Computing partition functions in the one clean qubit model

Anirban N. Chowdhury
Département de Physique & Institut Quantique, Université de Sherbrooke, QC J1K 2R1, Canada

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

Yi˘git Suba¸sı
Computer, Computational, and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA

(Dated: October 28, 2019)

We present a method to approximate partition functions of quantum systems using mixed-state quantum computation. For non-negative Hamiltonians, our method runs on average in time almost linear in $(M/(ε_{\text{rel}}Z))^2$, where $M$ is the dimension of the quantum system, $Z$ is the partition function, and $ε_{\text{rel}}$ is the relative precision. It is based on approximations of the exponential operator as linear combinations of certain operations related to block-encoding of Hamiltonians or Hamiltonian evolutions. The trace of each operation is estimated using a standard algorithm in the one clean qubit model. For large values of $Z$, our method may run faster than exact classical methods, whose complexities are polynomial in $M$. We also prove that a version of the partition function estimation problem within additive error is complete for the so-called DQC1 complexity class, suggesting that our method provides an exponential speedup. To attain the desired relative precision, we develop a procedure based on a sequence of approximations within predetermined additive errors that may be of independent interest.

I. INTRODUCTION

One of the most important quantities used to describe a physical system in thermodynamic equilibrium is the partition function, $Z$. Many thermodynamic properties, such as the free energy or entropy, can then be derived from $Z$ using simple mathematical relations [1]. Partition functions also appear naturally in many other problems in mathematics and computer science, such as counting the solutions of constraint satisfaction problems [2]. Therefore, developing novel algorithms for partition functions is of great importance [3–7].

In this paper, we present a method for approximating partition functions using the one clean qubit model of computation. In this model, referred to as DQC1 in the literature, the input to the computer is one qubit initialized in a pure state, in addition to $n$ qubits in the maximally mixed state [8]. This is in contrast to standard quantum computation, where the many-qubit initial state is pure [9]. DQC1 has attracted significant attention as it appears that some problems can be solved efficiently within this model, while no efficient classical algorithms may exist [10, 12]. Additionally, DQC1 is practically relevant for mixed-state quantum computation, e.g., it is suitable to describe liquid-state NMR [13, 14], and for quantum metrology [15].

We demonstrate further advantages of the one clean qubit model by describing a method to estimate partition functions of $m$-qubit systems. Our method outputs an estimate $\hat{Z}$ of $Z$, within given relative precision $ε_{\text{rel}} > 0$ and confidence level $c < 1$. For non-negative Hamiltonians, the average complexity is almost linear in $(M/(ε_{\text{rel}}Z))^2$ and $|\log(1 - c)|$, where $M = 2^m$ is the dimension of the quantum system. This complexity is mainly determined by the number of uses of a particular subroutine, which requires measuring a single qubit, while the complexity of each subroutine itself is not a dominating factor in general. For sufficiently large values of $Z$, our method is efficient and runs significantly faster than exact classical methods, whose complexities are polynomial in $M$.

Our basic idea relies on the fact that $Z$ can be approximated from linear combinations of the traces of certain unitary operations. Each such trace can be estimated by repeated uses of the well-known trace-estimation algorithm of Fig. 1. We present two approaches: one in which the unitaries are associated with a block-encoding of the Hamiltonian and another in which they are associated with Hamiltonian evolutions. Whether one approach is more suitable than the other will depend on the specification of the Hamiltonian of the system.

We note that, in the worst case, our method is inefficient and also has complexity polynomial in $M$. This is a drawback since practical implementations of our method could have errors that may not be corrected, potentially impacting the final accuracy when $M$ is large [16]. Nevertheless, the possibility of having efficient computational methods for this problem is basically ruled out by numerous complexity-theory results. For example, computing certain partition functions exactly is #P-hard [17], and even approximating them can be NP-hard; see Ref. [18] for some results. At the same time, our method is efficient (i.e., it runs in time polynomial in $m$) under certain constraints on the additive error and inverse temperature $β$. This problem was previously shown to be DQC1-hard in Ref. [3]. Our result implies that this problem is indeed DQC1-complete, suggesting that efficient classical algo-
rithms for these instances are unlikely, as classical computers are not expected to efficiently simulate the one clean qubit model [8]. Thus, in some cases, our method likely provides an exponential speedup.

The rest of the paper is organized as follows. In Sec. II we introduce the partition function problem where the goal is to estimate \( Z \) within given relative error. In Sec. III we describe the one clean qubit model and introduce the DQC1 complexity class. In Sec. IV we provide two approximations to the exponential operator as specific linear combinations, which will be used by our method. In Sec. V we provide our main algorithms. In Sec. VI we demonstrate the correctness of our method and estimate its complexity. As the unitaries in the approximation are related to block-encoding of Hamiltonians or Hamiltonian simulation, we describe their implementations in Sec. VII where we also obtain the complexity for various specifications of the Hamiltonian. In Sec. VIII we show that a version of the partition function problem is DQC1-complete and we conclude in Sec. IX.

We give some technical proofs in Appendices A, B, and C. In Appendix D we develop an efficient procedure to estimate quantities at a given relative error and confidence level, from estimations with proper additive errors and confidence levels. This procedure is formulated under fairly general assumptions and it can be applied to a wide range of problems beyond the one considered in this paper.

II. PROBLEM STATEMENT

We consider a discrete, \( M \)-dimensional quantum system with Hamiltonian \( H \). In the canonical ensemble, the partition function is

\[
Z := \text{Tr}(e^{-\beta H}) ,
\]

where \( \beta \) is the inverse temperature. That is, \( \beta = 1/k_B T \), with \( k_B \) the Boltzmann constant and \( T \) the temperature. For simplicity, we will focus on systems composed of \( n \) qubits, where \( m = \log_2(M) \). Nevertheless, if one is interested in partition functions of quantum systems obeying different particle statistics, such as bosonic or fermionic systems, the results in Refs. [19] [20] may be used to represent the corresponding operators in terms of Pauli operators acting on qubits. The techniques developed here can then be used to study such systems.

Our main goal is to provide an algorithm that uses one clean qubit to approximate Eq. (1) within given relative precision and confidence level. Formally, we define the partition function problem (PFP) as follows:

**Definition 1** (PFP). Given a Hamiltonian \( H \), an inverse temperature \( \beta \), a relative precision parameter \( \epsilon_{\text{rel}} > 0 \), and a confidence level \( c < 1 \), the goal is to output a positive number \( \hat{Z} \) such that

\[
\left| \hat{Z} - Z \right| \leq \epsilon_{\text{rel}} Z ,
\]

with probability greater than \( c \).

The reason why we focus on relative approximations of the partition function is because they translate to additive approximations for the estimation of thermodynamic quantities such as entropy and free energy. For example, the free energy in thermodynamic equilibrium is given by \( F = -(1/\beta) \log Z \). Using the estimate \( \hat{Z} \) to obtain an estimate \( \hat{F} \), we obtain \( |F - \hat{F}| = O(\epsilon_{\text{rel}}/\beta) \), with probability greater than \( c \). This contrasts the partition function problem studied in Ref. [3], where an additive approximation of \( Z \) is considered, and for which a quantum algorithm in the circuit model is given. Nevertheless, both problems are related, as discussed in Sec. V.

III. THE ONE CLEAN QUBIT MODEL

In the one clean qubit model, the initial state (density matrix) of a system of \( n + 1 \) qubits is

\[
\rho_i = |0\rangle\langle 0| \otimes \frac{1}{2^n} ,
\]

where \( I \) is the identity operator over \( n \) qubits. We write \( \mathcal{H}_n \equiv \mathbb{C}^{2^n} \) for the Hilbert space associated with \( n \)-qubit quantum states. A quantum circuit \( U = U_{L-1} \ldots U_0 \) can then be applied to \( \rho_i \), where each \( U_j \) is a two-qubit quantum gate, and a projective measurement is performed on the ancilla at the end. The outcome probabilities are \( p_+ \) and \( p_- = 1 - p_+ \), where

\[
p_+ = \text{Tr}[\rho_0 U_1 U_0^\dagger ] |0\rangle \langle 0| \otimes I \].
\]

The complexity class DQC1 consists of decision problems that can be solved within the one clean qubit model in polynomial time (in the problem size \( s \)) with correctness probability \( \geq 2/3 \). That is, we are allowed to act on
Corollary 1. Given $\delta > 0$, $\epsilon' < 1$, and a quantum circuit $W$ acting on $m + m'$ qubits, we can obtain an estimate $\hat{\chi}_W \in \mathbb{R}$ that satisfies

$$|\hat{\chi}_W - \text{Re Tr}_m[\langle 0|_m' W |0\rangle_{m'}]| \leq \delta,$$

with probability greater than $\epsilon'$, using the trace-estimation algorithm $Q = \lceil (2^{2(m+m')+1}/\delta^2) \log(2/(1 - \epsilon')) \rceil$ times. Here, $|0\rangle_{m'} \in \mathcal{H}_{m'}$ is the zero state of $m'$ qubits.

