LA-UR-16-21218 (Accepted Manuscript)

Quantum algorithms for Gibbs sampling and hitting-time estimation

Somma, Rolando Diego
Chowdhury, Anirban

To be published in: Quantum Information and Computation

DOI to publisher's version:

Permalink to record: http://permalink.lanl.gov/object/view?what=info:lanl-repo/lareport/LA-UR-16-21218

Quantum algorithms for Gibbs sampling and hitting-time estimation

Anirban Narayan Chowdhury
Center for Quantum Information and Control, University of New Mexico, Albuquerque, NM 87131, US and New Mexico Consortium, Los Alamos, NM 87545, US.

Rolando D. Somma
Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, US.

(Dated: March 10, 2016)

We present quantum algorithms for solving two problems regarding stochastic processes. The first algorithm prepares the thermal Gibbs state of a quantum system and runs in time almost linear in $\sqrt{N\beta/\mathcal{Z}}$ and polynomial in $\log(1/\epsilon)$, where N is the Hilbert space dimension, β is the inverse temperature, $\mathcal{Z}$ is the partition function, and ϵ is the desired precision of the output state. Our quantum algorithm exponentially improves the dependence on $1/\epsilon$ and quadratically improves the dependence on β of known quantum algorithms for this problem. The second algorithm estimates the hitting time of a Markov chain. For a sparse stochastic matrix P, it runs in time almost linear in $1/(\epsilon\Delta^{3/2})$, where ϵ is the absolute precision in the estimation and Δ is a parameter determined by P, and whose inverse is an upper bound of the hitting time. Our quantum algorithm quadratically improves the dependence on $1/\epsilon$ and $1/\Delta$ of the analog classical algorithm for hitting-time estimation. Both algorithms use tools recently developed in the context of Hamiltonian simulation, spectral gap amplification, and solving linear systems of equations.

PACS numbers: 03.67.Ac, 89.70.Eg

I. INTRODUCTION

Two important problems in statistical mechanics and stochastic processes are sampling from the thermal or Gibbs distribution of a physical system at a certain temperature and the estimation of hitting times of classical Markov chains. The first such problem has a wide range of applications as it allows us to compute quantities like the partition function, energy, or entropy of the system, and understand its physical properties in thermal equilibrium [1]. This problem has also applications in many other scientific areas including optimization [2]. Hitting times are also paramount in the study of classical random processes and they allow for a characterization of Markov chains [3]. Roughly, a hitting time is the time required by a diffusive random walk to reach a particular configuration with high probability. Besides their use in physics, hitting times are also important in solving search problems where the goal is to find a marked configuration of the Markov chain [4].

In a classical setting, these two problems are commonly solved using Monte-Carlo techniques [5]. Each step in a Monte-Carlo simulation corresponds to applying a particular probability rule that determines a Markov chain and an associated stochastic matrix. In the case of sampling from Gibbs distributions, for example, the fixed point of the Markov chain (i.e., the eigenvector of the stochastic matrix with eigenvalue 1) corresponds to the desired distribution. Such a distribution can then be prepared by repeated applications of the probability rule. To sample from probability distributions associated with thermal Gibbs states of quantum systems, quantum Monte-Carlo techniques may be used [6]. The running time of a Monte-Carlo simulation is typically dominated by the number of times the probability rule is applied to prepare the desired distribution with some given precision. This running time depends on properties of the Markov chain such as the spectral gap of the stochastic matrix [3].

In recent years, there has been significant interest in the development of quantum algorithms for simulating stochastic processes. Quantum algorithms for thermal Gibbs state preparation were developed in various works (c.f., [7–12]) and showed to provide polynomial quantum speedups in terms of various parameters, such as the spectral gap of the stochastic matrix or the dimension of the Hilbert space. The notion of quantum hitting time was also introduced in numerous works (c.f., [13–17]). Often, quantum hitting times of quantum walks on different graphs are significantly (e.g., polynomially) smaller than their classical counterparts. There are also various quantum algorithms to accelerate classical Monte-Carlo methods for estimating different quantities, such as expected values or partition functions (c.f., [18, 19]). Our results advance these areas further by providing new quantum algorithms with various improvements in the running time with respect to known classical and quantum algorithms for some of these problems.

In more detail, we present two quantum algorithms for preparing thermal Gibbs states of quantum systems and for estimating hitting times, respectively. The first algorithm runs in time $\tilde{O}(\sqrt{N\beta/\mathcal{Z}})$, where N is the Hilbert space dimension, β is the inverse temperature, and $\mathcal{Z}$ is the partition function of the quantum system. The $\tilde{O}$ notation hides polylogarithmic factors in these quantities and $1/\epsilon$, where ϵ is the desired precision of the output state. This is a quadratic improvement in β and an exponential improvement in $1/\epsilon$ with respect to a related algorithm presented in [8, 9]. In fact, the main difference

between our quantum algorithm and that of [8, 9] is in the implementation of the operator $e^{-\beta H/2}$, where H is the Hamiltonian of the system. Rather than using phase estimation, we use a technique introduced in [20–22] to decompose $e^{-\beta H/2}$ as a linear combination of unitary operations and then apply results of spectral gap amplification in [23] to implement each such unitary. The same idea can be used to improve the running time of the algorithm presented in [10]. The second algorithm provides an estimate of t_h, the hitting time of a reversible, irreducible, and aperiodic Markov chain. It runs in time $\tilde{O}(1/(\epsilon\Delta^{3/2}))$, where ϵ is the absolute precision in the estimation and Δ is a parameter that satisfies $1/\Delta \geq t_h$. The $\tilde{O}$ notation hides factors that are polynomial in $\log(1/(\epsilon\Delta))$ and $\log(N)$, where N is the dimension of the configuration space. In addition to the techniques used by the first algorithm, the second algorithm also uses recent methods for the quantum linear systems algorithm in [22] and methods to estimate quantities at the so-called quantum metrology limit described in [18].

The paper is organized as follows. In Sec. II we describe the main techniques introduced in [18, 21–24] that are also used by our algorithms. Then, the quantum algorithm for the preparation of thermal Gibbs states of quantum systems is described in Sec. III and the quantum algorithm for estimating hitting times of classical Markov chains is described in Sec. IV. We provide concluding remarks in Sec. V.

II. MAIN TECHNIQUES

Our algorithms are based on techniques developed in the context of spectral gap amplification [23], Hamiltonian simulation [21, 24], quantum metrology [18], and solving linear systems of equations [22]. We first consider an arbitrary finite-dimensional quantum system modeled by a Hamiltonian H that satisfies

$$H\left|\psi_j\right\rangle = E_j\left|\psi_j\right\rangle \ . \tag{1}$$

E_j are the eigenenergies and $|\psi_j\rangle$ are the eigenstates, $j = 0, 1, \ldots N-1$, and N is the dimension of the Hilbert space. We assume that H describes a system of n qubits and $N = 2^n$ [20, 25]. Furthermore, we assume that H can be decomposed as

$$H = \sum_{k=1}^{K} h_k \ , \tag{2}$$

where each $h_k \geq 0$ is a semidefinite positive Hermitian operator. In some cases, the assumption on h_k can be satisfied after a simple rescaling of H depending on its specification.

The results in [23] use the Hamiltonian

$$\tilde{H} = \sum_{k=1}^{K} \sqrt{h_k} \otimes \left(\left|k\right\rangle\!\left\langle 0\right|_{\mathrm{a}_1} + \left|0\right\rangle\!\left\langle k\right|_{\mathrm{a}_1}\right) \ , \tag{3}$$

where a_1 refers to an ancillary qubit register of dimension $O(\log(K))$. The important property is

$$(\tilde{H})^2\left|\phi\right\rangle \otimes \left|0\right\rangle_{\mathrm{a}_1} = \left(H\left|\phi\right\rangle\right) \otimes \left|0\right\rangle_{\mathrm{a}_1} \ , \tag{4}$$

for any $|\phi\rangle$. Roughly, $\tilde{H}$ can be thought of as the square root of H. Our algorithms will require evolving with $\tilde{H}$ for arbitrary time:

Definition 1. *Let $\tilde{W}(t) := \exp(-i\tilde{H}t)$ be the evolution operator of $\tilde{H}$ for time t, and $\epsilon > 0$ a precision parameter. We define W as a quantum circuit that satisfies $\|\tilde{W}(t) - W\| \leq \epsilon$. The number of two-qubit gates to implement W (i.e., the gate complexity) is $C_W(t, \epsilon)$.*

When H is a physical Hamiltonian described by local operators, $\tilde{H}$ may be efficiently obtained with some classical preprocessing. To obtain $C_W(t, \epsilon)$ in some instances, we note that the results in [21, 24] provide an efficient method for simulating Hamiltonians of complexity polylogarithmic in $1/\epsilon$. In more detail, we could assume that we have a presentation of the Hamiltonian as

$$H = \sum_{k=1}^{K} \alpha_k \Pi_k \ , \tag{5}$$

or

$$H = \frac{1}{2}\sum_{k=1}^{K} \alpha_k U_k \ , \tag{6}$$

where the coefficients satisfy $\alpha_k > 0$. The operators Π_k are projectors (i.e., $(\Pi_k)^2 = \Pi_k$) and U_k are unitaries of eigenvalues ± 1 in this case. Many qubit Hamiltonians can be represented in this way, where the U_k correspond, for example, to Pauli operators. We note that Eq. (6) can be reduced to Eq. (5) by a simple rescaling in which $\Pi_k = (U_k + \mathbb{1}_N)/2$ and disregarding the factor proportional to the $N \times N$ identity operator $\mathbb{1}_N$. In either case, we assume that there is a mechanism available to simulate Π_k or U_k; that is, we assume access to a unitary

$$\begin{aligned} Q &= -\sum_{k=1}^{K} e^{i\pi\Pi_k} \otimes \left|k\right\rangle\!\left\langle k\right|_{\mathrm{a}_2} \\ &= \sum_{k=1}^{K} U_k \otimes \left|k\right\rangle\!\left\langle k\right|_{\mathrm{a}_2} \ , \end{aligned} \tag{7}$$

where a_2 is also an ancillary register of $O(\log(K))$ qubits. The gate complexity of each U_k is C_U, which depends on the problem, and the gate complexity of the conditional U_k operation is $O(C_U \log(K))$.

