Electronic Colloquium on Computational Complexity, Report No. 174 (2021)

Sublinear quantum algorithms for estimating von Neumann entropy

Tom Gur * Min-Hsiu Hsieh † Sathyawageeswar Subramanian ‡

Abstract

Entropy is a fundamental property of both classical and quantum systems, spanning myriad theoretical and practical applications in physics and computer science. We study the problem of obtaining estimates to within a multiplicative factor $\gamma > 1$ of the Shannon entropy of probability distributions and the von Neumann entropy of mixed quantum states. Our main results are:

- an $\widetilde{\mathcal{O}}\left(n^{\frac{1+\eta}{2\gamma^2}}\right)$-query quantum algorithm that outputs a γ-multiplicative approximation of the Shannon entropy $H(\mathbf{p})$ of a classical probability distribution $\mathbf{p} = (p_1, \ldots, p_n)$;
- an $\widetilde{\mathcal{O}}\left(n^{\frac{1}{2}+\frac{1+\eta}{2\gamma^2}}\right)$-query quantum algorithm that outputs a γ-multiplicative approximation of the von Neumann entropy $S(\rho)$ of a density matrix $\rho \in \mathbb{C}^{n\times n}$.

In both cases, the input is assumed to have entropy bounded away from zero by a quantity determined by the parameter $\eta > 0$, since, as we prove, no polynomial query algorithm can multiplicatively approximate the entropy of distributions with arbitrarily low entropy. In addition, we provide $\Omega\left(n^{1/3\gamma^2}\right)$ lower bounds on the query complexity of γ-multiplicative estimation of Shannon and von Neumann entropies. We work with the quantum purified query access model, which can handle both classical probability distributions and mixed quantum states, and is the most general input model considered in the literature.

*University of Warwick Email: tom.gur@warwick.ac.uk.
†Hon Hai (Foxconn) Quantum Computing Research Center. Email: min-hsiu.hsieh@foxconn.com.
‡University of Warwick. Email: sathya.subramanian@warwick.ac.uk.
T. Gur and S. Subramanian are supported by the UKRI Future Leaders Fellowship MR/S031545/1.

ISSN 1433-8092

Contents

1. Introduction

Entropy as a scientific concept is central in the study of a vast variety of subjects, ranging from thermodynamics and information theory to networks and quantum entanglement. The notion of entropy mathematically measures the amount of disorder and uncertainty in a system composed of many parts. Indeed, the second law of thermodynamics can famously be encapsulated in the simple statement that the entropy of a closed system can never decrease.

In this work we focus on the most fundamental entropic functionals for both classical and quantum objects. For classical systems, the Shannon entropy $H(\mathbf{p}) := -\sum_{i\in[n]} p_i \log p_i$ of a probability distribution $\mathbf{p} = (p_1, \ldots, p_n)$ is a cornerstone of information theory. Notably, the Shannon entropy is proportional to the rates at which input data can be transmitted over communication channels [Sha48].

For quantum systems, the von Neumann entropy $S(\rho) := -\operatorname{Tr}(\rho \log \rho)$ [Neu27; Pet01] of a mixed state specified by its density matrix $\rho \in \mathbb{C}^{n\times n}$ is pivotal to our understanding of key properties in quantum mechanics, such as the amount of entanglement contained in bipartite quantum systems. In terms of information theory, von Neumann entropy gives an asymptotic lower bound for the rate at which quantum data can be compressed in a noiseless fashion [Sch95].

The von Neumann entropy is a strict generalisation of the Shannon entropy and reduces to the latter when viewed in appropriately restricted settings. Other entropic functionals built on the von Neumann entropy are also widely used in characterising quantum systems. Moreover, they arise extensively in condensed matter and high energy physics [Laf16], and are often used as operational measures in quantum information-processing tasks [KRS09; HOW06]. They have also had immense theoretical implications in the theory of gravity and black holes [Bek73; Don16], and their study from a quantum information-theoretic viewpoint continues to be a rich source of new physical insights [AS18; AK20].

Since obtaining a perfect description of a system is typically impossible, estimating the entropy of an unknown probability distribution or quantum state using a bounded number of samples, queries, or measurements is a vital algorithmic task. This question received much attention in classical information theory, and in a series of works [BDK+02; WY16; JVH+15; VV11] spanning the last two decades, nearly tight bounds were shown on the sample complexity of classical algorithms for estimating Shannon entropy.

In this paper, we study quantum algorithms for the problem of obtaining multiplicative estimates of the Shannon entropy of classical probability distributions and the von Neumann entropy of mixed quantum states. For generality, we focus on algorithms in the quantum purified query access model, which can handle both classical probability distributions and mixed quantum states, and is the most general input model considered in the literature. Thus, our algorithms can be implemented in all the four major input models for quantum algorithms accessing probability distributions: quantum samples, quantum queries to frequency vectors and classically drawn samples, as well as purifications.

Approximation algorithms that estimate a quantity to within a multiplicative factor, i.e., estimating $x > 0$ by outputting $\tilde{x} \in [x/\gamma, \gamma x]$ for some $\gamma > 1$, allow for much flexibility. On the one hand, via a correct choice of parameters they allow us to recover additive approximations, i.e., estimating $x > 0$ by outputting $\tilde{x} \in [x - \epsilon, x + \epsilon]$ for a small precision parameter $\epsilon \in (0, 1)$ (see Section 1.1.4). On the other hand, in natural settings of parameters, they also allow for a greater slack than additive approximation (for instance, in many applications we may only need to know the unknown quantity to within a factor of two, i.e., $\gamma = 2$). This slack allows us to obtain far more efficient algorithms than is possible with additive approximation; indeed, we attain sublinear complexity for entropy estimation as opposed to polynomial complexity (which is the best that can be achieved for additive estimates). In turn, these properties of multiplicative approximation algorithms make them often more involved and harder to construct.