Corollary 1 follows from the observation that there is a quantum circuit $V$, acting on $n = m + 2m'$ qubits, and

$$\frac{1}{2m'}\text{Tr}_n[V] = \text{Tr}_{m'}[\langle 0|_{m'} W |0\rangle_{m'}];$$

see Ref. [11]. The unitary $V$ is described in Fig. 2. The proof of Cor. 4 follows from Lemma 1, where the number of qubits is $n = m + 2m'$. The estimate in this case is $\hat{\chi}_W = 2^{m+m'}s_x$.

![FIG. 2. The quantum circuit $V$ that satisfies Eq. (9). The first operations are a sequence of $m'$ CNOT gates on the corresponding qubits [11].](image)

Alternatively, one can obtain $\hat{\chi}_W$ using

$$\text{Re} \text{Tr}_m[\langle 0|_{m'} W |0\rangle_{m'}] = \frac{1}{2} \text{Re} \left( \text{Tr}[W] - \text{Tr}[e^{i\pi|0\rangle_{m'}\langle 0|_{m'} W]} \right),$$

where each term on the right hand side can be obtained using the trace-estimation algorithm. This approach requires starting with the completely mixed state of $m + m'$ qubits (instead of $m + 2m'$), but in this paper we will use the construction of Fig. 2 for simplicity.

IV. APPROXIMATIONS OF THE EXPONENTIAL OPERATOR

Our method for estimating the partition function in the one clean qubit model proceeds by approximating it as a weighted sum of traces of unitary operators. Each such trace can then be computed through repeated uses of the trace-estimation algorithm of Fig. 1. We now describe two approximations of the exponential operator that will be used.
A. Chebyshev Approximation

The first approximation is based on Chebyshev polynomials. If the $m$-qubit Hamiltonian satisfies $\|H\| \leq 1$, we obtain

$$e^{-\beta H} = \sum_{k=-\infty}^{\infty} I_k(\beta)T_k(-H).$$  \hspace{1cm} (11)

Here, $I_k(x) \in \mathbb{R}$ are the modified Bessel functions of the first kind. $T_k(H)$ is an operator acting on $H_m$ obtained by replacing $x$ by $H$ in $T_k(x)$, the $k$-th Chebyshev polynomial of the first kind—see Appendix B. We will approximate the exponential operator by a finite sum, by noticing that $I_k(\beta)$ decays exponentially fast in the large $k$ limit (for fixed $\beta$). In Appendix B we show:

**Lemma 2.** Given $\epsilon_{abs} > 0$ and $\beta \geq 0$, there exists $K = [m + e\beta + \log_2(1/\epsilon_{abs}) + 2]$ such that

$$\|S_K - e^{-\beta H}\|_1 \leq \epsilon_{abs}/2,$$  \hspace{1cm} (12)

where

$$S_K := \sum_{k=-K}^{K} I_k(\beta)T_k(-H).$$  \hspace{1cm} (13)

We can also write $S_K = I_0(\beta) + 2 \sum_{k=1}^{K} I_k(\beta)T_k(-H)$.

To represent $S_K$ as a linear combination of suitable operations for our method, we further assume that there exists a unitary $W_H$, acting on $m + m'$ qubits, that satisfies

$$T_k(-H) = \langle 0|m'(W_H)^k|0\rangle_{m'}.$$  \hspace{1cm} (14)

The operation $W_H$ in Eq. (14) is the “unitary iterate” as used recently for Hamiltonian simulation [23] and linear algebra problems [24]. We discuss how to build $W_H$, which is a block-encoding of $-H$, in Sec. VII A.

Our first approach solves the PFP using the relation

$$\text{Tr}[S_K] = I_0(\beta)M + 2 \sum_{k=1}^{K} I_k(\beta)\text{Tr}_m([0]|m', (W_H)^k|0\rangle_{m'}).$$  \hspace{1cm} (15)

We can use the construction in Fig. 2 to obtain $\text{Tr}_m([0]|m', (W_H)^k|0\rangle_{m'})$ using the trace-estimation algorithm of Fig. 1. As we ultimately seek an estimate within certain relative precision, our main algorithm uses $\epsilon_{abs} \geq \epsilon_{rel} Z_{\text{min}}$, where $\epsilon_{rel}$ is given and $Z_{\text{min}}$ is a lower bound of $Z$; e.g., $Z_{\text{min}} = me^{-\beta}$. To attain this precision, we can always set $K = [5\beta + \log_2(1/\epsilon_{rel}) + 2]$.

B. LCU Approximation

The second approximation is based on the so-called Hubbard-Stratonovich transformation (HST) [3] [25]. If $H \geq 0$, we obtain

$$e^{-\beta H} = \frac{1}{\sqrt{2\pi}} \int dy e^{-y^2/2} e^{-iy\sqrt{2\beta H}}.$$  \hspace{1cm} (16)

Here, $\sqrt{H}$ is also a Hermitian operator that refers to one of the square roots of $H$. As the eigenvalues of $H$ are non-negative, this case appears to be more restrictive. Nevertheless, the assumption $H \geq 0$ may be met after a simple pre-processing step—see Sec. VII A. We wish to obtain an approximation of $e^{-\beta H}$ by a finite linear combination of unitaries following Eq. (16). This approximation is analyzed in Appendix C and was also studied in Ref. [6]. If $\|H\| \leq 1$, we obtain:

**Lemma 3.** Given $\epsilon_{abs} > 0$ and $\beta \geq 0$, there exists $J = [12(\sqrt{\beta} + \sqrt{m + \log_2(1/\epsilon_{abs})})\sqrt{m + \log_2(1/\epsilon_{abs})}]$

$$J = [12(\sqrt{\beta} + \sqrt{m + \log_2(1/\epsilon_{abs})})\sqrt{m + \log_2(1/\epsilon_{abs})}]$$  \hspace{1cm} (17)

and

$$\delta y = \left(2(\sqrt{\beta} + \sqrt{m + \log_2(1/\epsilon_{abs})})\right)^{-1}$$  \hspace{1cm} (18)

such that

$$\|X_J - e^{-\beta H}\|_1 \leq \epsilon_{abs}/2,$$  \hspace{1cm} (19)

where $y_j = j\delta y$ and

$$X_J := \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-J}^{J} e^{-y_j^2/2} e^{-iy_j\sqrt{2\beta H}}.$$  \hspace{1cm} (20)

The proof is in Appendix C. Lemma 3 relates the exponential operator with unitary operators that correspond to evolutions under $\sqrt{H}$ for various times. These evolution operators may not be available—in fact, computing the square root of a Hamiltonian can be related to other computationally hard problems. To overcome this issue, we additionally assume that we have access to a Hamiltonian $H'$, acting on $m + m'$ qubits, that satisfies

$$(H')^2 |\phi\rangle_m |0\rangle_{m'} = (H |\phi\rangle_m) |0\rangle_{m'},$$  \hspace{1cm} (21)

for all pure states $|\phi\rangle \in H_m$. Equation (21) resembles the spectral gap amplification technique discussed in Ref. [26], and was used to demonstrate a number of quantum speedups [27]. We discuss how to build $H'$ in Sec. VII B.

In Appendix C we show

$$\text{Tr}[X_J] = \text{Tr}_m([0]|m', X'_J |0\rangle_{m'}),$$  \hspace{1cm} (22)

with

$$X'_J := \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-J}^{J} e^{-y_j^2/2} e^{-iy_j\sqrt{2\beta H'}}.$$  \hspace{1cm} (23)

Then, if $W_t := e^{-itH'}$ is the evolution operator under $H'$,
our second approach solves the PFP using the relation
\[
\text{Tr}[X_j] = \frac{\delta y}{\sqrt{2\pi}} \left( M + 2 \sum_{j=1}^{J} e^{-y_j^2/2} \text{Re} \text{Tr}_m[\langle 0|m, W_t |0\rangle_m] \right),
\]
where \( t_j := j\delta y\sqrt{2\pi} \). We can use the construction in Fig. 2 to obtain \( \text{Re} \text{Tr}_m[\langle 0|m, W_t |0\rangle_m] \) using the trace-estimation algorithm of Fig. 1. As mentioned, our main algorithm uses \( \epsilon_{\text{abs}} \geq \epsilon_{\text{rel}} M e^{-\beta} \). To attain this precision, we can always set \( J = \lceil 24(\beta + \log_2(1/\epsilon_{\text{rel}})) \rceil \) and \( \delta y = (4\sqrt{\beta + \log_2(1/\epsilon_{\text{rel}})})^{-1} \).