Once the Hamiltonian H has been reduced to the form of Eq. (5), we obtain

$$\tilde{H} = \sum_{k=1}^{K} \sqrt{\alpha_k}\Pi_k \otimes \left(\left|k\right\rangle\!\left\langle 0\right|_{\mathrm{a}_1} + \left|0\right\rangle\!\left\langle k\right|_{\mathrm{a}_1}\right) \ . \tag{8}$$

To be able to use the results in [24] for simulating $\tilde{H}$ in this case, we note that

$$|k\rangle\langle 0|_{\mathrm{a}_1} + |0\rangle\langle k|_{\mathrm{a}_1} = \frac{i}{2}\left[e^{-i(\pi/2)(|k\rangle\langle 0|_{\mathrm{a}_1}+|0\rangle\langle k|_{\mathrm{a}_1})} - e^{i(\pi/2)(|k\rangle\langle 0|_{\mathrm{a}_1}+|0\rangle\langle k|_{\mathrm{a}_1})}\right] . \quad (9)$$

This provides a decomposition of $\tilde{H}$ as a linear combination of $\tilde{K} = O(K)$ unitary operations $\tilde{U}_k$; that is,

$$\tilde{H} = \sum_{k=1}^{\tilde{K}} \tilde{\alpha}_k \tilde{U}_k \ , \quad (10)$$

and $\tilde{\alpha}_k > 0$. The unitaries in the right hand side of Eq. (9) can be implemented with $O(\log(K))$ two-qubit gates using standard techniques. The algorithm in [24] assumes the ability to implement the unitary

$$\tilde{Q} = \sum_{k=1}^{\tilde{K}} \tilde{U}_k \otimes |k\rangle\langle k|_{\mathrm{a}_2} \ . \quad (11)$$

Since the unitaries $\tilde{U}_k$ are directly related to the U_k, $\tilde{Q}$ can be simulated with $O(1)$ uses of Q and additional two-qubit gates that do not contribute significantly to the final gate complexity.

The query complexity of the method in [24] is determined by the number of uses of $\tilde{Q}$ to implement an approximation of $\tilde{W}(t)$. The gate complexity stated in [24] is the number of additional two-qubit gates required. Then, the results in [24] provide a Hamiltonian simulation method W to approximate $\tilde{W}(t)$ for this case, within precision ϵ, of query complexity

$$O\left(\tau \log(\tau/\epsilon)/\log\log(\tau/\epsilon)\right) \ . \quad (12)$$

Here, $\tau = |t|\sum_k \tilde{\alpha}_k$ and thus $\tau = O(|t|\sum_k \sqrt{\alpha_k})$. The additional gate complexity of W obtained in [24] for this case is

$$O\left(K\tau \log(\tau/\epsilon)/\log\log(\tau/\epsilon)\right) \ . \quad (13)$$

These results also imply that the overall gate complexity of W is

$$C_W(t,\epsilon) = O\left((\log(K)C_U + K)\tau \frac{\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)}\right) \ . \quad (14)$$

We refer to [24] for more details.

In general, our quantum algorithm to sample from Gibbs distributions provides an exponential improvement in terms of $1/\epsilon$, with respect to other known algorithms [8, 9, 11], whenever $C_W(t,\epsilon)$ is polylogarithmic in $1/\epsilon$. As discussed, this is the case for a large class of Hamiltonians such as those when the U_k are presented as Pauli operators, so that $C_U = O(n)$.

For the quantum algorithm that computes an estimate of the hitting time of a Markov chain, we will assume that we have query access to the Hamiltonian H, and that H can be presented as in Eq. (5). This assumes the existence of a procedure that outputs the matrix elements of H. Constructing a quantum circuit W that approximates the evolution with $\tilde{H}$ in this case is technically involved and we leave that analysis for Appx. A. As in the previous case, we use the methods in [21, 24] to show that $C_W(t,\epsilon)$ is almost linear in $|t|$ and sublogarithmic in $|t|/\epsilon$.

Another useful technique for our quantum algorithms, also used in [20, 22, 24, 26], regards the implementation of linear combinations of unitary operations. More specifically, assume that $X = \sum_{l=0}^{L-1} \gamma_l V_l$, where $\gamma_l > 0$ and V_l are unitary operations, and that there is a mechanism to implement V_l. That is, we have access to the unitary

$$R = \sum_{l=0}^{L-1} V_l \otimes |l\rangle\langle l|_{\mathrm{a}_3} \ , \quad (15)$$

where a_3 is an ancillary register of $O(\log(L))$ qubits. Lemma 6 of [22] implies that we can prepare a normalized version of the state $X|\phi\rangle$ with $O(\gamma/\|X|\phi\rangle\|)$ uses of R in addition to $O(L\gamma/\|X|\phi\rangle\|)$ two-qubit gates, where $\gamma = \sum_{l=0}^{L-1}\gamma_l$. When $V_l = V^l$, for some unitary V, and the gate complexity of V is C_V, the gate complexity of R is $O(LC_V)$. In this case, the overall gate complexity of the algorithm is $O(LC_V\gamma/\|X|\phi\rangle\|)$. This result follows from Lemma 8 of [22]. The implication is that the overall gate complexity is dominated by the largest gate complexity of the unitaries in R times the number of amplitude amplification steps.

For completeness, the quantum algorithm to implement X is built upon $O(\gamma/\|X|\phi\rangle\|)$ amplitude amplification steps [27]. The operation for state preparation starts by preparing the ancillary state

$$B|0\rangle_{\mathrm{a}_3} = \frac{1}{\sqrt{\gamma}} \sum_{l=0}^{L-1} \sqrt{\gamma_l}\,|l\rangle_{\mathrm{a}_3} \ , \quad (16)$$

where B is unitary. Applying B requires $O(L)$ two-qubit gates and, in those cases where we can exploit the structure of the coefficients γ_l, it can be done more efficiently. The state preparation step then applies R followed by $B^\dagger$. One can show that the final state of this step is

$$\left(\frac{X}{\gamma}|\phi\rangle\right) \otimes |0\rangle_{\mathrm{a}_3} + |\Theta^\perp\rangle \ , \quad (17)$$

where $|\Theta^\perp\rangle$ is supported in the subspace orthogonal to $|0\rangle_{\mathrm{a}_3}$. Amplitude amplification allows us to amplify the probability of observing the state $|0\rangle_{\mathrm{a}_3}$ to a constant. This state corresponds to the desired outcome. The number of amplitude amplification steps is linear in the inverse of $\|(X/\gamma)|\phi\rangle\|$.

The third useful technique regards amplitude estimation [18]. Let T be a unitary that implements

$$T|\phi\rangle|0\rangle = (A|\phi\rangle)|0\rangle + |\Phi^\perp\rangle|1\rangle \ , \quad (18)$$

where A is an operator that satisfies $\|A\| \leq 1$ and $\| \left|\Phi^{\perp}\right\rangle \| \leq 1$. Our goal is to obtain an estimate of $\langle\phi| A |\phi\rangle = \langle\phi| \langle 0| T |\phi\rangle |0\rangle$. The results in [18] imply that there exists a quantum algorithm that outputs an estimate of the expectation value of T within precision ϵ. For constant confidence level ($c \approx 0.81$), the quantum algorithm uses T and other two-qubit gates $O(1/\epsilon)$ times. It also uses the unitary that prepares the initial state $|\phi\rangle$, $O(1/\epsilon)$ times. Increasing the confidence level can be done with an additional overhead that is logarithmic in $|1-c|$.

III. PREPARATION OF GIBBS STATES

The thermal Gibbs state of a quantum system H at inverse temperature $\beta \geq 0$ is the density matrix

$$\rho = \frac{1}{\mathcal{Z}} e^{-\beta H} , \tag{19}$$

where $\mathcal{Z} = \mathrm{Tr}[e^{-\beta H}] = \sum_j e^{-\beta E_j}$ is the partition function. Then, the probability of encountering the system in the quantum state $|\psi_j\rangle$, after measurement, is $p_j = e^{-\beta E_j}/\mathcal{Z}$.

Given a precision parameter $\epsilon > 0$, a quantum algorithm to sample from the Gibbs distribution p_j can be obtained from a unitary $\bar{V}$ that satisfies

$$\mathrm{Tr}_{\mathrm{a}} \left[\bar{V} \left(|0\rangle\langle 0| \otimes |0\rangle\langle 0|_{\mathrm{a}} \right) \bar{V}^{\dagger} \right] = \hat{\rho} \tag{20}$$

and

$$\frac{1}{2} \|\hat{\rho} - \rho\|_1 \leq \epsilon . \tag{21}$$

We use the label **a** for an ancillary qubit system that will be discarded at the end of the computation. The dimension of **a** depends on the algorithm. The requirement on the trace distance in Eq. (21) implies that no measurement can distinguish between ρ and $\hat{\rho}$ with probability greater than ϵ [28].

The main result of this section is:

Theorem 1. *There exists a quantum algorithm that prepares an approximation of the Gibbs state. The quantum algorithm implements a unitary $\bar{V}$ of gate complexity*

$$O\left(\sqrt{\frac{N}{\mathcal{Z}}} \left(C_W(t,\epsilon') + n + \log(J) \right) \right) , \tag{22}$$

with $t = O(\sqrt{\beta \log(1/\epsilon')})$, $\epsilon' = O(\epsilon \sqrt{\mathcal{Z}/N})$, and $J = O(\sqrt{\|H\|\beta} \log(1/\epsilon'))$.

When H is presented as in Eq. (5), we can replace $C_W(t,\epsilon')$ by Eq. (14) if we use the best-known Hamiltonian simulation algorithm. In cases of interest, such as qubit systems given by Hamiltonians that are linear combinations of Pauli operators, we have $\alpha_k = O(1)$, $C_U = O(n) = O(\log(N))$, and $K = O(n) = O(\log(N))$. In this case, the overall gate complexity is

$$O\left(\sqrt{\frac{N\beta}{\mathcal{Z}}} \mathrm{polylog}\left(\sqrt{\frac{N\beta}{\mathcal{Z}}} \frac{1}{\epsilon} \right) \right) . \tag{23}$$

The important result is that the complexity of our algorithm is polylogarithmic in $1/\epsilon$ and also improves upon the complexity in β with respect to the methods in [8, 9].

