frac{\Pi_1(x)}{\Pi_2(x)} - 1 \right)^2.$$

Now consider a Markov chain with stationary distribution Π, and suppose that we run this Markov chain on a starting distribution ν_0 for t steps to obtain distribution ν_t. Letting δ be the spectral gap of the Markov chain, we have

$$\|\nu_t - \Pi\|_{TV} \leq e^{-\delta t/2} \left\| \frac{\nu_0}{\Pi} - 1 \right\|_{2,\Pi} \tag{35}$$

(see, for example, SVV [25] Lemma 7.3). In particular, the idea behind warm starts is to pick a warm start distribution ν_0 so that the variance $\left\| \frac{\nu_0}{\Pi} - 1 \right\|_{2,\Pi}$ is bounded. A "cold start", on the other hand, would be a choice of ν_0 that is far from Π, such as putting probability 1 on a single point. Evaluating eq. (35) for such a distribution yields Aldous's inequality [2], which bounds the mixing time by $\leq \delta^{-1} \log(1/\min_x \Pi(x))$. Thus a warm start can be seen as avoiding the term $\log(1/\min_x \Pi(x))$, which often will be $O(n)$ for a Markov chain on n bits.

The benefits of warm starts for quantum algorithms, specifically that of Wocjan-Abeyesinghe [31], are much higher. Indeed, a reflection about $|\Pi\rangle$ takes time $O(1/\sqrt{\delta})$, while mapping an arbitrary starting state $|\psi\rangle$ to $|\Pi\rangle$ using a generalized Grover algorithm takes $O(1/|\langle\psi|\Pi\rangle|)$ reflections. Szegedy [26] and MNRS [15] perform such a series of reflections to obtain a quantum walk search algorithm whose runtime scales as $O(1/\sqrt{\delta \min_x \Pi(x)})$, resulting in a dependence on overlap that is exponentially worse than the classical case in eq. (35). By annealing through a judicious choice of starting states, Wocjan-Abeyesinghe [31] avoid this term at the cost of introducing a dependence on ℓ, the annealing schedule length.

Nonadaptive schedules. SVV focus specifically on the problem of approximate counting, not Bayesian inference, so they can use nonadaptive schedules to ensure warm starts at almost no additional cost. Suppose that we would like to construct an adaptive temperature schedule of length ℓ. In the case of approximate counting, where we need to estimate each of the ℓ terms in eq. (7), we need $O(\ell/\epsilon^2)$ (classical) samples at each temperature, incurring a total cost of $O(\ell^2/\epsilon^2)$. (Note that this oversimplifies slightly and leaves out some additional factors.)

Since a nonadaptive schedule has length $O(\ell^2)$, taking one sample from each of the $O(\ell^2)$ temperatures would not lead to any asymptotic increase in cost. For this reason SVV choose to begin with a nonadaptive schedule of length $O(\ell^2)$, where each temperature can be easily shown to provide a warm start for the next. Then they can select a subset of ℓ temperatures to repeatedly sample in order to estimate the partition function.

Montanaro observed (see [18, Section 3.3]) that this approach does not combine well with quantum walks. Quantum walks cannot directly create states at a given temperature without prohibitive cost, and the no-cloning theorem means that we cannot keep copies of the states produced along the way without recreating them from scratch. If we need one copy of each state at a sequence of ℓ temperatures then we need to run a quantum walk $(1+2+\ldots+\ell)/\sqrt{\delta} = O(\ell^2/\sqrt{\delta})$ times, which further increases to $O(\ell^3/\sqrt{\delta})$ if we need to select ℓ temperatures out of a list of ℓ^2 temperatures. (We ignore the dependence on accuracy and error probability here for simplicity.)

Our strategy for constructing the ℓ-step adaptive schedule never needs to create an $O(\ell^2)$-step nonadaptive schedule, and this change did not require major new ideas. However, it alone is not enough, because without the ability to reuse states we would still incur the $O(\ell^2/\sqrt{\delta})$ cost described above.