\section{V. ALGORITHMS}

We provide two algorithms in the one clean qubit model for solving the PFP, based on the previous approximations. We first focus on the case where the approximation of the partition function is obtained within a given additive error, and next use this case to obtain an approximation within given relative error. To this end, we assume that we are given \( Z_{\max} \) and \( Z_{\min} \) such that \( Z_{\max} \geq Z \geq Z_{\min} > 0 \). In particular, under our assumptions, we may choose \( Z_{\min} = Me^{-\beta} \) and \( Z_{\max} = Me^{\beta} \) or \( Z_{\max} = M \), depending on whether we work with the first or second approximations, respectively.

\subsection{A. Estimation within additive error}

Algorithm 1.A provides an estimate according to Eq. \((24)\). Specifically, for given \( \epsilon_{\text{abs}} > 0 \) and \( c < 1 \), it outputs \( \hat{Z} \) such that \( |\hat{Z} - Z| \leq \epsilon_{\text{abs}} \) with probability greater than \( c \). It assumes \( ||H|| \leq 1 \) and the existence of a procedure \( W_H \) that acts on \( m + m' \) qubits and satisfies Eq. \((14)\). This procedure is required to build unitaries \( V(k) \) acting on \( m + 2m' \) qubits according to Fig. 2. \( V(k) \) is obtained by replacing \( W \) with \( (W_H)^k \).

\textbf{Algorithm 1.A}

\textbf{Input:} \( \epsilon_{\text{abs}} > 0, c < 1, Z_{\max}, Z_{\min} \).

\begin{itemize}
  \item Obtain \( K \) according to Lemma 2.
  \item Set \( \delta = \epsilon_{\text{abs}}/(2e^\beta), c' = 1 - (1 - c)/K \), and obtain \( Q \) according to Cor. 1.
  \item For each \( k = 1, \ldots, K \):
    \begin{itemize}
      \item Run the trace estimation algorithm \( Q \) times with unitary \( V(k) \). Obtain \( \hat{\chi}_k = 2^{m+m'}s_x \) where \( s_x \) is the average of the measurement outcomes of \( \sigma_x \).
      \item Compute \( \hat{Y}_k = I_0(\beta)M + 2 \sum_{k=1}^{K} f_k(\beta)\hat{\chi}_k \).
      \item \( \hat{Z} = \hat{Y} \) if \( Z_{\max} > \hat{Y} > Z_{\min} \); \( \hat{Z} = Z_{\min} \) if \( \hat{Y} \leq Z_{\min} \); or \( \hat{Z} = Z_{\max} \) otherwise.
    \end{itemize}
\end{itemize}

Algorithm 1.B provides an estimate according to Eq. \((24)\). It assumes \( ||H|| \leq 1, H \geq 0 \), and the existence of a procedure \( W_t \) that implements the evolution under a Hamiltonian \( H_t \), which satisfies Eq. \((21)\). This procedure is required to build unitaries \( V'(t) \) acting on \( m + 2m' \) qubits according to Fig. 2. \( V'(t) \) is obtained by replacing \( W \) with \( W_t \). In the following, \( t_j = j\delta y\sqrt{2\pi} \).

\textbf{Algorithm 1.B}

\textbf{Input:} \( \epsilon_{\text{abs}} > 0, c < 1, Z_{\max}, Z_{\min} \).

\begin{itemize}
  \item Obtain \( J \) and \( \delta y \) according to Lemma 3.
  \item Set \( \delta = \epsilon_{\text{abs}}/4, c' = 1 - (1 - c)/J \), and obtain \( Q \) according to Cor. 1.
  \item For each \( j = 1, \ldots, J \):
    \begin{itemize}
      \item Run the trace estimation algorithm \( Q \) times with unitary \( V'(t_j) \). Obtain \( \hat{\chi}_j = 2^{m+m'}s_x \) where \( s_x \) is the average of the measurement outcomes of \( \sigma_x \).
      \item Compute \( \hat{Y}_j = (\delta y/\sqrt{2\pi})(M + 2 \sum_{j=1}^{J} e^{-y_j^2/2}\hat{\chi}_j) \).
      \item \( \hat{Z} = \hat{Y} \) if \( Z_{\max} > \hat{Y} > Z_{\min} \); \( \hat{Z} = Z_{\min} \) if \( \hat{Y} \leq Z_{\min} \), or \( \hat{Z} = Z_{\max} \) otherwise.
    \end{itemize}
\end{itemize}

The details of \( W_H \) and \( W_t \), for various specifications of \( H \), are discussed in Sec. \( \text{VII} \). For simplicity, in the following we will refer to both, Algorithms 1.A and 1.B, as \( \text{EstimatePF}(\epsilon_{\text{abs}}, c, Z_{\max}, Z_{\min}) \).

\subsection{B. Estimation within relative error}

The PFP is formulated in terms of a relative error \( \epsilon_{\text{rel}} \) and confidence level \( c \). In Appendix D we provide a procedure to attain the desired relative precision from at most \( R := \lceil \log_2(Z_{\max}/Z_{\min}) \rceil \) additive estimations. That procedure goes beyond the PFP and may be of independent interest.

\textbf{Algorithm 2}

\textbf{Input:} \( \epsilon_{\text{rel}}, c, Z_{\max}, Z_{\min} \).

\begin{itemize}
  \item Set \( c' = 1 - (1 - c)/R \), \( r = 0 \), \( Z_0 = Z_{\max} \), and \( \hat{Z}_0 = Z_{\min} \).
  \item While \( \hat{Z}_r > \hat{Z}_r \):
    \begin{itemize}
      \item \( r <- r + 1 \)
      \item Set \( \epsilon_{\text{abs}}(r) = \epsilon_{\text{rel}}Z_{\max}/2^{r+1} \), \( Z_r = Z_{\max}/2^{r} \)
      \item \( \hat{Z}_r = \text{EstimatePF}(\epsilon_{\text{abs}}(r), c', Z_{\max}, Z_{\min}) \)
    \end{itemize}
  \item Output: \( \hat{Z} = \hat{Z}_r \).
\end{itemize}

The value of \( r \) at which Algorithm 2 stops is a random variable. Since Algorithms 1.A and 1.B never report a value smaller than \( Z_{\min} \), \( r \) is bounded by \( R \). Moreover, the expected value of \( r \) is bounded by \( \lceil \log_2(Z_{\max}/Z_{\min}) \rceil + 2 \) and the probability of Algorithm 2 going past this
value decays exponentially – see Appendix [D]. This implies that the smallest additive precision required of EstimatePF is \( \epsilon_{abs} = \Omega(\epsilon_{rel}Z) \) with high probability. A naive approach that considers the worst-case scenario would require an additive precision \( \Omega(\epsilon_{rel}Z_{\min}) \), resulting in significant overheads when \( Z \gg Z_{\min} \).

VI. CORRECTNESS AND COMPLEXITY

Algorithm 1.A obtains \( K \) estimates of \( \text{Tr}_m[(0|m'|W_H)^k [0]_{m'}] \), each within additive error \( \delta = \epsilon_{abs}/(2e^\beta) \) and confidence level \( c' \). It follows that, with confidence at least \( c^K \geq c \),

\[
|\hat{Y} - \text{Tr}[S_K]| \\
\leq 2 \sum_{k=1}^K I_k(\beta) |\hat{X}_k - \text{Tr}_m[(0|m'|W_H)^k [0]_{m'}]| \leq \epsilon_{abs}/2 ,
\]

(25)

where we used \( \sum_{k=1}^K I_k(\beta) \leq e^\beta/2 \). In addition, Lemma 2 implies \( |\text{Tr}[S_K] - Z| \leq \epsilon_{abs}/2 \) and thus \( |\hat{Y} - Z| \leq \epsilon_{abs} \) with confidence greater than \( c \). We can then choose \( \hat{Y} \) as the estimate \( \hat{Z} \) for the partition function in all cases. However, if \( \hat{Y} < Z_{\min} \) or \( \hat{Y} > Z_{\max} \), we can set \( \hat{Z} = Z_{\min} \) or \( \hat{Z} = Z_{\max} \), respectively, and still satisfy

\[
|\hat{Z} - Z| \leq \epsilon_{abs} ,
\]

(27)

with probability greater than \( c \).

Algorithm 1.B obtains \( J \) estimates of \( \text{Tr}_m[(0|m'|e^{\sqrt{2\pi}H'}[0]_{m'}) \), each within additive error \( \delta = \epsilon_{abs}/4 \) and confidence level \( c' \) that satisfies \( c'^J \geq c \). A similar analysis to that of Algorithm 1.A, and using \( (2\hat{Y}/\sqrt{2\pi}) \sum_{j=1}^J e^{-\hat{Y}^2/2} \leq 2 \), implies Eq. (27).