Our quantum algorithm to sample from Gibbs distributions uses the two techniques discussed in Sec. II. In this case, we will be interested in implementing an operator proportional to $e^{-\beta H/2}$. To find a decomposition as a linear combination of unitaries, we invoke the so-called Hubbard-Stratonovich transformation [29]:

$$e^{-\beta H/2} = \sqrt{\frac{1}{2\pi}} \int_{-\infty}^{\infty} dy \; e^{-y^2/2} e^{-iy\sqrt{\beta H}} . \tag{24}$$

In our case, we do not have a method to simulate the evolution with $\sqrt{H}$. Nevertheless, we assume that we can evolve with $\tilde{H}$, which satisfies Eq. (4). Then,

$$\begin{aligned} (e^{-\beta H/2} |\phi\rangle) \otimes |0\rangle_{\mathrm{a}_1} = \\ = \left(\sqrt{\frac{1}{2\pi}} \int_{-\infty}^{\infty} dy \; e^{-y^2/2} e^{-iy\sqrt{\beta}\tilde{H}} \right) |\phi\rangle |0\rangle_{\mathrm{a}_1} , \end{aligned} \tag{25}$$

for any state $|\phi\rangle$. Note that the ancilla a_1 remains in the state $|0\rangle_{\mathrm{a}_1}$ and will be discarded at the end of the computation. Equation (25) implies that the operator $e^{-\beta H/2}$ can be approximated by a linear combination of evolutions under $\tilde{H}$. Because $y \in (-\infty, \infty)$, we will need to find an approximation by a finite, discrete sum of operators $e^{-iy_j\sqrt{\beta}\tilde{H}}$. We obtain:

Lemma 1. *Let*

$$X' = \sqrt{\frac{1}{2\pi}} \sum_{j=-J}^{J} \delta y \; e^{-y_j^2/2} e^{-iy_j\sqrt{\beta}\tilde{H}} , \tag{26}$$

where $y_j = j\delta y$, for some $J = \Theta(\sqrt{\|H\|\beta} \log(1/\epsilon'))$ and $\delta y = \Theta(1/\sqrt{\|H\|\beta \log(1/\epsilon')})$. Then, if $\|H\|\beta \geq 4$ and $\log(1/\epsilon') \geq 4$,

$$\|(e^{-\beta H/2} |\phi\rangle) \otimes |0\rangle_{\mathrm{a}_1} - X' |\phi\rangle |0\rangle_{\mathrm{a}_1} \| \leq \epsilon'/2 , \tag{27}$$

for all states $|\phi\rangle$.

Proof. Consider the real function

$$f(\tilde{x}) = \frac{1}{\sqrt{2\pi}} \sum_{j=-\infty}^{\infty} \delta y \; e^{-y_j^2/2} e^{-iy_j\sqrt{\beta}\tilde{x}} \tag{28}$$

where $\tilde{x} \in \mathbb{R}$ and assume $|\tilde{x}| \leq a < \infty$. The Poisson summation formula and the Fourier transform of the Gaussian imply

$$f(\tilde{x}) = \sum_{k=-\infty}^{\infty} e^{-\omega_k^2/2} , \tag{29}$$

where $\omega_k = -\sqrt{\beta}\tilde{x} + k/\delta y$. Then, there exists

$$\delta y = \Omega\left(\frac{1}{a\sqrt{\beta} + \sqrt{\log(1/\epsilon')}}\right) \tag{30}$$

such that $|f(\tilde{x}) - e^{-\omega_0^2/2}| \leq \epsilon'/4$. Note that if $a\sqrt{\beta} \geq 2$ and $\sqrt{\log(1/\epsilon')} \geq 2$, we can choose

$$\delta y = \Theta\left(\frac{1}{a\sqrt{\beta\log(1/\epsilon')}}\right) . \tag{31}$$

Also,

$$\begin{aligned}\left|\frac{1}{\sqrt{2\pi}}\sum_{j=J}^{\infty}\delta y\, e^{-y_j^2/2}e^{-iy_j\sqrt{\beta}\tilde{x}}\right| &\leq \frac{\delta y}{\sqrt{2\pi}}\sum_{j=J}^{\infty}e^{-y_j^2/2} \\ &\leq \frac{\delta y}{\sqrt{2\pi}}\sum_{j=J}^{\infty}e^{-y_J y_j/2} \\ &\leq \frac{\delta y}{\sqrt{2\pi}}\frac{e^{-y_j^2/2}}{1-e^{-y_J\delta y/2}} .\end{aligned}$$

It follows that there exists a value for J, which implies $y_J = \Theta(\sqrt{\log(1/\epsilon')})$, such that

$$\left|f(\tilde{x}) - \frac{1}{\sqrt{2\pi}}\sum_{j=-J}^{J}\delta y\, e^{-y_j^2/2}e^{-iy_j\sqrt{\beta}\tilde{x}}\right| \leq \epsilon'/4 . \tag{32}$$

Using the triangle inequality we obtain

$$\left|e^{-\omega_0^2/2} - \frac{1}{\sqrt{2\pi}}\sum_{j=-J}^{J}\delta y\, e^{-y_j^2/2}e^{-iy_j\sqrt{\beta}\tilde{x}}\right| \leq \epsilon'/2 , \tag{33}$$

and we can represent

$$e^{-\omega_0^2/2} = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}dy\, e^{-y^2/2}e^{-iy\sqrt{\beta}\tilde{x}} . \tag{34}$$

To prove the Lemma, it suffices to act with X' on the eigenstates of $\tilde{H}$. We can then use the previous bounds if we assume that $\tilde{x}$ denotes the corresponding eigenvalue of $\tilde{H}$. In particular, $a = \|\tilde{H}\|$. Then, if $\|\tilde{H}\|\sqrt{\beta} \geq 2$ and $\sqrt{\log(1/\epsilon')} \geq 2$, it suffices to choose

$$\delta y = \Theta\left(\frac{1}{\|\tilde{H}\|\sqrt{\beta\log(1/\epsilon')}}\right) \tag{35}$$

and

$$J = \frac{y_J}{\delta y} = \Theta\left(\log(1/\epsilon')\|\tilde{H}\|\sqrt{\beta}\right) . \tag{36}$$

The result follows from noticing that $\|\tilde{H}\| = O(\sqrt{\|H\|})$ and that X' then approximates $e^{-\beta(\tilde{H})^2/2}$. Since we act on initial states of the form $|\phi\rangle|0\rangle_{\mathrm{a}_1}$, the action of $e^{-\beta(\tilde{H})^2/2}$ is the same as that of $e^{-\beta H/2}$ on these states. □

In general, we cannot implement the unitaries $e^{-iy_j\sqrt{\beta}\tilde{H}}$ exactly but we can do so up to an approximation error. We obtain:

Corollary 1. *Let* $\log(1/\epsilon') \geq 4$, $\|H\|\beta \geq 4$, *and* W_j *be a unitary that satisfies*

$$\|W_j - e^{-iy_j\sqrt{\beta}\tilde{H}}\| \leq \epsilon'/4 \tag{37}$$

for all $j = -J, -J+1, \ldots, J$. *Let*

$$X = \sqrt{\frac{1}{2\pi}}\sum_{j=-J}^{J}\delta y\, e^{-y_j^2/2}W_j . \tag{38}$$

Then,

$$\|(e^{-\beta H/2}|\phi\rangle)\otimes|0\rangle_{\mathrm{a}_1} - X|\phi\rangle|0\rangle_{\mathrm{a}_1}\| \leq \epsilon' . \tag{39}$$

Proof. The coefficients in the decomposition of X' in Lemma 1 satisfy

$$\left|\sqrt{\frac{1}{2\pi}}\sum_{j=-J}^{J}\delta y\, e^{-y_j^2/2} - 1\right| \leq \epsilon'/2 \leq 1/4 , \tag{40}$$

and thus

$$\left|\sqrt{\frac{1}{2\pi}}\sum_{j=-J}^{J}\delta y\, e^{-y_j^2/2}\right| \leq 5/4 . \tag{41}$$

This follows from Eq. (27) for the case of $H = \tilde{H} = 0$. The triangle inequality and Eq. (37) imply

$$\|X - X'\| \leq (5/16)\epsilon' , \tag{42}$$

and together with Eq. (27) we obtain the desired result. □

In Sec. II we described a technique to implement $X = \sum_{l=0}^{L-1}\gamma_l V_l$. In this case, $l = j + J$ and $L = 2J + 1$. The coefficients and unitaries are $e^{-y_j^2/2}$ and W_j, respectively. The quantum algorithm for preparing Gibbs states will aim at preparing a normalized version of $X|\phi_0\rangle$, for a suitable initial state $|\phi_0\rangle$, using the technique of Sec. II.

A. Algorithm

We set $\epsilon' = O(\epsilon\sqrt{\mathcal{Z}/N})$. Our quantum circuit $\bar{V}$ is defined in two basic steps. The first step regards the preparation of a maximally entangled state

$$|\phi_0\rangle = \frac{1}{\sqrt{N}}\sum_{j=0}^{N-1}|\psi_j\rangle\otimes|\psi_j^*\rangle_{\mathrm{a}_4}\otimes|0\rangle_{\mathrm{a}_1,\mathrm{a}_2,\mathrm{a}_3} \tag{43}$$

where we used an additional ancillary system a_4 of n qubits. a_1, a_2, a_3, and a_4 build the ancillary register **a**

of Eq. (20). Note that $|\phi_0\rangle$ coincides with the maximally entangled state

$$|\phi_0\rangle = \frac{1}{\sqrt{N}} \sum_{\sigma=0}^{N-1} |\sigma\rangle\, |\sigma\rangle_{\mathrm{a}_4} \otimes |0\rangle_{\mathrm{a}_1,\mathrm{a}_2,\mathrm{a}_3} \ , \qquad (44)$$

where $|\sigma\rangle$ is a n-qubit state in the computational basis, i.e., $|\sigma\rangle = |0\ldots0\rangle, |0\ldots1\rangle, \ldots$.

The second step regards the preparation of a normalized version of $X|\phi_0\rangle$. This step uses the algorithm for implementing linear combinations of unitary operations described in Sec. II, which also uses amplitude amplification. The operator X is defined in Eq. (38) and requires a Hamiltonian simulation method for implementing W_j.