In the classical setting, Batu et al. [BDK+02] considered the problem of estimating Shannon entropy to multiplicative precision, showing that $\widetilde{\mathcal{O}}(n^{(1+\eta)/\gamma^2})$ samples suffice to obtain an estimate within a factor γ, for classical distributions p with $H(p) > \gamma/\eta$. This is almost matched by a lower bound of $\Omega(n^{(1-\eta)/\gamma^2})$ later proven in [Val11].

We build on the aforementioned line of work by extending it to the setting of quantum query algorithms for both classical distributions and mixed quantum states. In particular, we are motivated by the following question:

Is it possible to construct sublinear quantum algorithms for estimating von Neumann entropy?

1.1. Main results

This paper answers the foregoing question in the affirmative. We begin by presenting our construction of quantum query algorithms for estimating the Shannon entropy of a probability distribution to within a multiplicative factor $\gamma > 1$. Then, we proceed to estimating the von Neumann entropy of mixed quantum states. Finally, we show lower bounds on the quantum query complexity of the foregoing problems. The query complexity in our setting is the standard analogue to the classical sample complexity, and our results are in this sense information theoretic in nature.

1.1.1. Estimating Shannon entropy

We first consider quantum algorithms for classical distributions, accessed via quantum query oracles. We obtain a quadratic improvement in the information theoretic complexity over the best possible classical algorithm.

Theorem 1. *There is a quantum algorithm that outputs with high probability a γ-multiplicative approximation of the Shannon entropy $H(\mathbf{p})$ of a classical probability distribution $\mathbf{p}$ on $[n]$ accessed via a purified quantum query oracle $U_{\mathbf{p}}$ as in Eq. (2.10) using $U_{\mathbf{p}}$ and its inverse $\widetilde{\mathcal{O}}\left(n^{(1+\eta)/2\gamma^2}\right)$ times, provided $H(\mathbf{p}) > 3\gamma + 4/\eta$.*

We remark that there are four popular quantum input models that have been studied for quantum algorithms accessing classical input probability distributions [BHH11; CFM+10; MW16; LW19; Bel19; GL20], namely: (i) quantum query oracle to a frequency vector, (ii) quantum query oracle to list of classically generated samples, (iii) quantum samples with preparation oracle, and (iv) purified quantum query access. (See formal definitions in Section 2.1.) We stress that Theorem 1 holds for all of the models above.

In more detail, Belovs [Bel19] initiated a comparative study of the aforementioned four models and showed that the purified access model (i.e., Model (iv)) is the most general in the sense that it can capture both classical distributions and density matrices, as well as be emulated by all the other models with a constant overhead. Furthermore, he proved that the quantum samples model (i.e., Model (iii)) is strictly stronger than the rest of the models, and conjectured that Models (i), (ii), and (iv) are equivalent for classical probability distributions. Our algorithms are constructed using purified query oracles, and hence by the above, the upper bounds we prove automatically apply to the rest of the models.

1.1.2. Estimating von Neumann entropy

Next, we proceed to look at quantum algorithms for mixed quantum states. As far as we are aware, this work is the first to investigate multiplicative approximations of the von Neumann entropy of density matrices. We prove the following theorem showing that such approximation is possible with query complexity that is *sublinear* in the dimension of the state.

Theorem 2. *There is a quantum algorithm that outputs a γ-multiplicative approximation of the von Neumann entropy $S(\rho)$ of a density matrix $\rho \in \mathbb{C}^{n\times n}$ accessed via a purified quantum query oracle U_ρ as in Eq. (2.9) using U_ρ and its inverse $\widetilde{\mathcal{O}}\left(n^{1/2+(1+\eta)/2\gamma^2}\right)$ times, provided $S(\rho) > 3\gamma + 4/\eta$.*

To the best of our knowledge, this is the first example of an algorithm for estimating von Neumann entropy with sublinear complexity. In contrast, we remark that standard (tomographic) methods for learning the state, obtaining an additive estimate, or even *testing* properties of its spectrum [OW15] typically require a number of samples or queries that scales linearly (for additive approximation) or even quadratically (for learning and testing) in the dimension n of the system. We provide a detailed comparison of our algorithms with related works in Section 1.2.

1.1.3. Lower bounds

We complement the foregoing upper bounds by proving lower bounds on the query complexity of quantum algorithms for multiplicative entropy estimation. For the general purified query access model (i.e., Model (iv)), in which we also show our upper bounds, we show that $\Omega\left(n^{1/3\gamma^2}\right)$ uses of the quantum query oracle are necessary to γ-approximate the entropy of an unknown classical distribution, even when we are promised that the

input has entropy larger than $\log n/\gamma^2$. In fact, the aforementioned lower bound is proved via a reduction to a variant of the collision problem [AS04] in the frequency vector model (i.e., Model (i)), and so it also holds for this stronger model. See details in Section 4.3.

We also prove lower bounds in the quantum samples model (i.e., Model (iii)). This model is far stronger than the rest of the models, and in particular, it trivially admits $O(1)$-quantum-sample algorithms for problems such as uniformity testing, identity testing, and gap-support size testing, which are known to be hard in the other models. In the quantum samples model, we are able to prove a weak lower bound of $\Omega(\sqrt{\log n})$ by a reduction to the promise problem of testing identity of two known distributions in Hellinger distance [Bel19]. To our knowledge, this constitutes the first non-trivial lower bound on the capability of this powerful input model. See details in Section 4.2.