Finally, the proof of the correctness of Algorithm 2 follows directly from the analysis in Appendix [D]. The main observations are that the final \( \epsilon_{abs}(r) \) is sufficient for the desired relative precision and \( \epsilon^{\beta R} \geq \epsilon^{KR} \geq c \). This algorithm outputs an estimate \( \hat{Z} \) that satisfies

\[
|\hat{Z} - Z| \leq \epsilon_{rel}Z ,
\]

(28)

with probability greater than \( c \).

The complexity of the algorithms can be determined from the total number of trace-estimation algorithm runs and the complexity of each such run. Let \( C_V \) and \( C_V' \) be upper bounds to the complexities of \( V(k) \), for Algorithm 1.A, or \( V'(t) \), for Algorithm 1.B, respectively. Estimates of \( C_V \) and \( C_{V'} \) are obtained in Sec. [VI] for different specifications of the Hamiltonian. Then, for given \( \epsilon_{abs} \) and \( c \), Algorithm 1.A uses the trace estimation algorithm \( Q.K \) times, while Algorithm 1.B uses it \( Q.J \) times, for proper choices of \( Q, K, \) and \( J \). Using the results of Sec. [V] the overall asymptotic complexities are

\[
T_{1A} = \tilde{O} \left( C_V M^{2\epsilon_{rel} Z_{\min}}(\beta + m)|\log(1 - c)| \right) ,
\]

(29)

\[
T_{1B} = \tilde{O} \left( C_{V'} M^{2\epsilon_{rel} Z_{\min}}(\beta + m)|\log(1 - c)| \right) .
\]

(30)

The \( \tilde{O} \) notation hides factors that are logarithmic in \( \beta, m, \) and \( 1/\epsilon_{abs} \).

Algorithm 2 uses Algorithms 1.A and 1.B at most \( R \) times and, under the assumptions, \( R = O(\beta) \). It follows that the overall asymptotic complexity of Algorithm 2 is \( T_2 = O(\beta T_{1A}) \) or \( T_2 = O(\beta T_{1B}) \), where the smallest additive precision required on average is \( \epsilon_{abs} = \Omega(\epsilon_{rel}Z) \). The dominating factors in \( T_2 \) are then \( O(M^{2m}(\beta \epsilon_{rel}Z)^2) \) in one case and \( O((M^2\epsilon_{rel}Z)^2) \) in the other. While the latter appears to be exponentially smaller in \( \beta \), we note that, under the condition \( H \geq 0 \), the partition function may also be exponentially smaller in this case. Thus, both factors may be practically of the same order in \( \beta \), depending on \( H \) — see Sec. [VII] for more details.

VII. BLOCK-ENCODING AND HAMILTONIAN SIMULATION

Our algorithms for estimating the partition function require implementing the unitary \( W_H \), which satisfies Eq. (14), or simulating time evolution with a Hamiltonian \( H' \), which satisfies Eq. (21). These operations can be implemented efficiently for suitable specifications of \( H \). Of interest in this paper are the cases where \( H \) is given as a linear combination of unitary or positive semidefinite operators. These cases include physically relevant Hamiltonians that can be decomposed into a sum of tensor products of Pauli matrices, e.g., \( k \)-local Hamiltonians, Hamiltonians appearing in fermionic systems, and more. They also include the so-called frustration-free Hamiltonians that are of relevance in quantum computing and condensed matter (cf., [26]–[28]).

We note that both cases are connected via simple transformations, such as adding or subtracting an overall constant to the Hamiltonian. These transformations can also change \( Z \) by an overall constant factor that may be exponentially large or small in \( \beta \), potentially incurring in complexity overheads. One can therefore use the results in this section to decide which algorithm (1.A or 1.B) results in smaller complexity, for a given \( H \).

Our method may also be applied more broadly, e.g., to sparse Hamiltonians, but the resulting complexities may be large. This is because known methods for simulating sparse Hamiltonians may require, in general, a number of ancillary qubits \( m' = \text{poly}(m) \), and the resulting complexities of our method are exponential in \( m' \).
The overall average gate complexity of Algorithm 2 is \( C_H \), that implement the \( H_i \)'s. Algorithm 1.A requires that \( \|H\| \leq 1 \). We can satisfy this condition if we work with the renormalized Hamiltonian instead, so that \( H \leftarrow H/\alpha \) and \( \beta \leftarrow \beta \alpha \), where \( \alpha = \sum_{l=0}^{L-1} \alpha_l \). This is achieved by a simple pre-processing step, whose complexity is not significant. With no loss of generality, \( L = 2^m \).

The unitary \( W_H \) can be constructed following a procedure in Ref. [23]. We define

\[
|G\rangle := \frac{1}{\sqrt{\alpha}} \sum_{l=0}^{L-1} \sqrt{\alpha_l} |l\rangle ,
\]

which can be prepared with \( O(L) \) gates, and the unitary

\[
U := \sum_{l=0}^{L-1} \sqrt{H_l} \otimes |l\rangle |l\rangle_{m_1} ,
\]

which acts on \( m + m'_1 \) qubits [24] [30]. Then,

\[
H = (G|m_1\rangle U |G\rangle_{m_1} ;
\]

the same block-encoding as considered in Ref. [23]. Let \( U' := U \otimes |0\rangle \langle 0| + U^\dagger \otimes |1\rangle \langle 1| \), where we added an ancilla qubit \( a \). \( U' \) is a unitary acting on \( m + m' \) qubits, with \( m' = m'_1 + 1 \). We obtain

\[
W_H = (\mathbb{1}_m \otimes (2 |G\rangle \langle G|_{m_1} \otimes |+\rangle \langle +|_a - \mathbb{1}_{m'})) \cdot \sigma_a^U U' .
\]

Here, \( \sigma_a^U \) corresponds to the qubit flip Pauli operator of the ancilla \( a \) and \( \mathbb{1}_n \) is the identity operator acting on \( \mathcal{H}_n \). In Ref. [23] it is shown that such \( W_H \) satisfies Eq. (14).

The gate complexity of \( W_H \) is dominated by that of \( U \). This is \( O(LC_H) \). Since \( V(k) \) in Algorithm 1.A uses \( W_H \) at most \( K = O(\beta + \log_2(1/\epsilon_{rel})) \), assuming \( \epsilon_{abs} = \Omega(\epsilon_{rel} M e^{-\beta}) \), the gate complexity of \( V(k) \) is

\[
C_V = O(LC_H(\beta + \log_2(1/\epsilon_{rel}) )) .
\]

The overall average gate complexity of Algorithm 2 is \( O(\beta T_{1A}) \), where \( T_{1A} \) is obtained from Eq. (29) by replacing \( C_V \) with Eq. (35), setting \( \epsilon_{abs} = \Omega(\epsilon_{rel} Z) \), and \( 2^{2m'} = 2L \). That is,

\[
T_2 = \tilde{O} \left( \frac{e^{2\beta M^2} \log(1 - e^{1 - c})}{\epsilon_{rel}^2 Z^2} L^2 C_H \beta^2 (m + \beta) \right) ,
\]

where we dropped terms that are logarithmic in \( \beta \), \( m \), and \( 1/\epsilon_{rel} \). The factor \( e^{2\beta} \) appears as \( H \) can have negative eigenvalues and \( Z \) can be large, i.e., \( Z \leq Me^{2} \).

Due to the requirement \( H \geq 0 \), here we assume that the Hamiltonian is specified as \( H = \sum_{l=1}^{L-1} \alpha_l H_l \), where \( \alpha_l > 0 \) and each \( H_l \) is Hermitian and a projector, i.e., it satisfies \( (H_l)^2 = H_l \geq 0 \). That is, the eigenvalues of \( H_l \) are 0.1. Note that the case in Sec. VII.A can be simply reduced to this one if the eigenvalues of the unitaries are \( \pm 1 \). Algorithm 1.B also requires that \( \|H\| \leq 1 \). Again, we can satisfy this condition if we work with the renormalized Hamiltonian so that \( H \leftarrow H/\alpha \) and \( \beta \leftarrow \beta \alpha \), where \( \alpha = \sum_{l=0}^{L-1} \alpha_l \). With no loss of generality, \( L = 2^m \).