B. Validity and complexity

As described, our quantum algorithm prepares the normalized state

$$\frac{X|\phi_0\rangle}{\|X|\phi_0\rangle\|} \ , \qquad (45)$$

with constant probability. We also note that

$$\|e^{-\beta H/2}|\phi_0\rangle\| = \sqrt{\mathcal{Z}/N} \qquad (46)$$

and Eq. (39) implies

$$\left| \|X|\phi_0\rangle\| - \sqrt{\mathcal{Z}/N} \right| = O(\epsilon\sqrt{\mathcal{Z}/N}) \qquad (47)$$

for our choice of ϵ'. Then, the prepared state satisfies

$$\left\| \frac{X|\phi_0\rangle}{\|X|\phi_0\rangle\|} - \frac{e^{-\beta H/2}|\phi_0\rangle}{\|e^{-\beta H/2}|\phi_0\rangle\|} \right\| \leq \epsilon/2 \ . \qquad (48)$$

We note that

$$e^{-\beta H/2}|\phi_0\rangle = \\ = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{-\beta E_j/2} |\psi_j\rangle \otimes |\psi_j^*\rangle_{\mathrm{a}_4} \otimes |0\rangle_{\mathrm{a}_1,\mathrm{a}_2,\mathrm{a}_3} \ . \qquad (49)$$

If we disregard the ancillary system **a**, this is the Gibbs state: The probability of obtaining $|\psi_j\rangle$, after measurement, is proportional to $e^{-\beta E_j}$. Then, the property of the trace norm being non-increasing under quantum operations and Eq. (48) imply

$$\frac{1}{2}\|\hat{\rho} - \rho\| \leq \epsilon \ , \qquad (50)$$

where

$$\hat{\rho} = \mathrm{Tr}_{\mathrm{a}} \left[\frac{X|\phi_0\rangle\langle\phi_0|X^\dagger}{\|X|\phi_0\rangle\|^2} \right] \ ; \qquad (51)$$

see Eq. (20). That is, $\hat{\rho}$ is the state prepared by our algorithm after tracing out the ancillary register **a**.

The number of amplitude amplification steps is $O(1/\|X|\phi_0\rangle\|)$ and Eq. (46) implies that this number is also $O(\sqrt{N/\mathcal{Z}})$. The gate complexity of each step is the gate complexity of preparing $|\phi_0\rangle$ in addition to the gate complexity of implementing X. The former is $O(n)$ as $|\phi_0\rangle$ takes the simple form of Eq. (44) and can be prepared with $O(n)$ controlled operations. X is implemented in three stages as described in Sec. II. The first stage requires the unitary B used in Eq. (16). In this case, the coefficients γ_l are proportional to $e^{-y_j^2/2}$. Then, the gate complexity of B is $O(\log(J))$ in this case if we use one of the methods developed in [30, 31]. The second stage regards the implementation of R. In this example, R is the unitary that implements W_j conditional on the state $|j\rangle_{\mathrm{a}_3}$. Since W_j corresponds to a Hamiltonian simulation algorithm that approximates evolutions with $\tilde{H}$, the gate complexity of R is dominated by the largest gate complexity of W_j. In particular, W_j approximates $e^{-i\tilde{H}t}$ within precision ϵ' and for maximum $t = O(\sqrt{\beta\log(1/\epsilon')})$. Then the gate complexity of R is order $C_W(t,\epsilon')$.

The overall gate complexity is then

$$O\left(\sqrt{\frac{N}{\mathcal{Z}}} \left(C_W(t,\epsilon') + n + \log(J) \right) \right) \ , \qquad (52)$$

with $t = O(\sqrt{\beta\log(1/\epsilon')})$ and $\epsilon' = O(\epsilon\sqrt{\mathcal{Z}/N})$. This proves Thm. 1.

IV. ESTIMATION OF HITTING TIMES

We consider a stochastic process that models a Markov chain. The number of different configurations is N and P is the $N \times N$ stochastic matrix. We label each configuration as $\sigma = 0, 1, \ldots, N-1$ and the entries of P are transition or conditional probabilities $\Pr(\sigma'|\sigma)$. We will assume that P is reversible and irreducible, satisfies the so-called detailed balance condition [5], and has nonnegative eigenvalues. The unique fixed point of P is the N-dimensional probability vector π. It is useful to use the bra-ket notation, where $|\nu\rangle$ represents a vector $\nu \in \mathbb{C}^N$ and $\langle v| = (|v\rangle)^\dagger$. Then, $P|\pi\rangle = |\pi\rangle$,

$$|\pi\rangle = \sum_\sigma \pi_\sigma |\sigma\rangle \doteq \begin{pmatrix} \pi_0 \\ \vdots \\ \pi_{N-1} \end{pmatrix} \ , \qquad (53)$$

and π_σ is the probability of finding configuration σ when sampling from the fixed point of P.

The hitting time of a stochastic process is roughly defined as the first time at which the process is encountered in a particular subset of configurations. To define the hitting time in detail, we assume that there is a subset $\mathcal{M}$ of $N_{\mathcal{M}}$ configurations that are "marked" and the remaining $N_{\mathcal{U}}$ configurations constitute the "unmarked" subset $\mathcal{U}$.

Here, $N_{\mathcal{M}} + N_{\mathcal{U}} = N$. With no loss of generality, the stochastic matrix P takes the form

$$P = \begin{pmatrix} P_{\mathcal{U}\mathcal{U}} & P_{\mathcal{U}\mathcal{M}} \\ P_{\mathcal{M}\mathcal{U}} & P_{\mathcal{M}\mathcal{M}} \end{pmatrix} , \tag{54}$$

where $P_{\mathcal{U}\mathcal{U}}$ and $P_{\mathcal{M}\mathcal{M}}$ are matrices (blocks) of dimension $N_{\mathcal{U}} \times N_{\mathcal{U}}$ and $N_{\mathcal{M}} \times N_{\mathcal{M}}$, respectively, and $P_{\mathcal{M}\mathcal{U}}$ and $P_{\mathcal{U}\mathcal{M}}$ are rectangular blocks. The entries of the block $P_{\mathcal{S}'\mathcal{S}}$ determine the probability of a configuration being in the subset $\mathcal{S}'$ given that the previous configuration was in the subset $\mathcal{S}$. Our assumptions imply $\mathcal{U}, \mathcal{M} \neq \{\emptyset\}$ and $P_{\mathcal{U}\mathcal{M}}, P_{\mathcal{M}\mathcal{U}} \neq 0$. The hitting time is the expected time to find a marked configuration if the initial probability vector is $|\pi\rangle$. That is, as in [17], we define the hitting time of P via the following classical algorithm:

1. Set $t = 0$
2. Sample σ from $|\pi\rangle$
3. If $\sigma \in \mathcal{M}$, stop
4. Otherwise, assign $t \leftarrow t + 1$, apply P, and go to 3.

The hitting time t_h is the expected value of the random variable t.

We let $|\pi_{\mathcal{U}}\rangle$ and $|\pi_{\mathcal{M}}\rangle$ represent the probability vectors obtained by conditioning $|\pi\rangle$ on $\mathcal{U}$ and $\mathcal{M}$, respectively. These are

$$|\pi_{\mathcal{U}}\rangle = \frac{\sum_{\sigma\in\mathcal{U}} \pi(\sigma)\,|\sigma\rangle}{\pi_{\mathcal{U}}} , \; |\pi_{\mathcal{M}}\rangle = \frac{\sum_{\sigma\in\mathcal{M}} \pi(\sigma)\,|\sigma\rangle}{\pi_{\mathcal{M}}} , \tag{55}$$

with $\pi_{\mathcal{U}} = \sum_{\sigma\in\mathcal{U}} \pi(\sigma)$ and $\pi_{\mathcal{M}} = \sum_{\sigma\in\mathcal{M}} \pi(\sigma)$. It is useful to define the modified Markov chain

$$P' = \begin{pmatrix} P_{\mathcal{U}\mathcal{U}} & 0 \\ P_{\mathcal{M}\mathcal{U}} & \mathbb{1}_{N_{\mathcal{M}}} \end{pmatrix} , \tag{56}$$

which refers to an "absorbing wall" for the subset $\mathcal{M}$. Here, $\mathbb{1}_{N_{\mathcal{M}}}$ is the $N_{\mathcal{M}} \times N_{\mathcal{M}}$ identity matrix. As defined, P' does not allow for transitions from the subset $\mathcal{M}$ to the subset $\mathcal{U}$. We will observe below that P', and thus $P_{\mathcal{U}\mathcal{U}}$, play an important role in the determination of t_h.

Our definition of hitting time implies

$$t_h = \sum_{t=0}^{\infty} t\,\Pr(t) , \tag{57}$$

where $\Pr(t)$ is the probability of t if we use the previous classical algorithm. In particular, $\Pr(t = 0) = \pi_{\mathcal{M}}$. We rewrite

$$t_h = \sum_{t'=0}^{\infty} \Pr(t > t') \tag{58}$$

so that we take into account the factor t in Eq. (57), i.e., $\Pr(t > t') = \Pr(t' + 1) + \Pr(t' + 2) + \ldots$. Note that

$$\begin{aligned} \Pr(t > t') &= \pi_{\mathcal{U}} \langle 1_{\mathcal{U}}| \left(P'\right)^{t'} |\pi_{\mathcal{U}}\rangle \\ &= \pi_{\mathcal{U}} \langle 1_{\mathcal{U}}| \left(P_{\mathcal{U}\mathcal{U}}\right)^{t'} |\pi_{\mathcal{U}}\rangle , \end{aligned} \tag{59}$$

where $|1_{\mathcal{U}}\rangle = \sum_{\sigma\in\mathcal{U}} |\sigma\rangle$. This is because, conditional on $t > 0$, which occurs with probability $\pi_{\mathcal{U}}$, the initial probability vector is proportional to $\left(P_{\mathcal{U}\mathcal{U}}\right)^{t'} |\pi_{\mathcal{U}}\rangle$. Then, the probability of having $t > t'$ is measured by the probability of remaining in $\mathcal{U}$ after $P_{\mathcal{U}\mathcal{U}}$ was applied t' times. Equations (58) and (59) imply

$$t_h = \pi_{\mathcal{U}} \langle 1_{\mathcal{U}}| \left(\mathbb{1}_{N_{\mathcal{U}}} - P_{\mathcal{U}\mathcal{U}}\right)^{-1} |\pi_{\mathcal{U}}\rangle , \tag{60}$$

where we used $(1 - x)^{-1} = \sum_{t'=0}^{\infty} x^{t'}$. We note that $\mathbb{1} - P_{\mathcal{U}\mathcal{U}}$ is invertible under our assumptions, since the eigenvalues of $P_{\mathcal{U}\mathcal{U}}$ are strictly smaller than 1 (see below).