1.1.4. Additive approximation and gap problems

Multiplicative approximation can generally also capture the notion of additive approximation, and in particular for entropies we can recover ϵ-additive estimates by suitably choosing the multiplicative factor γ, while incurring only a small (logarithmic) overhead in the complexity. Multiplicative estimation has the added advantage of being closely related to the field of property testing, wherein we wish to test whether an input satisfies some global property (such as having high entropy) or is far from any possible input that has that property (say, in total variation distance). To illustrate the generality and utility of our results, we note the following immediate applications to additive estimation and testing.

Estimating the entropy to additive precision: Since both Shannon and von Neumann entropies of n-dimensional distributions or quantum systems are bounded by $\log n$, choosing $\gamma = 1 + \frac{\epsilon}{\log n}$ we see that a good γ-multiplicative approximation also yields a good ϵ-additive approximation (see Section 2.1 for more details). Furthermore, the complexity overhead can be bounded by noting that

$$\begin{aligned} \frac{1}{2\gamma^2} &= \frac{1}{2}\left(1 + \frac{\epsilon}{\log n}\right)^{-2} \\ &\leq \frac{1}{2} + \frac{3\epsilon^2}{\log^2 n}. \end{aligned} \tag{1.1}$$

Since the second term decays and is $o(1)$, we recover query complexities of $\mathcal{O}\left(n^{\frac{1}{2}+o(1)}\right)$ and $\mathcal{O}\left(n^{1+o(1)}\right)$ for estimating the Shannon entropy of a probability distribution or the von Neumann entropy of a density matrix to constant additive precision. This matches the results of [GL20], upto to polylogarithmic factors (or equivalently, $n^{o(1)}$ factors).

Testing whether the entropy is high or low: Suppose we wish to determine if a distribution on $[n]$ has entropy (1) larger than a threshold H_1, or (2) smaller than a threshold $H_2 < H_1$ for some $H_1, H_2 \in (0, \log n]$. If we are able to γ-approximate

the entropy with $\gamma = \sqrt{\frac{H_1}{H_2}}$, notice that in the first case the algorithm must output a value larger than $\sqrt{H_1 H_2}$, whereas in the second case it must output a value smaller than $\sqrt{H_1 H_2}$, hence allowing us to distinguish the two cases. Thus, we can solve this testing problem with nearly subquadratic quantum query complexity $\widetilde{\mathcal{O}}\left(n^{\frac{H_2}{2H_1}}\right)$. To compare, classical algorithms can solve this task with $\mathcal{O}\left(n^{\frac{H_2}{H_1}+o(1)}\right)$ samples and require $\Omega\left(n^{\frac{H_2}{H_1}-o(1)}\right)$ samples [Val11].

1.2. Related work

For easy reference, we collect in Tables 1 and 2 the best known results on estimating Shannon and von Neumann entropies in input models of relevance to our work.

Type of estimate	**Classical sample complexity**	**Quantum query complexity**
ϵ-Additive	[JVH+15; WY16] $\Theta\left(\frac{n}{\epsilon \log n} + \frac{\log^2 n}{\epsilon^2}\right)$	[GL20; LW19] & [BKT18] $\widetilde{\mathcal{O}}\left(\frac{\sqrt{n}}{\epsilon^{1.5}}\right)$ & $\widetilde{\Omega}\left(\sqrt{n}\right)$
γ-Multiplicative	$\widetilde{\mathcal{O}}\left(n^{\frac{1+\eta}{\gamma^2}}\right)$ & $\Omega\left(n^{1/\gamma^2-o(1)}\right)$ [BDK+02] & [Val11]	$\widetilde{\mathcal{O}}\left(n^{\frac{1+\eta}{2\gamma^2}}\right)$ & $\Omega\left(n^{1/3\gamma^2}\right)$ (this work)

Table 1: Classical and quantum sample and query complexities of estimating the Shannon entropy of classical distributions over an alphabet of size n, and $\eta > 0$ controls the amount by which the entropy of the input is bounded away from zero (see Lemma 1 for details).

	Type of estimate	**Input model**	**Complexity**
[AIS+19]	ϵ-Additive	Copies of ρ	$\mathcal{O}\left(\frac{n^2}{\epsilon^2}\right)$ & $\Omega\left(\frac{n^2}{\epsilon}\right)$
[GL20]	ϵ-Additive	Purified quantum queries	$\widetilde{\mathcal{O}}\left(\frac{n}{\epsilon^{1.5}}\right)$
This work	γ-Multiplicative	Purified quantum queries	$\widetilde{\mathcal{O}}\left(n^{1/2+(1+\eta)/2\gamma^2}\right)$ & $\Omega\left(n^{1/3\gamma^2}\right)$

Table 2: Comparing works on estimating the von Neumann entropy of an n-dimensional density matrix.

We can group studies of entropy estimation into four categories: (1) classical and (2) quantum algorithms for estimating entropies of classical distributions; (3) classical and (4) quantum algorithms for estimating the entropies of quantum states.

We have already seen the most relevant works of the first kind in Section 1; it is worth remarking however that the estimation of entropies in a variety of classical input models and computational settings continues to be an active area of research.

We only note studies of the third category in passing: [HGK+10], for instance, discuss a quantum Monte Carlo method to measure the 2-Rényi entropy of a many-body system by evaluating the expectation value of a unitary swap operator.

At the intersection of categories (2) and (4), [AIS+19] study the sample complexity of estimating von Neumann and Renyi entropies of mixed states of quantum systems, in an input model where one gets access to m independent copies of an unknown n-dimensional density matrix ρ. They allow arbitrary quantum measurements and classical post-processing, and show that in general the number of quantum samples required scales as $\Theta(n^2)$, which is asymptotically the same as the number of samples that would be required to learn the state completely via tomography. The experimental measurement of the entropy of certain kinds of quantum systems has also recently been studied, for example in [IMP+15].