Non-destructive amplitude estimation. The missing ingredient in previous work is our Theorem 6, which shows that amplitude estimation can be made nondestructive. We use this both to create the schedule and to estimate the terms $Z(\beta_{i+1})/Z(\beta_i)$ in eq. (7). For Bayesian inference this is an important piece of our speedup, as it allows us to achieve time $\tilde{O}(\ell/\sqrt{\delta})$ instead of $\tilde{O}(\ell^2/\sqrt{\delta})$. As a result it becomes worthwhile to drop the nonadaptive schedule of SVV. For approximate counting we cannot avoid an ℓ^2 dependence in our $\tilde{O}(\ell^2/\sqrt{\delta}\epsilon)$ runtime, but dropping the nonadaptive schedule does remove the additive term of $O(\ell^2/\delta)$ that appeared in [18].

5.3 Conclusion

To summarize, we have shown how to combine quantum simulated annealing with shorter adaptive annealing schedules, resulting in a QSA algorithm that displays a quadratic improvement in dependence on both schedule length and inverse spectral gap when compared against a nonadaptive classical annealing algorithm. We have demonstrated applications to Bayesian inference and estimating partition functions of counting problems, and in the process we have also shown that amplitude estimation can be made nondestructive, a result that is useful in its own right.

This paper can be viewed as part of the broader goal of finding quadratic (or other polynomial) speedups of as many general-purpose classical algorithms as possible. Grover's algorithm can be interpreted as a square-root speedup for exhaustive search, and likewise there are easy quantum quadratic speedups for rejection sampling. However, the best classical algorithms for counting and Bayesian inference are much better than naive enumeration or rejection sampling. While simulated annealing with an adaptive schedule is still a generic algorithm, it is often much closer to the state of the art, and so it is worthwhile to try to find a quantum speedup for it. We do not fully square root its runtime since our sequence length is essentially the same as the best classical result (instead of quadratically worse as in previous quantum results), but our runtime dependence on accuracy and spectral gap are both quadratically better than those of classical algorithms.

Within the paradigm of simulated annealing we are unlikely to see further improvements in sequence length or dependence on accuracy or spectral gap. However, our algorithm for Bayesian inference does improve on classical algorithms by returning a qsample instead of a classical sample. We hope that future algorithms will use this fact to find further quantum algorithmic advantages.

A Bounding the Length of the Cooling Schedule

Here we provide the proof of Lemma 2, which is a slight modification of Lemma 4.3 in SVV [25]. We use this result to demonstrate the existence of a temperature schedule satisfying the bounded variance (12) and slow-varying (13) conditions, or equivalently (14), and to bound the length of such a schedule.

Lemma 19. *(Modified from SVV [25] Lemma 4.3) For f a convex function over domain $[0,\gamma]$, there exists a sequence $\gamma_0 < \gamma_1 < \ldots < \gamma_\ell$ with $\gamma_0 = 0$ and $\gamma_\ell = \gamma$ satisfying*

$$f\left(\frac{\gamma_i+\gamma_{i+1}}{2}\right) \geq \frac{f(\gamma_i)+f(\gamma_{i+1})}{2} - 1 \tag{36}$$

with length

$$\ell \leq \sqrt{(f(0)-f(\gamma))\log\left(\frac{f'(0)}{f'(\gamma)+1}\right)}. \tag{37}$$

Proof of Lemma 19. Suppose we have already constructed the sequence up to γ_i. Let γ_{i+1} be the largest

value in $[\gamma_i, \gamma]$ so that γ_i and γ_{i+1} satisfy equation (36), and let $m_i = (\gamma_i + \gamma_{i+1})/2$, $\Delta_i = (\gamma_{i+1} - \gamma_i)/2$, and $K_i = f(\gamma_i) - f(\gamma_{i+1})$. Then, since γ_{i+1} satisfies equation (36),

$$f(m_i) \geq \frac{f(\gamma_i) + f(\gamma_{i+1})}{2} - 1. \tag{38}$$

By convexity,

$$f'(\gamma_i) \leq \frac{f(\gamma_{i+1}) - f(\gamma_i)}{\gamma_{i+1} - \gamma_i}. \tag{39}$$