The construction of \( H' \) was previously discussed in Ref. [6] and uses the technique of spectral gap amplification [26]. We obtained

\[
H' = \frac{1}{\sqrt{\alpha}} \sum_{l=1}^{L-1} \sqrt{\alpha_l} H_l \otimes \left( |0\rangle \langle 0|_{m'_1} + |0\rangle \langle 0| \right) ,
\]

which acts on a space of \( m + m'_1 \) qubits. It requires computing the coefficients \( \sqrt{\alpha_l} \) in a simple pre-processing step. Our goal is to simulate \( e^{-itH'} \) and, to this end, we seek a decomposition of \( H' \) as a linear combination of unitaries. Following Sec. VII.A we define

\[
|G\rangle = \frac{1}{\sqrt{\alpha}} \sum_{l=1}^{L-1} \sqrt{\alpha_l} |l\rangle .
\]

(The only difference with Eq. (31) is that the sum runs from \( l = 1 \).) We also define an operator which acts on \( m + m'_1 \) qubits, where \( H_0 := \mathbb{1}_m \)

\[
X := \sum_{l=0}^{L-1} H_l \otimes |l\rangle \langle l|_{m'_1} .
\]

Then,

\[
H' = X |G\rangle \langle G|_{m'_1} + H.c. .
\]

Since \( 2 |0\rangle \langle 0|_{m'_1} = \mathbb{1}_{m'_1} - e^{-i\pi |0\rangle \langle 0|_{m'_1}} \) and \( X = (\mathbb{1} + U)/2 \) for unitary \( U \), we can easily decompose \( H' \) as a linear combination of, at most, 8 unitaries. If \( C_H \) is an upper bound on the gate complexity of the unitaries \( 2(H_l - \mathbb{1}_m) \), the gate complexity of \( U \) is \( O(LC_H) \). Furthermore, the gate complexity of each of the unitaries in the decomposition of \( H' \) will also be \( O(LC_H) \). Note that \( \|H'\| = O(1) \).

Once the decomposition of \( H' \) is obtained, we can use the result in Ref. [23] for Hamiltonian simulation. This provides a quantum algorithm to implement a unitary \( W'_l \), acting on \( m + m' \) qubits, such that it approximates \( W_l \). Here, \( m' = m'_1 + m' \) and, for our case, \( m'_2 = O(1) \). More specifically, the results in Ref. [23] imply

\[
|\text{Tr}_{m'}|(|0\rangle \langle 0|_{m'} W_l |0\rangle \langle 0|_{m'}) - |\text{Tr}_{m'}|(|0\rangle \langle 0|_{m'} W'_l W_l |0\rangle \langle 0|_{m'})| \leq \delta ,
\]

and the gate complexity of \( W'_l \) is \( O(LC_H((|t| + \log(M/\delta)))) \).
Algorithm 1.B requires computing $\text{Tr}_{m}[(0|_{m'} W_{t} |0|_{m'})]$, within additive precision $\delta = O(\epsilon_{\text{abs}})$. This will be obtained by computing $\text{Tr}_{m}[(0|_{m'} W_{t'} |0|_{m'})]$ instead, within the same precision. To this end, we will use the construction of Fig. 2 where $W$ will be replaced by $W_{t'}$. The resulting unitary is $V(t)$. As the additive error is $\epsilon_{\text{abs}} = O(\epsilon_{\text{rel}} M e^{-\beta})$, then $M/\delta = O(e^{\beta}/\epsilon_{\text{rel}})$. Algorithm 1.B also requires simulating $W_{t}$ for a maximum time $t_{J} := J\delta y = O(\sqrt{\beta + \log(1/\epsilon_{\text{rel}})})$.

The gate complexity of $V(t)$ is dominated by that of $W_{t}$. That is,

$$C_{V'} = O\left( L C_{H}(\beta + \log(1/\epsilon_{\text{rel}})) \right).$$  \hspace{1cm} (42)

The overall average gate complexity of Algorithm 2 for this case is $O(\beta_{T_{1.B}})$, where $T_{1.B}$ is obtained from Eq. (30) by replacing $C_{V'}$ with Eq. (42), setting $\epsilon_{\text{abs}} = \Omega(\epsilon_{\text{rel}} Z)$, and $2^{m'} = O(L)$. That is,

$$T_{2} = \tilde{O}\left( M^{2} \log(1-c) \right) L^{3} C_{H} \beta^{2}(m + \beta)$$

where we dropped terms that are logarithmic in $\beta$, $m$, and $1/\epsilon_{\text{rel}}$. The factor $e^{2\beta}$ in Eq. (30) does not appear here as $H \geq 0$ by assumption and $Z \leq M$.

VIII. DQC-1 COMPLETENESS

In Ref. [3] it was shown that, under certain conditions, estimating the partition function to within additive error is DQC1-hard [3]. We now show that such version of the partition function problem is in fact DQC1-complete; our method provides a polynomial-time algorithm in the one clean qubit model. In the following, $\lambda_{\text{min}}(A)$ refers to the lowest eigenvalue of a Hermitian operator $A$.

Definition 2 (PFP-additive [3]). We are given a Hamiltonian $\hat{H}$ acting on $m$ qubits, $\hat{H} = \sum_{i=1}^{L} \hat{h}_{i}$, with $L = \text{poly}(m)$, and three real numbers $\tilde{\beta} = O(\text{poly}(m))$, $c > 0$, and $\epsilon > 0$. Each $\hat{h}_{i}$ acts on at most $k$ qubits and has bounded operator norm $\|\hat{h}_{i}\| = O(\text{poly}(m))$. We are also given a lower bound $\lambda$ to the ground state energy of $\hat{H}$; i.e. $\lambda \leq \lambda_{\text{min}}(\hat{H})$. The goal is to find a number $\hat{\gamma}$ such that, with probability greater than $c$,

$$\left| \hat{\gamma} - \frac{\tilde{Z}}{2m e^{-\tilde{\beta} \lambda}} \right| \leq \epsilon,$$

where $\tilde{Z} = \text{Tr}[e^{-\tilde{\beta} \hat{H}}]$.

Our result in this section is:

Theorem 1. PFP-additive is DQC1-complete for $c > 1/2$, $\epsilon = O(1/\text{poly}(m))$, $k = \log(m)$, and $\lambda = \sum_{i=1}^{L} \lambda_{\text{min}}(\hat{h}_{i})$.

Proof. Reference [3] shows that PFP-additive is DQC1-hard under the stated conditions. It remains to be shown that this problem is indeed in the DQC1 complexity class; we will prove this using Algorithm 1.B.

Let $H := (\hat{H} - \lambda)/\gamma$, where $\gamma := 2 \sum_{i=1}^{L} \|\hat{h}_{i}\|$, and $\beta := \tilde{\beta} \gamma$. Then, $H \geq 0$ and $\|H\| \leq 1$, fitting the framework of Algorithm 1.B. Moreover, the estimate $\hat{\gamma}$ in Eq. (44) is an estimate of $\gamma$ within additive error $\epsilon_{\text{abs}} = \epsilon M$, since $Z = \tilde{Z} e^{\beta \gamma}$. Given access to evolutions under a Hamiltonian $H'$ that satisfies Eq. (21), we can obtain $\hat{\gamma}$.

We now show how to construct $H'$ and approximate its evolution operator in time polynomial in $L$ and $2^{k}$; that is, time polynomial in $m$ under the assumptions. First, we classify obtain a matrix representation for each $\hat{h}_{i}$, whose dimensions are, at most, $2^{k} \times 2^{k}$. We also compute $\lambda_{\text{min}}(\hat{h}_{i})$ and $\|\hat{h}_{i}\|$ for each $l$, and obtain $\gamma$. Next, we construct the operators or matrices $\hat{h}_{i} = (\hat{h}_{i} - \lambda_{\text{min}}(\hat{h}_{i}))/\gamma$ and note that $H = \sum_{i=1}^{L} \hat{h}_{i}$, with $\hat{h}_{i} \geq 0$. We proceed with the spectral decomposition of each $\hat{h}_{i}$ and write it as $\hat{h}_{i} = \sum_{l=1}^{d_{i}} \alpha_{l}^{i} \hat{H}_{l}^{i}$, where $\alpha_{l}^{i} \geq 0$, each $\hat{H}_{l}^{i}$ satisfies $(\hat{H}_{l}^{i})^{2} = \hat{H}_{l}^{i}$ and acts on, at most, $k$ qubits. Using this decomposition, we can write $H = \sum_{i=1}^{L} \alpha_{i} \hat{H}_{i}$, where $\alpha_{l} \geq 0$ and $H_{l}$ are projectors. We can then build $H'$ and a quantum algorithm to simulate $W_{t}$ following Sec. VII.B. In particular, we will construct a unitary $W_{t}'$ that acts on $m + m'$ qubits and satisfies Eq. (41), where it suffices to set $\delta = O(\epsilon_{\text{abs}})$ according to Algorithm 1.B. Following Sec. VII.B, the number of ancillary qubits is $m' = O(\log(L))$ and the gate complexity of $W_{t}'$ is $O(L C_{H}(|t| + \log(M/\delta)))$. We note that $L = O(L 2^{k})$ and $\alpha = \sum_{i=1}^{L} \alpha_{i} = O(2^{k})$. The complexity of all classical steps is then polynomial in $L$ and $2^{k}$. The gate complexity of $W_{t}'$ is $O(L C_{H}(|t| + \log(1/\epsilon)))$, with $C_{H} = O(\text{poly}(2^{k}))$, since each unitary $2H_{l} - 1_{m}$ can be implemented using $O(\text{poly}(2^{k}))$ quantum gates.