Other oracular input models may be potentially stronger than simple samples with measurement. The purified quantum query access model that we study in this work, wherein data is accessed in the form of a quantum state, is one such model. This state may be the output of some other quantum subroutine, in which case that subroutine itself is the oracle. Such input models can capture the fact that we have access to the process generating the unknown state, which we may *a priori* expect to be useful in reducing the effort required in estimating its properties.

With regard to quantum algorithms for estimating the entropies of quantum states (which subsumes the case of classical probability distributions), [LW19] provide upper and lower bounds on the query complexity for the task of additive approximation of von Neumann and Renyi entropies in the quantum frequency vector input model (see Eq. (2.7)). [GL20] study another similar oracular model, known as the quantum purified query access model, which essentially provides a pure state, sampling from which reproduces the statistics of the original mixed state, or target classical distribution (see Eqs. (2.9) and (2.10)). [SH21] consider the estimation of Renyi entropies in the same purified query access model, to both additive and multiplicative precision. Their focus however is on how this task may be solved on restricted models of quantum computation (namely, DQC1), and they do not obtain optimal query complexities.

Finally, we remark that we use the approximation of the logarithm by power functions that is defined and studied in [ZLO+07], who show that the Shannon entropy can be estimated to any desired precision by interpolation using estimates of Rényi-α entropies for values of $\alpha \in (0, 2]$. They study a streaming input model and use techniques that are otherwise very different from ours.

2. Preliminaries and notation

We assume the reader is familiar with the quantum computing framework and notation, such as Dirac's bra-ket notation. We refer to standard texts such as [NC10; Wol19] for a detailed introduction to quantum computation. Here, we discuss concepts and notation of specific relevance to this paper.

For $n \in \mathbb{N}$ we denote by $[n]$ the set $\{1, \ldots, n\}$. All logarithms that we use throughout this paper are taken with base 2. For a probability distribution $\mathbf{p} = (p_1, \ldots, p_n)$ on $[n]$, we use the notation $\mathrm{w}_{\mathbf{p}}(A) := \sum_{i \in A} p_i$ for the weight of a subset of labels $A \subseteq [n]$.

The Shannon entropy H of $\mathbf{p}$ is defined by [Sha48]

$$H(\mathbf{p}) := -\sum_{i \in [n]} p_i \log p_i. \tag{2.1}$$

We will let $H_{\mathbf{p}}(A) := -\sum_{i \in A} p_i \log p_i$ denote the entropy of the set of labels $A \subseteq [n]$ under the distribution $\mathbf{p}$.

Quantum registers can also exist in probabilistic mixtures of states; the simpler superposition states are called pure states, and their probabilistic mixtures are known as mixed states. The most general description of an n-dimensional quantum state ρ is in terms of an $n \times n$ positive semi-definite matrix with complex entries, normalised to have unit trace. The von Neumann entropy of a quantum state represented by its density matrix $\rho \in \mathbb{C}^{n \times n}$ is defined by [MDS+13]

$$S(\rho) := -\operatorname{Tr}(\rho \log \rho). \tag{2.2}$$

For two distributions $\mathbf{p}$ and $\tilde{\mathbf{p}}$ on $[n]$, we define the Hellinger distance between them by

$$d_H(\mathbf{p}, \tilde{\mathbf{p}}) := \sqrt{\frac{1}{2} \sum_i (\sqrt{p_i} - \sqrt{\tilde{p}_i})^2}, \tag{2.3}$$

and the total variation distance by

$$d_{\mathrm{TV}}(\mathbf{p}, \tilde{\mathbf{p}}) := \frac{1}{2} \sum_i |p_i - \tilde{p}_i|. \tag{2.4}$$

We use the standard complexity theoretic notation of $\widetilde{\mathcal{O}}$ to hide polylogarithmic factors. Finally, all our algorithms succeed with constant probability which can be boosted by standard techniques, and we omit the resulting factors from the complexity and discussion for brevity.

2.1. Multiplicative and additive estimates

A good estimate $\tilde{x}$ of some unknown quantity $x > 0$ to a multiplicative factor $\gamma > 1$ satisfies

$$\frac{x}{\gamma} \leq \tilde{x} \leq \gamma x. \tag{2.5}$$

Similarly, an estimate $\tilde{x}$ of an unknown quantity x to additive precision $\epsilon > 0$ must satisfy

$$x - \epsilon \leq \tilde{x} \leq x + \epsilon. \tag{2.6}$$

When we know an upper bound $0 < X < \infty$ on x, we can obtain an ϵ-additive approximation from a γ-multiplicative approximation by choosing $\gamma = 1 + \frac{\epsilon}{X}$, since

$$x - \epsilon < \left(1 - \frac{\epsilon}{X}\right) x < \left(1 + \frac{\epsilon}{X}\right)^{-1} x < \tilde{x} < \left(1 + \frac{\epsilon}{X}\right) x < x + \epsilon.$$

In particular, we know that the entropy of any distribution over $[n]$ is bounded above by $\log n$, and so by choosing $\gamma = 1 + \frac{\epsilon}{\log n}$ we can always obtain good additive estimates using multiplicative estimation subroutines.

2.2. Input models

We now formally define the four input models touched upon in Section 1. We refer to [Bel19] for a more detailed discussion of these models and their relations to each other.

(i) **Frequency vectors with quantum query access:** A standard unitary quantum query oracle U to a string $\mathbf{v} \in [n]^m$ for some large m, where $\forall\ j \in [m]$, $i \in [n]$, and

$$\begin{aligned} U\left|i\right\rangle\left|0\right\rangle &= \left|i\right\rangle\left|v_i\right\rangle \\ p_i &= \frac{1}{n}\left|\{j | v_j = i\}\right|. \end{aligned} \tag{2.7}$$

(ii) **Quantum oracle that generates a sample from p:** A standard unitary quantum query oracle to a string $\mathbf{v} \in [n]^m$ for some large m, where each v_i is drawn independently at random according to the distribution $\mathbf{p}$.