The complexity of a method that estimates t_h using the previous classical algorithm also depends on the variance of the random variable t. This is

$$\sigma^2 = \sum_{t=0}^{\infty} t^2 \Pr(t) - (t_h)^2 , \tag{61}$$

and after simple calculations, we can rewrite it as

$$\sigma^2 = 2\pi_{\mathcal{U}} \langle 1_{\mathcal{U}}| \sum_{t'=0}^{\infty} t' \left(P_{\mathcal{U}\mathcal{U}}\right)^{t'} |\pi_{\mathcal{U}}\rangle + t_h - (t_h)^2 . \tag{62}$$

For constant confidence level and precision ϵ in the estimation of t_h, Chebyshev's inequality implies that the previous classical algorithm must be executed $M = O((\sigma/\epsilon)^2)$ times to obtain $t_1, \ldots, t_M$ and estimate t_h as the average of the t_i. The expected number of applications of P is then $Mt_h = O(t_h(\sigma/\epsilon)^2)$.

To bound the classical complexity, we consider the worst case scenario in which $|\pi_{\mathcal{U}}\rangle$ is an eigenvector of $P_{\mathcal{U}\mathcal{U}}$ corresponding to its largest eigenvalue $1 - \Delta < 1$. In that case, $t_h = \pi_{\mathcal{U}}/\Delta$ and $\sigma^2 = O(\pi_{\mathcal{U}}/\Delta^2)$. When $\Delta \ll 1$, the expected number of applications of P is then $O(1/(\Delta^3\epsilon^2))$ in this case. This determines the average complexity of the classical algorithm that estimates t_h.

The entries of the symmetric discriminant matrix S of P are

$$S_{\sigma\sigma'} = \sqrt{\left(P \circ P^{\dagger}\right)_{\sigma\sigma'}} , \tag{63}$$

where $\circ$ is the Hadamard product. The detailed balance condition implies

$$\pi(\sigma')\Pr(\sigma|\sigma') = \Pr(\sigma'|\sigma)\pi(\sigma) \tag{64}$$

and thus

$$\sqrt{\Pr(\sigma|\sigma')\Pr(\sigma'|\sigma)} = \Pr(\sigma|\sigma')\sqrt{\frac{\pi(\sigma')}{\pi(\sigma)}} . \tag{65}$$

Then,

$$S = D^{-1}PD , \tag{66}$$

where D is a diagonal matrix of dimension N with entries given by $\sqrt{\pi(\sigma)}$. The symmetric matrix or Hamiltonian $\bar{H} = \mathbb{1}_N - S$ is known to be "frustration free" [32] and can

be represented as in Eq. (5) using a number of techniques. For example, if P has at most d nonzero entries per row or column (i.e., P is d-sparse), the number of terms K in the representation of $\bar{H}$ can be made linear or quadratic in d; see Appx. A or [33] for more details.

We now let $\Pi_{\mathcal{U}}$ be the projector into the subset $\mathcal{U}$ and define

$$H = \Pi_{\mathcal{U}} \bar{H} \Pi_{\mathcal{U}} \ . \tag{67}$$

Note that

$$H = \mathbb{1}_{N_{\mathcal{U}}} - D_{\mathcal{U}}^{-1} P_{\mathcal{U}\mathcal{U}} D_{\mathcal{U}} \ , \tag{68}$$

where $D_{\mathcal{U}}$ is the diagonal matrix obtained by projecting D of Eq. (66) into the subspace $\mathcal{U}$. That is, $D_{\mathcal{U}} = \Pi_{\mathcal{U}} D \Pi_{\mathcal{U}}$ and Eq. (68) implies $H > 0$. Then, Eqs. (66) and (60) imply

$$t_h = \pi_{\mathcal{U}} \langle \sqrt{\pi_{\mathcal{U}}} | \, (1/H) | \sqrt{\pi_{\mathcal{U}}} \rangle \ , \tag{69}$$

where we defined $|\sqrt{\pi_{\mathcal{U}}}\rangle = \sum_{\sigma \in \mathcal{U}} \sqrt{\pi(\sigma)} \, |\sigma\rangle \, / \sqrt{\pi_{\mathcal{U}}}$ so that $|\sqrt{\pi_{\mathcal{U}}}\rangle$ is normalized according to the Euclidean norm. A similar expression for t_h was obtained in [17].

In Appx. A we describe how H can be specified as $H = \sum_{k=1}^{K} \alpha_k \Pi_k$, where $\alpha_k > 0$ and Π_k are projectors. Then H is of the form of Eq. (5) and we write $\tilde{H}$ for the associated Hamiltonian according to Eq. (3). $C_W(t, \epsilon)$ is the complexity of approximating $\tilde{W}(t) = \exp(-i\tilde{H}t)$, and we roughly describe a method for simulating $\tilde{W}(t)$ below.

A quantum algorithm to obtain t_h can be constructed from the relation in Eq. (69). That is, t_h coincides with $\pi_{\mathcal{U}}$ times the expected value of the operator $1/H$ in the pure state $\left|\sqrt{\pi_{\mathcal{U}}}\right\rangle$. For our quantum algorithm, we also assume that there is a unitary procedure (oracle) $Q_{\mathcal{U}}$ that allows us to implement the transformation

$$Q_{\mathcal{U}} \, |\sigma\rangle = - \, |\sigma\rangle \ \text{ if } \sigma \in \mathcal{U} \tag{70}$$

and $Q_{\mathcal{U}} \, |\sigma\rangle = |\sigma\rangle$ otherwise. We also assume access to a unitary $Q_{\sqrt{\pi}}$ such that

$$Q_{\sqrt{\pi}} \, |0\rangle = \left|\sqrt{\pi}\right\rangle \ . \tag{71}$$

We write $C_{\mathcal{U}}$ and $C_{\sqrt{\pi}}$ for the respective gate complexities. The main result of this section is:

Theorem 2. *There exists a quantum algorithm to estimate t_h within precision ϵ and constant confidence level that implements a unitary $\bar{V}$ of gate complexity*

$$O\left(\frac{1}{\epsilon'} \left(C_W(t, \epsilon') + C_{\mathcal{U}} + C_{\sqrt{\pi}} + C_B\right)\right) \ , \tag{72}$$

where $C_B = O(\log(1/(\Delta\epsilon)))$, $\epsilon' = O(\epsilon\Delta/\log(1/(\epsilon\Delta)))$, and $t = O\left(\log(1/(\epsilon\Delta))/\sqrt{\Delta}\right)$.

In Appx. A we describe a method to simulate the evolution with $\tilde{H}$. To this end, we also assume that there exists a procedure Q_P that computes the locations and magnitude of the nonzero entries of the matrix P. More specifically, Q_P performs the map

$$\begin{aligned} Q_P \, |\sigma\rangle \to |\sigma\rangle \, |\sigma'_1, \ldots, \sigma'_d\rangle \otimes \\ \otimes \, |\Pr(\sigma|\sigma'_1), \Pr(\sigma'_1|\sigma), \ldots, \Pr(\sigma|\sigma'_d), \Pr(\sigma'_d|\sigma)\rangle \ , \end{aligned} \tag{73}$$

where d is the sparsity of P. The configurations σ'_i are such that $\Pr(\sigma|\sigma'_i), \Pr(\sigma'_i|\sigma) \neq 0$. We write C_P for the complexity of implementing Q_P. In Appx. A we describe a decomposition of $\tilde{H} = \sum_{k=1}^{\tilde{K}} \tilde{U}_k/2$ in terms of $\tilde{K} = O(d^2)$ unitaries, so that we can use the results of [24] to simulate $\tilde{W}(t) = \exp(-i\tilde{H}t)$. Each $\tilde{U}_k$ can be implemented with $O(1)$ uses of $Q_{\mathcal{U}}$ and Q_P, and $O(d \log(N))$ additional gates. Using the results of [24] and Sec. II, the complexity for simulating $\tilde{W}(t)$ within precision ϵ', obtained in Eq. (A24), is

$$C_W(t, \epsilon') = O\left((d \log(N) + C_P + C_{\mathcal{U}}) \frac{\tau \log(\tau/\epsilon')}{\log\log(\tau/\epsilon')}\right) \ , \tag{74}$$

where $\tau = |t| d^2$. Note that $C_W(t, \epsilon')$ is almost linear in $|t|$ and polynomial in d, and the dependence on d may be improved by using the results in [34]. Then, assuming access to Q_P, we obtain:

Corollary 2. *There exists a quantum algorithm to estimate t_h within precision ϵ and constant confidence level that implements a unitary $\bar{V}$ of gate complexity*

$$\tilde{O}\left(\frac{1}{\epsilon\Delta}\left(\frac{d^2}{\sqrt{\Delta}}(d \log(N) + C_P + C_{\mathcal{U}}) + C_{\sqrt{\pi}}\right)\right) \ . \tag{75}$$

The $\tilde{O}$ notation hides factors that are polylogarithmic in $d/(\epsilon\Delta)$.

The dominant scaling of the complexity in terms of Δ and ϵ is then $\tilde{O}(1/(\epsilon\Delta^{3/2}))$, which is a quadratic improvement over the classical complexity obtained above.