Algorithm 1.B uses the trace-estimation algorithm $T = JQ = \tilde{O}(\sqrt{L} 2^{k}/\epsilon^{2})$ times. This follows from Cor. 1 and Lemma 3 using $2^{m'} = O(L)$, $\delta = O(\epsilon_{\text{abs}})$, and $1 - \epsilon = O(1-c)/J$. The $O$ notation hides factors that are logarithmic in $1/c$ and $\beta$. Each run of the trace-estimation algorithm requires applying a unitary $V(t)$ acting on $m + 2m'$ qubits, which uses $W_{t}'$ and other standard gates (see Sec. VII.B). The maximum estimation time is $t_{J} = J\delta y$ and, according to Lemma 3, this is $t_{J} = O(\sqrt{L}/\epsilon)$. Thus, the overall gate complexity of Algorithm 1.B to obtain $\hat{\gamma}$ is $T_{1.B} = \tilde{O}(\sqrt{L} 2^{k} C_{H}/\epsilon^{2})$. That is, $T_{1.B} = O(\text{poly}(m))$ under the assumptions.

In summary, we showed that the complexity of all steps to obtain $\hat{\gamma}$ is polynomial in $m$ and thus PFP-additive is in the DQC1 complexity class.

\[\Box\]

IX. DISCUSSIONS

We provided a method to compute partition functions of quantum systems in the one clean qubit model. For fixed relative precision and confidence levels, and when
the Hamiltonian is non-negative, the complexity of our method is almost linear in $(M/Z)^2$. While our method may be inefficient in general, it can outperform classical methods whose complexity is polynomial in $M$. We also showed that, under certain constraints in the inverse temperature and Hamiltonian, the problem of estimating partition functions within additive error is complete for the DQC1 complexity class. This result suggests that no efficient classical algorithm for this problem exists, while our method is efficient for those instances. It also demonstrates the power of the one clean qubit model for solving problems of relevance in science.

Several simple variants of our method may be considered. For example, instead of estimating each of the traces appearing in Eqs. (15) and (24) and computing the linear combination, we could sample each unitary with a probability that is proportional to the corresponding coefficient. We could also aim at improving our error bounds by avoiding the union bound and noticing that $Z$ is ultimately obtained by sampling independent $\pm 1$ random variables. However, these improvements may not reduce the complexity significantly. The reason is that we use fairly efficient approximations to $e^{-\beta H}$, where the number of terms in the corresponding linear combinations have a mild dependence on the relevant parameters of the problem.

We note that quantum algorithms for partition functions that have complexity almost linear in $\sqrt{M/Z}$ exist (cf. [4, 6]). Whether the complexity of our method can be improved or not is then an interesting open problem. For example, it would imply a quadratic speedup for unstructured search under the presence of oracles in the one clean qubit model. Such a speedup can be ruled out from a theorem in Ref. [8].

\section{Acknowledgements}

ANC acknowledges the Center for Quantum Information and Control, University of New Mexico and the Theoretical Division, Los Alamos National Laboratory, where a part of this work was done. This work was supported by the Laboratory Directed Research and Development program of Los Alamos National Laboratory and by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Quantum Algorithms Teams and Accelerated Research in Quantum Computing programs. Los Alamos National Laboratory is managed by Triad National Security, LLC, for the National Nuclear Security Administration of the U.S. Department of Energy under Contract No. 89233218CNA000001.

\section*{Appendix A: Proof of Lemma 1}

Let $X_1, \ldots, X_Q$ be a set of independent and identically distributed random variables and $X_i \in \{1, -1\}$. These variables can be associated with the outcomes of the $Q$ projective measurements of $\sigma_x$. Let $s_x := \sum_i X_i/Q$. According to Hoeffding’s inequality [22],

$$\Pr(|s_x - \langle \sigma_x \rangle| \geq t) \leq 2 \exp \left( -\frac{t^2 Q}{2} \right),$$

(A1)

where $\langle \sigma_x \rangle$ is also the expected value of each $X_i$. Our estimate is the random variable $\hat{\xi}_V = 2^n s_x$. We can use Eq. (A1) to obtain

$$\Pr(|\hat{\xi}_V - 2^n \langle \sigma_x \rangle| \geq |\delta|) \leq 2 \exp \left( -\frac{\delta^2 Q}{2^{2n+1}} \right).$$

(A2)

Then, it suffices to choose

$$Q = \left\lceil \frac{2^{2n+1}}{\delta^2} \log \left( \frac{2}{1-c'} \right) \right\rceil$$

(A3)

to satisfy $\Pr(|\hat{\xi}_V - 2^n \langle \sigma_x \rangle| \geq \delta) \leq (1-c')$ or, equivalently,

$$\Pr(|\hat{\xi}_V - 2^n \langle \sigma_x \rangle| < \delta) \geq c'.$$

(A4)

Last, we note that $2^n \langle \sigma_x \rangle = \Re \text{Tr}[V]$.

\section*{Appendix B: Approximation of the exponential operator in terms of Chebyshev polynomials}

Let $I_k(z)$ be the modified Bessel function of the first kind. The generating function is $e^{-z \cos \theta} = \sum_{k=-\infty}^{\infty} I_k(-z) e^{ik\theta}$, which implies

$$e^{-\beta H} = \sum_{k=-\infty}^{\infty} I_k(\beta) T_k(-H),$$

(B1)

where $T_k(x)$ is the $k$-th Chebyshev polynomial of the first kind. Equation (B1) was obtained using $I_k(z) = I_{-k}(z)$, $T_k(x) = T_{-k}(x) = \cos(k \arccos x)$, $I_k(-z) = (-1)^k I_k(z)$, and $T_k(-x) = (-1)^k T_k(x)$.

\subsection{1. Proof of Lemma 2}

We wish to approximate the exponential operator by a finite sum of Chebyshev polynomials in $H$. For $|x| \leq 1$, $|T_k(x)| \leq 1$ and thus $\|T_k(H)\|_1 \leq M$. This implies

$$\|S_K - e^{-\beta H}\|_1 \leq M \sum_{|k| \neq K} |I_k(\beta)|,$$

(B2)

where $S_K = \sum_{k=-K}^{K} I_k(\beta) T_k(-H)$.

To bound the right hand side of Eq. (B2), we note that
for $k \geq 0$ the following holds
\[ I_k(\beta) = \left( \frac{\beta}{2} \right)^k \sum_{r=0}^{\infty} \frac{(\beta^2/4)^r}{r!(r+k)!}. \tag{B3} \]
\[ \leq \left( \frac{\beta}{2} \right)^k \frac{I_0(\beta)}{k!}, \tag{B4} \]
where we used $(r+k)! \geq k!r!$. Additionally, $k! > (k/e)^k$ and $I_0(\beta) = \int_0^\infty e^{\beta \cos \theta} d\theta/\pi \leq e^{\beta}$. It follows that
\[ \| S_K - e^{-\beta H} \|_1 \leq M e^{\beta} \sum_{k>0} \left( \frac{\beta e}{2K} \right)^k. \tag{B5} \]
Assume $K \geq \beta e$ and $K \geq 1$ in general, and $\beta \geq 0$. Then,
\[ \| S_K - e^{-\beta H} \|_1 \leq M e^{\beta} \frac{1}{2K-1}. \tag{B6} \]
To bound the right hand side by $\epsilon_{\text{abs}}/2$ we choose
\[ K = \lceil m + e\beta + \log_2(1/\epsilon_{\text{abs}}) + 2 \rceil. \tag{B7} \]
It is easy to show that both assumptions, $K \geq \beta e$ and $K \geq 1$, are satisfied with this choice. We also note that to bound the right hand side of Eq. \text{[B6]} by $\epsilon_{\text{rel}} M e^{-\beta/2}$, which will be the smallest additive error used by our algorithm, it suffices to chose $K = O(\beta + \log_2(1/\epsilon_{\text{rel}})).$

\section*{Appendix C: Approximation of the exponential operator as a linear combination of unitaries}

For $\beta \geq 0$, the Fourier transform of the Gaussian gives
\[ e^{-\beta x^2} = \frac{1}{\sqrt{2\pi}} \int dy \ e^{-y^2/2} e^{-iy\sqrt{2}\beta x}. \tag{C1} \]
This formula can be used to obtain the HST: if $x^2 = \lambda \geq 0$ corresponds to the eigenvalue of $H$, then we can replace $x^2$ by $H$ in Eq. \text{[C1]} and obtain Eq. \text{[C1]}. \]