(iii) **Quantum samples for classical distributions**: A unitary that prepares the state

$$U_{\mathbf{p}}\left|0\right\rangle = \sum_{i=0}^{d-1} \sqrt{p_i}\left|i\right\rangle := \left|\mathbf{p}\right\rangle, \tag{2.8}$$

such that measuring the state $\left|\mathbf{p}\right\rangle$ in the computational basis reproduces the effect of sampling from $\mathbf{p}$.

(iv) **Purified quantum query access**: A unitary U_ρ on $\mathbb{C}^n \otimes \mathbb{C}^n$ which produces a purification $|\psi_\rho\rangle$ of the actual input state ρ in $\mathbb{C}^{n\times n}$

$$U_\rho |0\rangle = \sum_{i=1}^{n} \sqrt{p_i} |\psi_i\rangle |\phi_i\rangle := |\psi_\rho\rangle \tag{2.9}$$

such that the partial trace over the ancillary register $\mathrm{Tr}_2 (|\psi_\rho\rangle\langle\psi_\rho|) = \rho$. The states $\{|\psi\rangle\}$ and $\{|\phi\rangle\}$ are sets of orthonormal vectors on the system and ancillary subspaces respectively. Classical probability distributions can be considered as density matrices that are diagonal in the computational basis, so we consider a unitary $U_{\mathbf{p}}$ with a simplified action

$$U_{\mathbf{p}} |0\rangle = \sum_{i=1}^{n} \sqrt{p_i} |\psi_i\rangle |i\rangle := |\psi_{\mathbf{p}}\rangle . \tag{2.10}$$

In all four cases, as is standard in the theory of quantum query complexity, we assume access to both the oracle and its conjugate $U^\dagger$.

The frequency vector model can emulate the purified access model for classical distributions, i.e., given a frequency vector oracle, we can construct a purified access oracle with a single query:

$$\begin{aligned} U\left(\frac{1}{\sqrt{m}}\sum_{j=1}^{m}|j\rangle|0\rangle\right) &= \frac{1}{\sqrt{m}}\sum_{j=1}^{m}|j\rangle|v_j\rangle \\ &= \sum_{i=1}^{n}\left(\frac{\sum_{j:v_j=i}|j\rangle}{\sqrt{m}}\right)|i\rangle \\ &= \sum_{i=1}^{n}\sqrt{p_i}|\psi_i\rangle|i\rangle , \end{aligned} \tag{2.11}$$

using the definition of the frequency vector and defining the normalised version of the state in parenthesis on the second line. A similar calculation shows model (ii) can also emulate model (iv).

The vanilla quantum samples model for access to classical distributions too can emulate the purified query access model: applying a single layer of $\log n$ two-qubit CNOT gates to $|\mathbf{p}\rangle$ suffices to copy the computational basis states $|i\rangle$ into an ancillary register, reproducing the action in (2.10) for classical distributions.

It is also worth noting that analogous to model (iii), for the case of mixed quantum states we may have access to multiple independent copies of the unknown state ρ, which is the model studied in works including [AIS+19; OW15].

2.3. Algorithmic tools

The main standard quantum algorithmic techniques that we use are the method of quantum singular value transformations (QSVT), quantum singular value estimation (QSVE), and quantum amplitude estimation (QAE). We give a brief and high level overview of these methods here.

QSVT: The QSVT [GSL+19] is a powerful framework for describing and constructing quantum algorithms. At its heart lies the linear algebraic formulation of quantum algorithms, and the theory of quantum walks. Given a unitary U in $a+s$ dimensions that in a certain way encodes a possibly rectangular matrix P with singular value decomposition $\sum_i \sigma_i |v_i\rangle\langle w_i|$, the map QSVT(P,f) uses U as a black box and implements a unitary quantum circuit that approximately encodes a matrix P_f with singular values $f(\sigma_i)$ transformed according to a polynomial f defined on the values σ_i. This framework has been found to be immensely general in scope, in that most known quantum algorithms can be recast in terms of a QSVT. We will use this technique explicitly for the estimating the contribution to the entropy from points with high probability mass, in Section 3.2. We discuss more details regarding the QSVT of relevance to our work in Appendix B.

QSVE: The QSVE [KP20; CGJ19; GSL+19] is a generalisation of the popular and fundamental quantum phase estimation algorithm, generalising it to the task of coherently (i.e., in superposition) estimating the singular values of rectangular matrices. We use QSVE as a subroutine in Sections 3.1 and 3.2 for coherently distinguishing points with high probability mass from those with low probability mass. The subroutine QSVE(P,m) uses U as a black box and maps an input state $\sum_i \alpha_i |v_i\rangle |0\rangle$ to $\sum_i \alpha_i |v_i\rangle |\tilde{\sigma}_i\rangle$, where the $\tilde{\sigma}_i$ approximate the singular values σ_i of P to m bits of precision. We give more details and a discussion on the relation between QPE and QSVE in Appendix C.

QAE: Being a subroutine that grew out of Grover's search algorithm and the associated amplitude amplification technique, QAE [BHM+02] estimates the amplitude a quantum state $|\psi\rangle = U|0\rangle$ puts on a particular flagged subspace, essentially by running QPE on a Grover iterate (or diffusion operator) constructed from the unitary U. The map QAE(flag, ϵ) takes U as input and outputs an ϵ-additive estimate $\tilde{p}$ where $|\psi\rangle = \sqrt{p}\,|\text{flag}\rangle + \sqrt{1-p}\,|\text{junk}\rangle$. We use this technique to estimate various quantities obtained after processing by QSVT and QSVE in both Algorithms 1 and 2. For convenience we recall the standard formal statement of how QAE works in Appendix D.