Our quantum algorithm uses the three techniques described in Sec. II and uses some other results in [22]. In fact, since $H > 0$, we can improve some results in [22] that regard the decomposition of the inverse of a matrix as a linear combination of unitaries. That is, for a positive matrix H, we can use the identity

$$\frac{1}{H} = \frac{1}{2} \int_0^\infty d\beta \ e^{-\beta H/2} \ , \tag{76}$$

and use the Hubbard-Stratonovich transformation of Eq. (25) to simulate $e^{-\beta H/2}$. Roughly, $1/H$ can be simulated by a linear combination of unitaries, each corresponding to an evolution with $\tilde{H}$ for time $y\sqrt{\beta}$. Since $\beta \in [0, \infty)$ and $y \in (-\infty, \infty)$, we will need to find an approximation by a finite, discrete sum of operators $e^{-iy_j\sqrt{\beta_k}\tilde{H}}$. We obtain:

Lemma 2. *Let*

$$X' = \frac{1}{\sqrt{2\pi}} \sum_{k=0}^{K} \delta z \sum_{j=-J}^{J} \delta y \; e^{-y_j^2/2} e^{-iy_j\sqrt{z_k}\tilde{H}} \; , \qquad (77)$$

where $y_j = j\delta y$, $z_k = k\delta z$, *and* $\|H\| \geq \Delta > 0$. *Then, there exists* $J = \Theta(\sqrt{1/\Delta}\log^{3/2}(1/(\Delta\epsilon)))$, $K = \Theta((1/\Delta)\log(1/(\Delta\epsilon))/\epsilon)$, $\delta y = \Theta(\sqrt{\Delta}/\log(1/(\Delta\epsilon)))$, *and* $\delta z = \Theta(\epsilon)$ *such that*

$$\left\| \left(\frac{1}{H} |\phi\rangle \right) \otimes |0\rangle_{\mathrm{a}_1} - X' |\phi\rangle |0\rangle_{\mathrm{a}_1} \right\| \leq \epsilon/2 \qquad (78)$$

for all states $|\phi\rangle$.

Proof. We first consider the approximation of $1/x$ by a finite sum of $e^{-z_k H}$:

$$\left| \frac{1}{x} - \delta z \sum_{k=0}^{K-1} e^{-z_k x} \right| = \left| \frac{1}{x} - \delta z \frac{1 - e^{-z_K x}}{1 - e^{-\delta z x}} \right| \qquad (79)$$

$$= \left| \frac{1}{x} O(\delta z x + e^{-z_K x}) \right| \qquad (80)$$

Assuming that $1 \geq x \geq 1/\kappa$ so that $1/x \leq \kappa$, we can upper bound the above quantity by $\epsilon/4$ if we choose $e^{-z_K/\kappa} = \Theta(\epsilon/\kappa)$ and $\delta z = \Theta(\epsilon)$. These imply

$$z_K = \Theta(\kappa \log(\kappa/\epsilon)) \qquad (81)$$

and

$$K = z_K/\delta z = \Theta(\kappa \log(\kappa/\epsilon)/\epsilon) \; . \qquad (82)$$

In the next step, we invoke the proof of Lemma 1 and approximate each $e^{-z_k x}$ as

$$g_k(x) = \frac{1}{\sqrt{2\pi}} \sum_{j=-J}^{J} \delta y \; e^{-y_j^2/2} e^{-iy_j\sqrt{z_k x}} \; , \qquad (83)$$

and we need to choose J and δy so that the approximation error is bounded by $\epsilon/(4z_K)$. Then,

$$\left| \delta z \sum_{k=0}^{K-1} (e^{-z_k x} - g_k(x)) \right| \leq (\delta z K) \frac{\epsilon}{4 z_K} \leq \epsilon/4 \; . \qquad (84)$$

Lemma 1 then implies

$$\delta y = \Theta\left(\frac{1}{\sqrt{z_K \log(z_K/\epsilon)}} \right) = \Theta\left(\frac{1}{\sqrt{\kappa \log(\kappa/\epsilon) \log(\kappa \log(\kappa/\epsilon)/\epsilon)}} \right) = \Theta\left(\frac{1}{\sqrt{\kappa}\log(\kappa/\epsilon)} \right) \; , \qquad (85)$$

and

$$J = \Theta\left(\sqrt{z_K}\log(z_K/\epsilon)\right) = \Theta\left(\sqrt{\kappa}\log^{3/2}(\kappa/\epsilon)\right) \; . \qquad (86)$$

Thus far, we presented an approximation of $1/x$, for $1 \geq x \geq 1/\kappa$, as a doubly weighted sum of terms $\exp(-iy_j\sqrt{z_k x})$. To obtain the desired result, it suffices to act with X' on any eigenstate of $\tilde{H}$ and replace $\sqrt{x}$ by the corresponding eigenvalue, as we did in Lemma 1. Since $H \geq \Delta$, we need to replace κ by $1/\Delta$ in the bounds obtained for δz, δy, K, J, z_K, and y_J. □

In general, we cannot implement the unitaries $e^{-iy_j\sqrt{2z_k}\tilde{H}}$ exactly but we can do so up to an approximation error. We obtain:

Corollary 3. *Let* $\epsilon > 0$ *and* W_{jk} *be a unitary that satisfies*

$$\|W_{jk} - e^{-iy_j\sqrt{2z_k}\tilde{H}}\| \leq \frac{\epsilon}{4z_K} \; . \qquad (87)$$

Let

$$X = \frac{1}{\sqrt{2\pi}} \sum_{k=0}^{K} \delta z \sum_{j=-J}^{J} \delta y \; e^{-y_j^2/2} W_{jk} \; . \qquad (88)$$

Then,

$$\left\| \left(\frac{1}{H} |\phi\rangle \right) \otimes |0\rangle_{\mathrm{a}_1} - X |\phi\rangle |0\rangle_{\mathrm{a}_1} \right\| \leq \epsilon \; . \qquad (89)$$

Proof. If we replace each $e^{-iy_j\sqrt{z_k x}}$ by a term that is an $\epsilon/(2z_K)$ approximation in the definition of $g_k(x)$, it yields an approximation of $1/x$ within precision $\epsilon/2$ plus

$$\frac{\delta z K}{\sqrt{2\pi}} \sum_{j=-J}^{J} \delta y e^{-y_j^2/2} (\epsilon/(4z_K)) = \frac{\epsilon}{4\sqrt{2\pi}} \sum_{j=-J}^{J} \delta y e^{-y_j^2/2} \; . \qquad (90)$$

Our choice of parameters in Lemma 2 implies

$$\left| 1 - \frac{1}{\sqrt{2\pi}} \sum_{j=-J}^{J} \delta y e^{-y_j^2/2} \right| \leq \epsilon/4 \qquad (91)$$

so that the additional error is bounded by $\epsilon/2$. The proof follows by replacing $\sqrt{x}$ by the corresponding eigenvalue of $\tilde{H}$ and $g_k(x)$ by the linear combination of the W_{jk} with weights $\delta y e^{-y_j^2/2}$. □

So far we showed that $1/H$ can be approximated within precision ϵ by a linear combination of unitaries that correspond to evolutions under $\tilde{H}$ for maximum time $y_J\sqrt{z_K} = \Theta((1/\sqrt{\Delta})\log(1/(\Delta\epsilon)))$. Each such evolution must be implemented by a method for Hamiltonian simulation that approximates it within precision $\epsilon' = O(\epsilon/z_K) = O(\epsilon\Delta/(\log(1/(\epsilon\Delta))))$.

In Sec. II we described a technique to implement $X = \sum_{l=0}^{L-1} \gamma_l V_l$. In this case, the coefficients and unitaries are $\delta z \delta y e^{-y_j^2/2}/\sqrt{2\pi}$ and W_{jk}, respectively. From Lemma 2 and Eq. (91), it is simple to show

$$\begin{aligned} |z_K - \gamma| &\leq \sum_{k=0}^{K} \delta z \left| 1 - \frac{1}{\sqrt{2\pi}} \sum_{j=-J}^{J} \delta y e^{-y_j^2/2} \right| \\ &\leq z_K \epsilon/4 \end{aligned} \tag{92}$$

and thus $\gamma \approx z_K$ or $\gamma = \Theta((1/\Delta)\log(1/(\Delta\epsilon)))$.

Last, we define the unitary $T = (T_2)^\dagger T_1$ such that

$$T_1 |0\rangle |0\rangle_{\mathrm{a}} = \sqrt{\pi_{\mathcal{U}}} \frac{X'}{\gamma} |\sqrt{\pi_{\mathcal{U}}}\rangle |0\rangle_{\mathrm{a}} + |\Theta^\perp\rangle , \tag{93}$$

and $|\Theta^\perp\rangle$ is supported in the subspace orthogonal to $|0\rangle_{\mathrm{a}}$. The ancillary register **a** includes the ancillary registers $\mathrm{a}_1, \mathrm{a}_2, \mathrm{a}_3$ as needed for evolving with $\tilde{H}$ and implementing X'. That is, $|0\rangle_{\mathrm{a}} = |0\rangle_{\mathrm{a}_1,\mathrm{a}_2,\mathrm{a}_3}$. T_1 can be implemented as follows. It first uses $Q_{\sqrt{\pi}}$ to prepare the quantum state $|\sqrt{\pi}\rangle$. It then uses $Q_{\mathcal{U}}$ to prepare $\sqrt{\pi_{\mathcal{U}}} |\sqrt{\pi_{\mathcal{U}}}\rangle |0\rangle_{\mathrm{a}'} + \sqrt{\pi_{\mathcal{M}}} |\sqrt{\pi_{\mathcal{M}}}\rangle |1\rangle_{\mathrm{a}'}$, where the ancilla qubit a' is part of the register **a**. Then, conditional on $|0\rangle_{\mathrm{a}'}$, it implements X'/γ as discussed in Sec. II. T_2 is the unitary that prepares $\sqrt{\pi_{\mathcal{U}}} |\sqrt{\pi_{\mathcal{U}}}\rangle |0\rangle_{\mathrm{a}} + \sqrt{\pi_{\mathcal{M}}} |\sqrt{\pi_{\mathcal{M}}}\rangle |0^\perp\rangle_{\mathrm{a}}$, where $|0^\perp\rangle_{\mathrm{a}}$ is orthogonal to $|0\rangle_{\mathrm{a}}$. Then, if $T = (T_2)^\dagger T_1$, we obtain

$$\langle 0| \langle 0|_{\mathrm{a}} T |0\rangle |0\rangle_{\mathrm{a}} = \frac{\pi_{\mathcal{U}}}{\gamma} \langle \sqrt{\pi_{\mathcal{U}}}| X' |\sqrt{\pi_{\mathcal{U}}}\rangle . \tag{94}$$

A. Algorithm

We set $\epsilon' = O(\epsilon\Delta/\log(1/(\epsilon\Delta)))$. The quantum algorithm for estimating the hitting time consists of two basic steps. The first step uses the amplitude estimation algorithm of [18] to provide an estimate of $\langle 0| \langle 0|_{\mathrm{a}_3} T |0\rangle |0\rangle_{\mathrm{a}_3}$ within precision ϵ' and constant confidence level ($c \approx 0.81$). Call that estimate $\tilde{t}_h$. The output of the algorithm is $\hat{t}_h = z_K \tilde{t}_h$.