\subsection*{1. Proof of Lemma 3}

The Poisson summation formula implies
\[ \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-\infty}^\infty e^{-y_j^2/2} e^{-iy_j\sqrt{2}\beta x} = \sum_{k=-\infty}^\infty e^{-\omega_k^2/2}, \tag{C2} \]
where $\delta y > 0$, $y_j = j\delta y$, and $\omega_k = 2\pi k/\delta y + \sqrt{2}\beta x$. Let us choose $\delta y$ and $1 \geq \epsilon > 0$ so that
\[ (2\pi/\delta y) \geq \sqrt{2}\beta x + \sqrt{2}\log(5/\epsilon). \tag{C3} \]
Then, for $|k| \geq 1$, we have $\omega_k^2 \geq k^22\log(5/\epsilon)$, where we considered the worst case in which $\lambda = 0$. We obtain
\[ \sum_{k\neq 0} e^{-\omega_k^2/2} \leq 2 \sum_{k=1}^\infty e^{-\kappa^2 \log(5/\epsilon)} \leq 2 \sum_{k=1}^\infty (\epsilon/\beta)^k \leq 2 \frac{\epsilon/5}{1-\epsilon/5} \leq \epsilon/2. \tag{C4} \]
It follows that
\[ \left| \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-\infty}^\infty e^{-y_j^2/2} e^{-iy_j\sqrt{2}\beta x} - e^{-\beta \lambda} \right| \leq \epsilon/2. \tag{C8} \]
Moreover, we can choose $J$ such that
\[ y_J \geq \sqrt{6\log(2/\epsilon)} \geq 2, \tag{C9} \]
and
\[ \frac{\delta y}{\sqrt{2\pi}} \sum_{|j| > J} e^{-y_j^2/2} \leq \int_{y_J}^{\infty} dy \ e^{-y^2/2} \leq (2/\epsilon) e^{-y_J^2/2} \leq (\epsilon/2)^3 \leq \epsilon/2. \tag{C13} \]
In particular, we can choose $\delta y = (\sqrt{3} + \sqrt{\log(5/\epsilon)})^{-1}$ and $J = \lceil 3(\sqrt{3} + \sqrt{\log(5/\epsilon)}) \sqrt{\log(5/\epsilon)} \rceil$ so that
\[ \left| \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-J}^J e^{-y_j^2/2} e^{-iy_j\sqrt{2}\beta x} - e^{-\beta \lambda} \right| \leq \epsilon. \tag{C14} \]
Note that we assumed $0 \leq \lambda \leq 1$. Larger values of $J$ and/or a smaller values of $\delta y$ will also imply Eq. \text{[C14]}: Lemma 3 then follows from replacing $\lambda$ by $H$ and $\epsilon$ by $\epsilon_{\text{abs}}/(2M)$ in Eq. \text{[C14]}:
\[ \left| \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-J}^J e^{-y_j^2/2} e^{-iy_j\sqrt{2}\beta H} - e^{-\beta H} \right|_1 \leq \epsilon_{\text{abs}}/2. \tag{C15} \]
We can simplify the expressions for $\delta y$ and $J$ using $\log(10M/\epsilon_{\text{abs}}) \leq 4(m + \log_2(1/\epsilon_{\text{abs}}))$, where $m = \log_2(M)$. In particular, we can choose
\[ \delta y = \left( 2(\sqrt{3} + \sqrt{m + \log_2(1/\epsilon_{\text{abs}})}) \right)^{-1}, \tag{C16} \]
\[ J = \left[ 12(\sqrt{3} + \sqrt{m + \log_2(1/\epsilon_{\text{abs}})}) \sqrt{m + \log_2(1/\epsilon_{\text{abs}})} \right]. \tag{C17} \]
To bound the right hand side of Eq. (C15) by $\epsilon_{\text{rel}} M e^{-\beta/2}$, which will be the smallest additive error used by our algorithm, we suffice to choose $J = O(\beta + \log_2(1/\epsilon_{\text{rel}}))$ and $\delta y = \Omega((\sqrt{\beta} + \log_2(1/\epsilon_{\text{rel}}))^{-1})$. Then, $t_J := J \delta y = O((\sqrt{\beta} + \log_2(1/\epsilon_{\text{rel}}))$.

2. Proof of Eq. (22)

Let $|\psi_\lambda\rangle$ be an eigenvector of $H$ of eigenvalue $\lambda \geq 0$, that is $H |\psi_\lambda\rangle = \lambda |\psi_\lambda\rangle$. Then, we can write $(X_J |\psi_\lambda\rangle_m \rangle |0\rangle_{m'})$ using Eq. (20) as

$$\left(\frac{\delta y}{\sqrt{2\pi}} \sum_{j=-J}^J e^{-y_j^2/2} e^{-iy_j \sqrt{2\pi} m |\psi_\lambda\rangle_m \rangle |0\rangle_{m'}\right). \quad (C18)$$

If $\lambda = 0$, the Hamiltonian $H'$ of Eq. (21) has $|\psi_\lambda\rangle_m \rangle |0\rangle_{m'}$ as eigenvector of eigenvalue 0. Otherwise, $H'$ leaves the subspace spanned by $\left\{|\psi_\lambda\rangle_m \rangle |0\rangle_{m'} |H' |\psi_\lambda\rangle_m \rangle |0\rangle_{m'}\right\}$ invariant. We let $|\psi_\lambda^+\rangle$ be the normalized state in this subspace that is orthogonal to $|\psi_\lambda\rangle_m \rangle |0\rangle_{m'}$. The two-dimensional representation of $H'$ is

$$H' = \begin{pmatrix} a_\lambda & b_\lambda \\ b_\lambda & c_\lambda \end{pmatrix}. \quad (C19)$$

With no loss of generality, $a_\lambda, b_\lambda, c_\lambda \in \mathbb{R}$. According to Eq. (21), $H'$ must satisfy

$$(H')^2 = \begin{pmatrix} \lambda & 0 \\ 0 & \gamma \end{pmatrix}, \quad (C20)$$

where $\gamma \geq 0$. It follows that $(a_\lambda^2 + b_\lambda^2) = \lambda$, and either $b_\lambda = 0$ or $(a_\lambda + c_\lambda) = 0$. In the first case, $|\psi_\lambda\rangle_m \rangle |0\rangle_{m'}$ is an eigenvector of $H'$ with eigenvalue $\pm \sqrt{\lambda}$. In the second case, $\lambda = \lambda$ and $H'$ has two eigenvectors with distinct eigenvalues $\pm \sqrt{\lambda}$. Thus, in general, $|\psi_\lambda^+\rangle = \alpha_+ |\psi_\lambda^+\rangle + \alpha_- |\psi_\lambda^-\rangle$, where $|\psi_\lambda^\pm\rangle$ are the eigenvectors of $H'$ of eigenvalues $\pm \sqrt{\lambda}$, respectively. We obtain:

$$X_J |\psi_\lambda^+\rangle \rangle |0\rangle_{m'} = \alpha_+ \delta y \sum_{j=-J}^J e^{-y_j^2/2} e^{-iy_j \sqrt{2\pi} m |\psi_\lambda^+\rangle +}$$

$$+ \alpha_- \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-J}^J e^{-y_j^2/2} e^{-iy_j \sqrt{2\pi} m |\psi_\lambda^-\rangle} \quad (C21)$$

$$= \frac{\delta y}{\sqrt{2\pi}} \sum_{j=-J}^J e^{-y_j^2/2} e^{-iy_j \sqrt{2\pi} m |\psi_\lambda\rangle_m \rangle |0\rangle_{m'}} \quad (C22)$$

$$= (X_J |\psi_\lambda\rangle_m \rangle |0\rangle_{m'}) \quad (C23)$$

We used the property that the sums are invariant under the transformation $y_j \rightarrow -y_j$, together with Eq. (C18).

The last step is

$$\text{Tr}[X_J] = \sum_{\lambda} \langle \psi_\lambda | X_J | \psi_\lambda \rangle \quad (C24)$$

$$= \sum_{\lambda} \langle \psi_\lambda | (0)_{m'} X_J (0)_{m'} | \psi_\lambda \rangle \quad (C25)$$

$$= \text{Tr}[m] [(0)_{m'} X_J (0)_{m'}]. \quad (C26)$$

Appendix D: Estimation within relative error

Our algorithm to estimate the partition function within given relative precision proceeds by making estimations within decreasing additive error until success. We now present this method and prove its correctness.