3. Estimating entropy to multiplicative precision using purified quantum queries

In this section, we prove Theorems 1 and 2 by constructing our quantum algorithms for estimating Shannon and von Neumann entropies, and analysing their correctness and purified-quantum-query complexity. In fact, we prove the following, more general lemma, which can, with simple modifications, handle multiplicative approximation of any entropic functional $f : \mathcal{D} \to \mathbb{R}$ defined on vectors in $\mathcal{D} := \mathbb{R}^n_{\geq 0}$, and may be of more general

interest in distributional property testing, as well as in the context of space-bounded computation and streaming input models.

To formally state the lemma, we shall need the following notion of projected unitary encodings (which we will loosely and interchangeably refer to as block encodings), as defined in [GSL+19].

Definition 1. *An (α, a, δ) projected unitary encoding of an operator A acting on s qubits is a unitary U acting on $a + s$ qubits, such that*

$$\|A - \alpha \Pi^\dagger U \Pi\| \leq \epsilon, \tag{3.1}$$

where the first register consists of ancillary qubits, $\Pi := |0\rangle^{\otimes a} \otimes \mathbb{1}_s$ is an isometry mapping $(\mathbb{C}^2)^{\otimes s} \mapsto \mathrm{span}_{\mathbb{C}}\{|0\rangle^{\otimes a}\} \otimes (\mathbb{C}^2)^{\otimes s}$, and $\alpha, \epsilon \in (0, \infty)$.

We will prove the following lemma for projected unitary encodings. Subsequently, using the techniques of [GL20] to obtain projected unitary encodings from purified access oracles corresponding to classical distributions and density matrices respectively (see Appendix A), we will obtain Theorems 1 and 2 as immediate corollaries.

Lemma 1. *For any $\gamma > 1$ and $0 < \epsilon < 1$, given an (α, a, δ) projected unitary encoding U of a matrix P with singular values $\sqrt{p_1}, \ldots, \sqrt{p_n}$ where $\mathbf{p} = (p_1, \ldots, p_n)$ defines a probability distribution, there is a quantum algorithm which outputs with high probability a $(1 + 2\epsilon)\gamma$-multiplicative estimate $\tilde{H}$ of the entropy of $H(\mathbf{p})$, for distributions with entropy at least $3\gamma + 1/2\epsilon$. This algorithm makes*

$$m = \widetilde{\mathcal{O}}\left(\frac{\alpha n^{1/2\gamma^2}}{\epsilon}\right)$$

uses of U and $U^\dagger$, $\mathcal{O}(1)$ uses of controlled-U, and needs $\mathcal{O}(ma)$ additional one- and two-qubit gates.

Remark 1. The statement of this result is in direct analogy with [BDK+02, Theorem 1]. In particular, in order for the algorithm to be correct we need the entropy of the input to be bounded away from zero, since no algorithm using any of the input models discussed in Section 2.2 can output multiplicative estimates of arbitrary distributions, as we will see in Section 4.1. If we desire a multiplicative factor of γ, we can first choose $\gamma' = \frac{\gamma}{(1+2\epsilon)}$. This leads to a small overhead in the complexity scaling as $\mathcal{O}\left(n^{8\epsilon/\gamma^2}\right)$. Given $\eta > 0$ we can rephrase this as saying that by choosing $\epsilon < \eta/8$, the algorithm can deal with any distribution with $H(\mathbf{p}) = \Omega(1/\eta)$ using $\widetilde{\mathcal{O}}\left(n^{\frac{1+\eta}{\gamma^2}}\right)$ queries — that is, we can weaken the promise on the input and enlarge the class of distributions that the algorithm is correct on by paying a small appropriate price in the complexity.

Remark 2. The $\widetilde{\mathcal{O}}$ in Lemma 1 hides factors that scale as (a) $\mathcal{O}(\log^2 n)$; (b) $\log n/\epsilon$; and (c) $\mathcal{O}\left(\sqrt{\gamma^3}/\log\gamma\right)$. It turns out that with a purified quantum query access oracle, we can always create an exact encoding U of the distribution or quantum state with $\delta = 0$, and

so we will avoid discussing the dependence of the complexity on δ to avoid clutter.

In particular, Lemma 1 captures both the case of Shannon entropy of classical probability distributions and von Neumann entropy density matrices, since the latter is definitionally the Shannon entropy of the eigenvalue spectrum of the density matrix (see Section 3.4 for more details). We devote the rest of this section to proving Lemma 1.

The theory behind our estimator is drawn from [BDK+02]. We show how the estimator used therein can be computed more efficiently on a quantum computer with purified query access to the input. Recall that the Shannon entropy is defined by Eq. (2.1)

$$H(\mathbf{p}) = \sum_i p_i \log \frac{1}{p_i}.$$

To estimate $H(\mathbf{p})$, we first divide the domain into two sets, of 'big' and 'small' elements with respect to a choice of threshold $\beta \in (0, 1)$:

$$B := B_\beta = \{i \in [n] : p_i \geq \beta\} ,$$

and

$$S := S_\beta = [n] \setminus B_\beta .$$

Then $H(\mathbf{p}) = H_{\mathbf{p}}(B) + H_{\mathbf{p}}(S)$, since the Shannon entropy is linear as a function of subsets of its domain. We will aim to make the threshold value β as large as possible, and in particular, we would like it to scale inverse sublinearly as a function of n.

We start by considering the lightweight elements first.

3.1. Estimating the entropy of the low weight elements

The set S_β of light elements can contribute heavily to the entropy as evidenced for instance by the uniform distribution. However, the low probability mass of these elements can be hard to estimate. If the (unknown) weight of these elements is some $\mathrm{w}_{\mathbf{p}}(S)$, Lemma 4 of [BDK+02] gives us a way to handle S_β.