B. Validity and complexity

As described, our quantum algorithm provides a $O(\epsilon' z_K)$ estimate of $z_K \langle 0| \langle 0|_{\mathrm{a}_3} T |0\rangle |0\rangle_{\mathrm{a}_3}$. Using Eqs. (69) and (94), the output is an estimate of $(z_K/\gamma)t_h$ within precision $O((z_K/\gamma)\epsilon + z_K\epsilon')$. Our choice of ϵ' implies that this is $O((z_K/\gamma)\epsilon)$. Also, using Eq. (92), we obtain

$$\left| 1 - \frac{z_K}{\gamma} \right| = O(\epsilon). \tag{95}$$

Then, our quantum algorithm outputs $\hat{t}_h$, an estimate of t_h within absolute precision $O(\epsilon)$.

Our quantum algorithm uses T, $O(1/\epsilon')$ times. Each T uses $Q_{\mathcal{U}}$ and $Q_{\sqrt{\pi}}$ two times, in addition to the unitaries needed to implement X'. Each such unitary requires evolving with $\tilde{H}$ for maximum time $t = O((1/\sqrt{\Delta})\log(1/(\Delta\epsilon)))$. In addition, each such unitary requires preparing a quantum state proportional to

$$\frac{1}{\sqrt{\gamma}} \sum_{j,k} \left(\frac{\delta y \delta z e^{-y_j^2/2}}{\sqrt{2\pi}} \right)^{1/2} |j,k\rangle . \tag{96}$$

The gate complexity for preparing this state using the results in [30, 31] is $C_B = O(\log(J) + \log(K))$ and then $C_B = O(\log(1/(\Delta\epsilon)))$. The overall gate complexity is

$$O\left(\frac{1}{\epsilon'} \left(C_W(t,\epsilon') + C_{\mathcal{U}} + C_{\sqrt{\pi}} + C_B \right) \right) . \tag{97}$$

This proves Thm. 2. Using Eq. (A24) and replacing for ϵ' and t, and disregarding terms that are polylogarithmic in $d/(\epsilon\Delta)$, the gate complexity is

$$\tilde{O}\left(\frac{1}{\epsilon\Delta} \left(\frac{d^2}{\sqrt{\Delta}} (d\log(N) + C_P + C_{\mathcal{U}}) + C_{\sqrt{\pi}} \right) \right) . \tag{98}$$

V. CONCLUSIONS

We provided quantum algorithms for solving two problems of stochastic processes, namely the preparation of a thermal Gibbs state of a quantum system and the estimation of the hitting time of a Markov chain. Our algorithms combine many techniques, including Hamiltonian simulation, spectral gap amplification, and methods for the quantum linear systems algorithm. They provide significant speedups with respect to known classical and quantum algorithms for these problems and are expected to be relevant to research areas in statistical physics and computer science, including optimization and the design of search algorithms.

We first showed that, starting from a completely entangled state, we can prepare a state that is ϵ-close (in trace distance) to a thermal Gibbs state using resources that scale polylogarithmic in $1/\epsilon$. This is an exponential improvement over previously known algorithms that rely on phase estimation and have complexity that depends polynomially in $1/\epsilon$ [8, 9]. Our algorithm circumvents the limitations of phase estimation by approximating the exponential operator as a finite linear combination of unitary operations and using techniques developed in [24] to implement it. We also used techniques developed in the context of spectral gap amplification [23] to improve the complexity dependence on the inverse temperature, from almost linear in β to almost linear in $\sqrt{\beta}$.

Next, we presented a quantum algorithm to estimate the hitting time of a Markov chain, initialized in its stationary distribution, with almost quadratically less resources in all parameters than a classical algorithm (in a worst-case scenario). This is done by first expressing the hitting time as the expectation value of the inverse of an operator H, which is obtained by a simple transformation

of the Markov chain stochastic matrix. We then used results from [22] to apply $1/H$; in this particular case, H is positive and we showed that the implementation of $1/H$ can be done more efficiently than the algorithm in [22], in terms of the condition number of H. Such an expected value can be computed using methods for amplitude estimation. For constant confidence level ($c \approx 0.81$), the use of amplitude estimation limits us to a complexity dependence that is $\tilde{O}(1/\epsilon)$, where ϵ is the absolute precision of our estimate. It is possible to increase the confidence level towards c with an increase in complexity that is $O(\log(|1-c|))$.

VI. ACKNOWLEDGEMENTS

AC was supported by a Google research award. RS acknowledges support of the LDRD program at LANL and NRO. We thank Sergio Boixo and Ryan Babbush at Google Quantum A.I. Lab Team for discussions.

[1] K. Huang, *Statistical Mechanics* (John Wiley and Sons, Inc., 1963).
[2] R. Agarwal, *Recent Trends in Optimization Theory and Applications* (World Scientific, 1995).
[3] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov chains and mixing times*, http://pages.uoregon.edu/dlevin/MARKOV/markovmixing.pdf.
[4] R. Portugal, *Quantum Walks and Search Algorithms* (Springer, New York, 2013).
[5] M. Newman and G. Barkema, *Monte Carlo Methods in Statistical Physics* (Oxford University Press, 1998).
[6] J. B. Anderson, *Quantum Monte Carlo : Origins, Development, Applications: Origins, Development, Applications* (Oxford University Press, 2007).
[7] R. D. Somma, S. Boixo, H. Barnum, and E. Knill, Phys. Rev. Lett. **101**, 130504 (2008).
[8] D. Poulin and P. Wocjan, Phys. Rev. Lett. **103**, 220502 (2009).
[9] C. Chiang and P. Wocjan, in *Quantum Cryptography and Computing* (2010), pp. 138–147.
[10] E. Bilgin and S. Boixo, Phys. Rev. Lett. **105**, 170405 (2010).
[11] K. Temme, T. Osborne, K. Vollbrecht, D. Poulin, and F. Verstraete, Nature **471**, 87 (2011).
[12] M. Schwarz, K. Temme, and F. Verstraete, Phys. Rev. Lett. **108**, 110502 (2012).
[13] A. Ambainis, J. Kempe, and A. Rivosh, in *Proceedings of the 16th ACM-SIAM Symposium on Discrete Algorithms* (2005), pp. 1099–1108.
[14] M. Szegedy, in *Proceedings of the 45th IEEE Symposium on Foundations of Computer Science* (2004), pp. 575–584.
[15] H. Krovi and T. A. Brun, Phys. Rev. A **73**, 032341 (2006).
[16] F. Magniez, A. Nayak, J. Roland, and M. Santha, in *Proceedings of the 39th ACM Symposium on Theory of Computing* (2007), pp. 575–584.
[17] H. Krovi, M. Ozols, and J. Roland, Phys. Rev. A **82**, 022333 (2010).
[18] E. Knill, G. Ortiz, and R. D. Somma, Phys. Rev. A **75**, 012328 (2007).
[19] A. Montanaro, Proceedings of the Royal Society A **471**, 20150301 (2015).
[20] R. D. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, Phys. Rev. A **65**, 042323 (2002).
[21] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in *Proc. of the 46th ACM Symposium on Theory of Computing* (2014), pp. 283–292.
[22] A. Childs, R. Kothari, and R. D. Somma, preprint arXiv:1511.02306 (2015).
[23] R. D. Somma and S. Boixo, SIAM J. Comp **42**, 593 (2013).
[24] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Phys. Rev. Lett. **114**, 090502 (2015).
[25] G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, Phys. Rev. A **64**, 022319 (2001).
[26] A. M. Childs and N. Wiebe, Quantum Information and Computation **12**, 901 (2012).
[27] L. K. Grover, in *Proceedings of the 28th ACM Symposium on Theory of Computing* (1996), pp. 212–219.
[28] C. W. Helstrom, *Quantum Detection and Estimation Theory* (Academic Press, New York, 1976).
[29] J. Hubbard, Phys. Rev. Lett. **3**, 77 (1959).
[30] A. Kitaev and W. A. Webb, preprint arXiv:0801.0342 (2008).
[31] R. D. Somma, preprint arXiv:1503.06319 (2015).
[32] R. D. Somma, C. D. Batista, and G. Ortiz, Phys. Rev. Lett. **99**, 030603 (2007).
[33] S. Boixo, G. Ortiz, and R. D. Somma, The European Physical Journal **224**, 35 (2015).
[34] D. W. Berry, A. M. Childs, and R. Kothari, in *Proceedings of the 56th Symposium on Foundations of Computer Science* (2015), pp. 792–809.

Appendix A: Simulation of $\tilde{H}$

We provide a method to simulate $\tilde{H}$ in time polylogarithmic in $1/\epsilon$, as required by the algorithm for estimating hitting times of Sec. IV. We assume that there exists a procedure Q_P that computes the locations and magnitude of the nonzero entries of the matrix P. More precisely, Q_P performs the map

$$Q_P |\sigma\rangle = |\sigma\rangle |\sigma'_1, \ldots, \sigma'_d\rangle \otimes \\ \otimes |\Pr(\sigma|\sigma'_1), \Pr(\sigma'_1|\sigma), \ldots, \Pr(\sigma|\sigma'_d), \Pr(\sigma'_d|\sigma)\rangle \ , \tag{A1}$$

where d is the sparsity of P, i.e., the largest number of nonzero matrix elements per row or column. The transition probabilities are assumed to be exactly represented by a constant number of bits and we disregard any rounding-off errors. We also assume access to the

oracle $Q_{\mathcal{U}}$ such that

$$Q_{\mathcal{U}} |\sigma\rangle = -|\sigma\rangle \text{ if } \sigma \in \mathcal{U} \tag{A2}$$

and $Q_{\mathcal{U}} |\sigma\rangle = |\sigma\rangle$ otherwise.