Let $X$ be the quantity to estimate and assume $X_{\text{max}} \geq X \geq X_{\text{min}} > 0$, for known $X_{\text{max}}$ and $X_{\text{min}}$. For $r = 1, 2, \ldots$, we define $X_r := X_{\text{max}}/2^r$ and $\epsilon_{\text{abs}}(r) := \epsilon_{\text{rel}} X_{\text{max}}/2^{r+1}$, for given $\epsilon_{\text{rel}} \leq 1$. Let us assume that there is a procedure $\text{Estimate}(\epsilon_{\text{abs}}(r), c', X_{\text{max}}, X_{\text{min}})$ that outputs $X_r$, $X_{\text{max}} \geq X_r \geq X_{\text{min}}$, satisfying

$$\text{Pr}(|X_r - X| \leq \epsilon_{\text{abs}}(r)) \geq c', \quad (D1)$$

where $c' < 1$ is a confidence level. In particular, we choose $c' \geq 1 - (1 - c)/R$, where $R = \log_2(X_{\text{max}}/X_{\text{min}})$, for given $c < 1$. We claim that the following algorithm outputs an estimate $\tilde{X}$ such that

$$\text{Pr}(|\tilde{X} - X| \leq \epsilon_{\text{rel}}(R)) \geq c. \quad (D2)$$

Algorithm 3

**Input:** $\epsilon_{\text{rel}}, c$, $X_{\text{max}}, X_{\text{min}}$.

- Set $c' = 1 - (1 - c)/R$, $r = 0$, $X_0 = X_{\text{max}}$, and $X_0 = X_{\text{min}}$.
- While $X_r > X_r$:
  
  - $r \leftarrow r + 1$
  
  - Set $\epsilon_{\text{abs}}(r) := \epsilon_{\text{rel}} X_{\text{max}}/2^{r+1}$, $X_r = X_{\text{max}}/2^r$.
  
  - $X_r := \text{Estimate}(\epsilon_{\text{abs}}(r), c', X_{\text{max}}, X_{\text{min}})$

**Output:** $X_r = \tilde{X}$.

Let $R$ be the random variable associated with the number of times the procedure $\text{Estimate}$ is used until the algorithm stops. In the worst case, all estimates $X_r$ are $X_{\text{min}}$, and since $X_{\text{min}} \geq X_{\text{max}}/2^R$, then $1 \leq R \leq R$. Additionally, if we consider all the estimates $X_1, \ldots, X_R$, the probability that any such estimate is more than $\epsilon_{\text{abs}}(R)$ from $X$ is bounded from above by $R(1 - c') \leq 1 - c$. If $|X_r - X| \leq \epsilon_{\text{rel}}(R)$, then

$$|X_r - X| \leq \epsilon_{\text{rel}}(R)/\tilde{X}_{r} \quad (D3)$$

$$\leq \epsilon_{\text{abs}}(R)/\tilde{X}_{r} \quad (D4)$$

$$\leq \epsilon_{\text{rel}}/2. \quad (D5)$$
$$\left(1 - \epsilon_{\text{rel}}/2\right)^{-1} \geq \hat{\lambda}_R/\lambda \geq \left(1 + \epsilon_{\text{rel}}/2\right)^{-1}, \quad (D6)$$

or

$$\left|1 - \hat{\lambda}_R/\lambda\right| \leq \sum_{j \geq 1} \left(\epsilon_{\text{rel}}/2\right)^j \leq \epsilon_{\text{rel}}. \quad (D7)$$

The condition in Eq. (D7) occurs with probability greater than c, and this proves that Algorithm 3 is correct.

It is important to understand the properties of the distribution for $\mathcal{R}$, especially in those cases where $\lambda \gg \lambda_{\min}$, and obtain a bound on the expected runtime that depends on the actual $\lambda$. With no loss of generality, there exists $q \geq 1$, such that $\lambda_{\max}/2^{q-1} > \lambda \geq \lambda_{\max}/2^q$. For all $r \geq q+1$, the algorithm Estimate implies $\Pr(\hat{\lambda}_r < \lambda_r) \leq 1 - c^r$. Let $p^r$ be the cumulative probability that $\mathcal{R} \leq r$, i.e. the probability that the algorithm stops at or before the $r^{th}$ step. Then, for $r \geq q+1$,

$$p^r \geq p_q + (1 - p_q) \left[1 - \left(1 - c^r\right)^{r-q}\right],$$

which rapidly converges to 1. Alternatively, the probability that the algorithm goes beyond a step $r > q$ decays exponentially with $r$.

The running time of this algorithm is given by the expected value of $\mathcal{R}$. To find an upper bound on $E_{\mathcal{R}}$, we consider the worst case scenario in which the algorithm never stops for $r < q+1$. Then the probability that the algorithm stops at step $r = q + k$ is upper bounded by $(1 - c^r)^{k-1}c^r$ for $k \geq 1$. It follows that:

$$E_{\mathcal{R}} \leq \sum_{k=1}^{\infty} c^r(1 - c^r)^{k-1}(q + k) \leq q + 1 + \frac{1 - c^r}{c^r} \leq q + 2 \quad (D12)$$

where we assumed $c^r \geq 1/2$ — a mild constraint as we are interested in the cases where $c \to 1$. Then,

$$E_{\mathcal{R}} \leq \left[\log_2(\lambda_{\max}/\lambda)\right] + 2. \quad (D14)$$

[1] R. Pathria, *Statistical Mechanics* (Pergamon Press, New York, 1972).
[2] A. Bulatov and M. Grohe, Theoretical Computer Science 348, 148 (2005).
[3] F. G. dos Santos Lins Brandão, *Entanglement Theory and the Quantum Simulation of Many-Body Physics*, Ph.D. thesis, Imperial College of Science, Technology and Medicine (2008).
[4] D. Poulin and P. Wocjan, Phys. Rev. Lett. 103, 220502 (2009).
[5] M. V. den Nest, W. Dür, R. Raussendorf, and H. J. Briegel, Phys. Rev. A 80, 052334 (2009).
[6] A. N. Chowdhury and R. D. Somma, Quant. Inf. Comp. 17, 0041 (2017).
[7] A. Harrow, S. Mehraban, and M. Soleimanfar, arXiv:1910.09071 (2019).
[8] E. Knill and R. Laflamme, Phys. Rev. Lett. 81, 5672 (1998).
[9] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).
[10] D. Poulin, R. Blume-Kohout, R. Laflamme, and H. Ollivier, Phys. Rev. Lett. 92, 177906 (2004).
[11] P. Shor and S. Jordan, Quant. Inf. Comp. 8, 681 (2008).
[12] C. Cade and A. Montanaro, in 13th Conf. on the Theory of Quantum Computation, Communication and Cryptography (TQC 2018) (Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018).
[13] R. Laflamme, E. Knill, D. Cory, E. Fortunato, T. Havel, C. Miquel, R. Martinez, C. Negrevergne, G. Ortiz, M. Pravia, Y. Sharf, S. Sinha, R. Somma, and L. Viola, arXiv quant-ph/0207172 (2002).
[14] C. Negrevergne, R. Somma, G. Ortiz, E. Knill, and R. Laflamme, Phys. Rev. A 71, 032344 (2005).
[15] S. Boixo and R. D. Somma, Phys. Rev. A 77, 052320 (2008).
[16] How large $M$ can be will depend on the gate errors. A bound on $M$ could be set by allowing an overall complexity such that the accumulation of gate errors is well below the desired final precision.
[17] M. Jerrum and A. Sinclair, SIAM J. Comp. 22, 1087 (1993).
[18] C. McQuillan, *Computational complexity of approximation of partition functions*, Ph.D. thesis, University of Liverpool (2013).
[19] R. D. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, Phys. Rev. A 65, 042323 (2002).
[20] R. D. Somma, G. Ortiz, E. Knill, and J. E. Gubernatis, Int. J. Quant. Inf. 1, 189 (2003).
[21] D. Shepherd, arXiv quant-ph/0608132 (2006).
[22] W. Hoefding, J. of the Am. Stat. Assoc. 58 (301), 13 (1963).
[23] G. H. Low and I. L. Chuang, Quantum 3, 163 (2019).
[24] A. M. Childs, R. Kothari, and R. D. Somma, SIAM J. Comp. 46, 1920 (2017).
[25] J. Hubbard, Phys. Rev. Lett. 3, 77 (1959).
[26] R. D. Somma and S. Boixo, SIAM J. Comp 42, 593 (2013).
[27] S. Boixo, G. Ortiz, and R. Somma, Eur. Phys. J. Special Topics 224 (2015).
[28] S. Bravyi and B. Terhal, SIAM J. Comp. 39, 1462 (2009).
[29] C. D. Batista and R. D. Somma, Phys. Rev. Lett. 109, 227203 (2012).
[30] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in *Proc. of the 40th ACM Symposium on Theory of Computing* (2014) pp. 283–292.