Lemma 2. $\mathrm{w}_{\mathbf{p}}(S) \cdot \log 1/\beta \leq H_{\mathbf{p}}(S) \leq \mathrm{w}_{\mathbf{p}}(S) \cdot \log n + \frac{1}{\mathrm{e}}$.

Proof. The lower bound is attained by a distribution that has as many points as possible with extremal weight (equal to β or 0), e.g., by a distribution with $\frac{1}{\beta} \cdot \mathrm{w}_{\mathbf{p}}(S)$ elements having probability mass β and the rest being zero.

The upper bound is given by the distribution that puts the weight $\mathrm{w}_{\mathbf{p}}(S)$ uniformly on all of its n points. □

This enables us to simply estimate the total weight of the light elements, and use this to improve our approximation to $H(\mathbf{p})$, as we shall see below in Section 3.3. For the moment, we note that for the choice of threshold $\beta = n^{-1/\gamma^2}$, Lemma 2 tightly bounds

the entropy of the lightweight elements as lying between their net weight times $\frac{\log n}{\gamma^2}$ and the weight times $\log n$ (up to a small constant).

Furthermore, we can also see from the upper bound in Lemma 2 that unless the net weight of the light elements scales at least inverse logarithmically in n, the contribution of the light set S_β to the entropy is bounded by a constant. In particular,

$$\mathrm{w}_{\mathbf{p}}(S) = o\left(\frac{1}{\log n}\right) \implies H_{\mathbf{p}}(S) = \frac{1}{e} + o(1). \tag{3.2}$$

Thus we may choose to estimate $\mathrm{w}_{\mathbf{p}}(S)$ to additive precision $\epsilon_1 = \mathcal{O}\left(1/\log^2 n\right)$. Given an (α, a, δ)-projected unitary encoding of P, intuition suggests that we can perform Quantum Phase Estimation (QPE) with this encoding to single out the lightweight elements and then estimate their net amplitude by performing Quantum Amplitude Estimation (QAE, Appendix D). This indeed turns out to work.

Algorithm 1 LightWeight(β) – Estimate $\mathrm{w}_{\mathbf{p}}(S)$ to additive precision ϵ_1.

1: $m = \log \frac{1}{\sqrt{\beta}}$ ▷ Number of bits of precision for QSVE

2: input $\leftarrow U_{\mathbf{p}} |0\rangle |0^m\rangle = \sum_{i\in[n]} \sqrt{p_i} |\phi_i\rangle |i\rangle |0^m\rangle$ ▷ Input state for QSVE

3: $\ldots \xrightarrow{\text{QSVE}(P,\ m)} \sum_{i\in[n]} \sqrt{p_i} |\phi_i\rangle |i\rangle |q_i\rangle$ ▷ $q_i = 0 \iff p_i < \beta$

$= \sum_{i\in S} \sqrt{p_i} |\phi_i\rangle |i\rangle |0^m\rangle_{\text{flag}} + \sum_{i\in B} \sqrt{p_i} |\phi_i\rangle |i\rangle |\perp\rangle_{\text{flag}}$ ▷ $\langle \perp | 0^m \rangle = 0$

4: Pick $\epsilon_1 = \epsilon/\log^2 n$ and let ▷ $\implies$ QAE cost $= \mathcal{O}\left(\log^2 n\right)$

$\tilde{w}_{\mathbf{p}}(S) = \text{QAE}(\text{flag} = |0^m\rangle, \epsilon_1)$ ▷ $|\tilde{w}_{\mathbf{p}}(S) - w_{\mathbf{p}}(S)| \le \epsilon_1$

5: **return** $\tilde{w}_{\mathbf{p}}(S)$

Using quantum singular value estimation to flag the light subspace: We would like to use QPE as a subroutine to separately flag the subspaces of heavy and light elements. In essence, we want to perform the map

$$\sum_{i\in[n]} \sqrt{p_i} |\phi_i\rangle |i\rangle \otimes |0^m\rangle \mapsto \sum_{i\in[n]} \sqrt{p_i} |\phi_i\rangle |i\rangle |q_i\rangle, \tag{3.3}$$

where $\left|\sqrt{p_i} - q_i\right| \le 2^{-(m+1)} =: \epsilon$, and m is the number of bits of precision. When given a unitary block encoding for the matrix P, this problem is termed Quantum Singular Value Estimation (QSVE) and is solved in [KP20; CGJ19; GSL+19]; the complexity of their algorithm is essentially $\widetilde{O}\left(1/\epsilon\right)$ where ϵ is the precision to which we would like to

estimate the singular values. We defer the full discussion of the intuition and details behind doing this to Appendix C.

The accuracy of the QPE subroutine is normally defined in terms of the number of bits of precision. Thus, to obtain a clean split between heavy and light elements according to the chosen threshold

$$\beta = n^{-\frac{1}{\gamma^2}} = 2^{-\frac{\log n}{\gamma^2}},$$

recalling that P has singular values $\sqrt{p_i}$, it will be convenient for us to first round $\sqrt{\beta}$ down to the nearest power of 2, and then scale down the approximation factor γ appropriately. To this end, we set

$$\sqrt{\beta'} = 2^{-\left\lceil \frac{\log n}{2\gamma^2} \right\rceil} =: n^{-\frac{1}{2\gamma'^2}},$$

which suggests that we should choose a tighter approximation factor, given by

$$\gamma' = \gamma \cdot \sqrt{\frac{\log n/2\gamma^2}{\lceil \log n/2\gamma^2 \rceil}}.$$