The Hamiltonians $\bar{H}$ and H can be constructed as follows. For each pair (σ, σ'), such that $\sigma \neq \sigma'$ and $\Pr(\sigma|\sigma') \neq 0$, we define an unnormalized state

$$|\mu_{\sigma,\sigma'}\rangle = \frac{1}{\sqrt{2}}(\sqrt{\Pr(\sigma|\sigma')}\,|\sigma'\rangle - \sqrt{\Pr(\sigma'|\sigma)}\,|\sigma\rangle) \,. \tag{A3}$$

Then, if $\sigma_f \neq \sigma_0$,

$$\langle\sigma_f| \sum_{\sigma,\sigma'} |\mu_{\sigma,\sigma'}\rangle \langle\mu_{\sigma,\sigma'}|\,\sigma_0\rangle = -\sqrt{\Pr(\sigma_f|\sigma_0)\Pr(\sigma_0|\sigma_f)}\,, \tag{A4}$$

and if $\sigma_f = \sigma_0$,

$$\begin{aligned}\langle\sigma_0| \sum_{\sigma,\sigma'} |\mu_{\sigma,\sigma'}\rangle \langle\mu_{\sigma,\sigma'}|\,\sigma_0\rangle &= \sum_{\sigma'\neq\sigma} \Pr(\sigma'|\sigma) \\ &= 1 - \Pr(\sigma|\sigma) \,. \end{aligned} \tag{A5}$$

These are the same matrix entries of $\bar{H}$ and the implication is that

$$\bar{H} = \sum_{\sigma,\sigma'} |\mu_{\sigma,\sigma'}\rangle \langle\mu_{\sigma,\sigma'}| \,. \tag{A6}$$

This is the desired representation of the $\bar{H}$ as a sum of positive operators. In particular, we can normalize the states and define

$$|\bar{\mu}_{\sigma,\sigma'}\rangle = \frac{|\mu_{\sigma,\sigma'}\rangle}{\| \,|\mu_{\sigma,\sigma'}\rangle\,\|}\,, \tag{A7}$$

$$\sqrt{\bar{\alpha}_{\sigma,\sigma'}} = \| \,|\mu_{\sigma,\sigma'}\rangle\,\| = \sqrt{\frac{\Pr(\sigma|\sigma') + \Pr(\sigma'|\sigma)}{2}} \,. \tag{A8}$$

Then,

$$\bar{H} = \sum_{\sigma,\sigma'} \bar{\alpha}_{\sigma,\sigma'} |\bar{\mu}_{\sigma,\sigma'}\rangle \langle\bar{\mu}_{\sigma,\sigma'}| \,. \tag{A9}$$

We let $\Pi_{\mathcal{U}}$ be the projector into the subspace $\mathcal{U}$. The Hamiltonian is $H = \Pi_{\mathcal{U}} \bar{H} \Pi_{\mathcal{U}}$, and using Eq. (A6), we obtain

$$\begin{aligned} H = &\sum_{\sigma,\sigma'\in\mathcal{U}} \bar{\alpha}_{\sigma,\sigma'} |\bar{\mu}_{\sigma,\sigma'}\rangle\langle\bar{\mu}_{\sigma,\sigma'}| + \\ &+ \sum_{\sigma'\in\mathcal{U}} \left(\sum_{\sigma\in\mathcal{M}} \Pr(\sigma|\sigma') \right) |\sigma'\rangle\langle\sigma'| \,, \end{aligned} \tag{A10}$$

which is the desired decomposition as a linear combination of rank-1 projectors.

To build $\tilde{H}$, we need to take square roots of the projectors. In principle, the dimension $N_{\mathcal{U}}$ is large and we want to avoid a presentation of $\tilde{H}$ as a sum of polynomially many terms in $N_{\mathcal{U}}$. We are also interested in a decomposition of $\tilde{H}$ in terms of simple unitary operations so that we can use the results of [24] to devise a method to simulate $\exp(-i\tilde{H}t)$. We begin with the second term in the right hand side of Eq. (A10). Its square root is

$$\sum_{\sigma'\in\mathcal{U}} \left(\sqrt{\sum_{\sigma\in\mathcal{M}} \Pr(\sigma|\sigma')} \right) |\sigma'\rangle\langle\sigma'| \,. \tag{A11}$$

This term can be simply obtained as a sum of two diagonal unitary operations:

$$\frac{1}{2}(U_D + U_D^\dagger) \,. \tag{A12}$$

U_D applies a phase to the state $|\sigma'\rangle$ as

$$U_D |\sigma'\rangle = e^{i\theta_{\sigma'}} |\sigma'\rangle \tag{A13}$$

with

$$\cos(\theta_{\sigma'}) = \sqrt{\sum_{\sigma\in\mathcal{M}} \Pr(\sigma|\sigma')}\,, \tag{A14}$$

if $\sigma' \in \mathcal{U}$. Otherwise, $U_D |\sigma'\rangle = i |\sigma'\rangle$. U_D can then be implemented by first using $Q_{\mathcal{U}}$ to detect if σ' is in $\mathcal{U}$ or not. It next applies Q_P and computes $\theta_{\sigma'}$ in an additional register. Conditional on the value of $\theta_{\sigma'}$, it applies the corresponding phase to $|\sigma'\rangle$. It then applies the inverse of Q_P to undo the computation. That is, U_D requires $O(1)$ uses of $Q_{\mathcal{U}}$ and Q_P, and the additional gate complexity is $O(d)$ due to the computation of $\theta_{\sigma'}$.

The first term in the right hand side of Eq. (A10) can be written as a sum of $K' = O(d^2)$ terms as follows. Using Q_P we can implement a coloring of the graph G with vertex set $V(G) = \{\sigma : \sigma \in \mathcal{U}\}$ and edge set $E(G) = \{(\sigma, \sigma') : \sigma, \sigma' \in U, \Pr(\sigma|\sigma') \neq 0\}$. We can use the same coloring as that described in [21], which uses a bipartite graph coloring and was used for Hamiltonian simulation. Each of the K' terms corresponds to one color and is then a sum of commuting rank-1 projectors. That is, the first term in the right hand side of Eq. (A10) is $\sum_{k=1}^{K'} h_k$ and

$$h_k = \sum_{\sigma,\sigma'\in c_k} \bar{\alpha}_{\sigma,\sigma'} |\bar{\mu}_{\sigma,\sigma'}\rangle \langle\bar{\mu}_{\sigma,\sigma'}| \,, \tag{A15}$$

where c_k are those elements of $E(G)$ associated with the k-th color. By the definition of coloring, each rank-1 projector in Eq. (A15) is orthogonal and commutes with each other, and then

$$\sqrt{h_k} = \sum_{\sigma,\sigma'\in c_k} \sqrt{\bar{\alpha}_{\sigma,\sigma'}}\, |\bar{\mu}_{\sigma,\sigma'}\rangle \langle\bar{\mu}_{\sigma,\sigma'}| \,. \tag{A16}$$

We can write

$$\sqrt{h_k} = \frac{-iZ_k + iZ_k^\dagger}{2}\,, \tag{A17}$$

where Z_k is the unitary

$$Z_k = \exp\left(i \sum_{\sigma,\sigma' \in c_k} \delta_{\sigma,\sigma'} \left|\bar{\mu}_{\sigma,\sigma'}\right\rangle \left\langle\bar{\mu}_{\sigma,\sigma'}\right| \right) . \tag{A18}$$

The coefficients are chosen so that

$$\sin(\delta_{\sigma,\sigma'}) = \sqrt{\bar{\alpha}_{\sigma,\sigma'}} , \tag{A19}$$

and $0 \leq \bar{\alpha}_{\sigma,\sigma'} \leq 1$.

We can simulate each Z_k as follows. Note that

$$Z_k \left|\sigma\right\rangle = \xi_{\sigma,\sigma'} \left|\sigma\right\rangle + \xi'_{\sigma,\sigma'} \left|\sigma'\right\rangle \tag{A20}$$

where σ' is such that $(\sigma,\sigma') \in c_k$. The complex coefficients $\xi_{\sigma,\sigma'}$ and $\xi'_{\sigma,\sigma'}$ can be simply obtained from the $\delta_{\sigma,\sigma'}$, and depend only on $\Pr(\sigma|\sigma')$ and $\Pr(\sigma'|\sigma)$. Then, on input $|\sigma\rangle$, we first use $Q_{\mathcal{U}}$ to decide whether $|\sigma\rangle \in \mathcal{U}$ or not. We then apply Q_P once and look for σ' such that $(\sigma,\sigma') \in c_k$. We use an additional register to write a classical description of a quantum circuit that implements the transformation in Eq. (A20). We apply the inverse of Q_P and $Q_{\mathcal{U}}$ and only keep the last register. This is sufficient information to apply the map in Eq. (A20). We can then erase all the additional registers by applying the inverse of the operation that computed the quantum circuit. This works because the quantum circuit is invariant under the permutation of σ and σ'. To implement Z_k we need to use $Q_{\mathcal{U}}$ and Q_P, $O(1)$ times. The additional gate complexity is $O(d\log(N))$ for searching for σ' and describing the quantum circuit.

In summary, we found a decomposition of $\tilde{H}$ as

$$\tilde{H} = \frac{1}{2} \sum_{k=1}^{\tilde{K}} \tilde{U}_k \tag{A21}$$

where $\tilde{U}_k$ are unitaries. The number of terms is $\tilde{K} = O(d^2)$. Using Eq. (9) and the results above, each $\tilde{U}_k$ can be implemented with $O(1)$ uses of $Q_{\mathcal{U}}$ and Q_P, and at most $O(d\log(N))$ additional gates.

Using the results of [24] (see Sec. II), the complexity for simulating $\exp(-i\tilde{H}t)$ within precision ϵ for this case is as follows. The number of uses of $Q_{\mathcal{U}}$ and Q_P is

$$O\left(\tau \log(\tau/\epsilon)/\log\log(\tau/\epsilon)\right) , \tag{A22}$$

where $\tau = |t|d^2$. The additional gate complexity is

$$O\left(d\log(N)\tau \log(\tau/\epsilon)/\log\log(\tau/\epsilon)\right) . \tag{A23}$$

If we write $C_{\mathcal{U}}$ and C_P for the gate complexities of $Q_{\mathcal{U}}$ and Q_P, respectively, the overall gate complexity to simulate the evolution under $\tilde{H}$ is

$$C_W(t,\epsilon) = O\left((d\log(N) + C_{\mathcal{U}} + C_P)\tau \frac{\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)} \right) . \tag{A24}$$