We can rewrite this as

$$-f'(\gamma_i) \geq \frac{K_i}{2\Delta_i}. \tag{40}$$

Also by convexity,

$$f'(\gamma_{i+1}) \geq \frac{f(m_i) - f(\gamma_{i+1})}{m_i - \gamma_{i+1}}. \tag{41}$$

Combining this with equation (38),

$$-f'(\gamma_{i+1}) \leq \frac{K_i - 2}{2\Delta_i} \tag{42}$$

and

$$-f'(\gamma_{i+1}) - 1 \leq \frac{K_i - 2}{2\Delta_i} - 1. \tag{43}$$

Then, combining equations (40) and (42),

$$\frac{f'(\gamma_{i+1})}{f'(\gamma_i)} \leq 1 - \frac{2}{K_i} \leq 1 - \frac{1}{K_i} \leq e^{-1/K_i}. \tag{44}$$

Similarly, combining equations (40) and (43),

$$\frac{f'(\gamma_{i+1}) + 1}{f'(\gamma_i)} \leq 1 - \frac{2 + 2\Delta_i}{K_i} \leq 1 - \frac{1}{K_i} \leq e^{-1/K_i}. \tag{45}$$

Summing the K_i,

$$\sum_{i=0}^{\ell-1} K_i = f(0) - f(\gamma). \tag{46}$$

Summing equation (44) over K_i for $i = 0$ to $\ell - 2$, and adding equation (45) for $i = \ell - 1$, we get that

$$\sum_{i=0}^{\ell-1} \frac{1}{K_i} \leq \log\left(\frac{f'(0)}{f'(\gamma) + 1}\right). \tag{47}$$

By the Cauchy-Schwarz inequality on equations (46) and (47),

$$\ell^2 \leq (f(0) - f(\gamma)) \log\left(\frac{f'(0)}{f'(\gamma) + 1}\right). \tag{48}$$

□

Acknowledgements

We would like to thank Ashley Montanaro and Fernando Brandão for helpful conversations and suggestions. AYW would like to acknowledge the support of the DOE CSGF. AWH was funded by NSF grants CCF-1452616, CCF-1729369, PHY-1818914, ARO contract W911NF-17-1-0433 and a Samsung Advanced Institute of Technology Global Research Partnership.

References

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

[2] D. Aldous. Some inequalities for reversible Markov chains. *Journal of the London Mathematical Society*, 25:564–576, 1982.

[3] A. Ambainis, A. Gilyen, S. Jeffery, and M. Kokainis. Quantum speedup for finding marked vertices by quantum walks, 2019, arXiv:1903.07493.

[4] S. Apers and A. Sarlette. Quantum fast-forwarding: Markov chains and graph property testing, 2018, arXiv:1804.02321.

[5] G. Brassard, P. Høyer, M. Mosca, and A. Tapp. *Quantum Amplitude Amplification and Estimation*, volume 305 of *Contemporary Mathematics Series Millenium Volume*. AMS, 2002, arXiv:quant-ph/0005055.

[6] M. Dyer, A. Frieze, and R. Kanna. A random polynomial time algorithm for approximating the volume of convex bodies. *Journal of the ACM*, 38(1):1–17, 1991.

[7] L. Grover and T. Rudolph. Creating superpositions that correspond to efficiently integrable probability distributions, 2002, arXiv:quant-ph/0208112.

[8] M. Huber. Approximation algorithms for the normalizing constant of Gibbs distributions. *arXiv e-prints*, page arXiv:1206.2689, Jun 2012, arXiv:1206.2689.

[9] M. Jerrum. A very simple algorithm for estimating the number of k-colorings of a low-degree graph. *Random Structures & Algorithms*, 7(2):157–165, 1995.

[10] M. Jerrum and A. Sinclair. Approximating the permanent. *SIAM Journal on Computing*, 18:1149–1178, 1989.

[11] M. Jerrum, A. Sinclair, and E. Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with nonnegative entries. *J. ACM*, 51(4):671–697, 2004.

[12] M. Jerrum, L. Valiant, and V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoretical computer science*, 43(2-3):169–188, 1986.

[13] P. Kaye and M. Mosca. Quantum Networks for Generating Arbitrary Quantum States. *arXiv e-prints*, pages quant–ph/0407102, Jul 2004, arXiv:quant-ph/0407102.

[14] G. H. Low, T. J. Yoder, and I. L. Chuang. Quantum inference on bayesian networks. *Physical Review A*, 89(6):062315, 2014, arXiv:1402.7359.

[15] F. Magniez, A. Nayak, J. Roland, and M. Santha. Search via quantum walk. *SIAM Journal on Computing*, 40(1):142–164, 2011, arXiv:quant-ph/0608026.

[16] C. Marriott and J. Watrous. Quantum arthur-merlin games. *Computational Complexity*, 14(2):122–152, 2005, arXiv:cs/0506068.

[17] F. Martinelli and E. Olivieri. Approach to equilibrium of Glauber dynamics in the one phase region. *Communications in Mathematical Physics*, 161(3):447–486, 1994.

[18] A. Montanaro. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 471(2181), 2015, arXiv:1504.06987.

[19] E. Mossel, A. Sly, et al. Exact thresholds for Ising–Gibbs samplers on general graphs. *The Annals of Probability*, 41(1):294–328, 2013.

[20] D. Orsucci, H. J. Briegel, V. Dunjko, et al. Faster quantum mixing for slowly evolving sequences of Markov chains. *Quantum*, 2:105, 2018, arXiv:1503.01334.

[21] M. Ozols, M. Roetteler, and J. Roland. Quantum rejection sampling. *ACM Transactions on Computation Theory (TOCT)*, 5(3):11:1–11:33, 2013, arXiv:1103.2774.

[22] P. C. Richter. Quantum speedup of classical mixing processes. *Physical Review A*, 76(4):042306, 2007, arXiv:quant-ph/0609204.

[23] R. Somma, S. Boixo, and H. Barnum. Quantum simulated annealing, 2007, arXiv:0712.1008.

[24] R. Somma, S. Boixo, H. Barnum, and E. Knill. Quantum simulations of classical annealing processes. *Phys. Rev. Lett.*, 101(13):130504, 2008, arXiv:0804.1571.

[25] D. Štefankovič, S. Vempala, and E. Vigoda. Adaptive simulated annealing: A near-optimal connection between sampling and counting. *Journal of the ACM (JACM)*, 56(3):18, 2009, arXiv:cs.DS/0612058.

[26] M. Szegedy. Quantum speed-up of Markov chain based algorithms. In *FOCS '04: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 32–41, Washington, DC, USA, 2004. IEEE Computer Society, arXiv:quant-ph/0401053.

[27] E. Tang. Some settings supporting efficient state preparation. https://ewintang.com/blog/2019/06/13/some-settings-supporting-efficient-state-preparation/ 2019.

[28] K. Temme, T. J. Osborne, K. G. Vollbrecht, D. Poulin, and F. Verstraete. Quantum Metropolis sampling. *Nature*, 471(7336):87–90, 2011, arXiv:0911.3635.

[29] E. Vigoda. A note on the glauber dynamics for sampling independent sets. *The Electronic Journal of Combinatorics*, 8(1):8, 2001.

[30] N. Wiebe and C. Granade. Can small quantum systems learn?, 2015, arXiv:1512.03145.

[31] P. Wocjan and A. Abeyesinghe. Speedup via quantum sampling. *Phys. Rev. A*, 78:042336, 2008, arXiv:0804.4259.

[32] M.-H. Yung and A. Aspuru-Guzik. A quantum–quantum metropolis algorithm. *Proceedings of the National Academy of Sciences*, 109(3):754–759, 2012, arXiv:1011.1468.

[33] C. Zalka. Efficient simulation of quantum systems by quantum computers. *Proc. Roy. Soc. Lond.*, A454:313–322, 1998, arXiv:quant-ph/9603026.