Note that $\gamma' \leq \gamma$, and so any good γ'-approximation is also a good γ-approximation. With this choice, we can run QSVE with the block encoding of P to $m = \log \frac{1}{\sqrt{\beta'}}$ bits of precision, implying an additive precision of $\epsilon = 2^{-(m+1)}$, incurring a complexity of

$$\begin{aligned} \mathcal{O}\left(\frac{1}{\epsilon}\right) &= \mathcal{O}\left(\frac{1}{\sqrt{\beta'}}\right) \\ &= \mathcal{O}\left(n^{\frac{1}{2\gamma'^2}}\right) = \mathcal{O}\left(n^{\frac{1}{2\gamma^2}}\right), \end{aligned} \tag{3.4}$$

where we have used that

$$\begin{aligned} \frac{1}{\gamma^2} \cdot \frac{\lceil \log n/2\gamma^2 \rceil}{\frac{\log n}{2\gamma^2}} &\leq \frac{1}{\gamma^2} \cdot \frac{\frac{\log n}{2\gamma^2} + 1}{\frac{\log n}{2\gamma^2}} \\ &\leq \frac{1}{\gamma^2}\left(1 + \frac{2\gamma^2}{\log n}\right), \end{aligned}$$

and $n^{1/\log n} = \mathcal{O}(1)$.

Correctness: With these steps in hand, one can readily see that Algorithm 1 outputs an additive estimate of the net weight of all the elements whose probability mass lies below the threshold β, i.e.,

$$|\tilde{\mathrm{w}}_{\mathbf{p}}(S) - \mathrm{w}_{\mathbf{p}}(S)| \leq \epsilon_1. \tag{3.5}$$

Complexity: With the reasoning of Eq. (3.2) and our choice of $\epsilon_1 = \epsilon/\log^2 n$, the query complexity of Algorithm 1 in terms of queries to the block encoding of P is given by

$$\mathcal{O}\left(\frac{\log^2 n}{\epsilon}\right) \cdot \mathcal{O}\left(\frac{1}{\sqrt{\beta}}\right), \tag{3.6}$$

where the first term is the complexity of the amplitude estimation step (Appendix D). For the choice $\beta = n^{-1/\gamma^2}$, this becomes

$$\mathcal{O}\left(\frac{n^{1/2\gamma^2} \log^2 n}{\epsilon}\right). \tag{3.7}$$

Intuitively, as previously noted in [GL20], the projected unitary encoding of the input gives us *operational access* to the square roots of the point probabilities. Hence, since $p_i < \sqrt{p_i}$, quantum algorithms can in a sense 'see' lightweight elements more easily than classical algorithms.

The normalisation α of the block encoding: So far we have not discussed how the normalisation factor $\alpha \geq 1$ and precision $\delta \in (0, 1)$ of the (α, a, δ) projected unitary encoding U of P affects the algorithm and its complexity. This is, fortunately, easy to do. First we note that the states the purified access oracles generate, both in Eqs. (2.9) and (2.10), actually encodes *the exact* values of $\sqrt{p_i}$ and require no additional normalisation factor. Thus the only step where α and δ enter the picture are in the QSVE step. Next, as we remarked previously, the constructions that we use will have $\delta = 0$, so we ignore this precision, which even otherwise would only contribute logarithmic factors to the complexity. Finally, since the singular values of P are $\sqrt{p_i}$ and U encodes P/α, the normalisation factor effectively means that in the QSVE step we must work harder and estimate them to precision $\sqrt{\beta}/\alpha$ — this directly contributes exactly a factor of α to the overall complexity in Eq. (3.7) of Algorithm 1.

This can also be seen directly from the complexity of QSVE in [CGJ19, Theorem 27], which scales as $\mathcal{O}(\alpha/\epsilon)$ for performing QSVE to precision ϵ using an (α, a, δ) block encoding.

3.2. Estimating the entropy $H_{\mathbf{p}}(B)$ of the heavy elements

So far we have looked at the set S_β of lightweight elements. We now turn to estimating the contribution to the entropy from the heavy elements, i.e., those with probability mass greater than the threshold value of β.

The first ingredient we use for this is a result from classical approximation theory that provides a multiplicative approximation of the logarithm by a combination of functions of the form $x^{\pm a}$ for small values of $a > 0$ [ZLO+07].

3.2.1. Multiplicative approximation of entropy using power functions

For any $a \in (0,1)$, consider the following functions

$$f_{\pm}(x) = x^{\pm a}, \qquad f(x) = -\frac{f_+(x) - f_-(x)}{2a}. \tag{3.8}$$

On a domain $(\beta, 1]$ for some $\beta > 0$, we have the following series expansions for $f_{\pm}(x)$

$$\begin{aligned} x^{\pm a} &= e^{\pm a \log x} \\ &= 1 \pm a \log x + \frac{(a \log x)^2}{2!} \pm \frac{(a \log x)^3}{3!} + \dots, \end{aligned} \tag{3.9}$$

from which we deduce that

$$f(x) = -\log x \cdot \left(1 + \frac{(a \log x)^2}{3!} + \frac{(a \log x)^4}{5!} + \dots\right). \tag{3.10}$$

Since $\log x < 0$ on our interval of interest $x \in (\beta, 1]$, it will be convenient to write $-\log x = \log 1/x > 0$. We see that $f(x)$ is a one sided approximation for $\log x$, in the sense that it is always at least $\log 1/x$, and at most as large as $\log 1/x \cdot g(a,x)$, where the multiplicative factor is

$$\begin{aligned} g(a,x) &= 1 + \frac{(a \log x)^2}{3!} + \frac{(a \log x)^4}{5!} + \dots \\ &\le 1 + |a \log x| + \frac{(a \log x)^2}{2!} + \frac{|a \log x|^3}{3!} + \frac{(a \log x)^4}{4!} + \dots \\ &= \mathrm{e}^{|a \log x|}. \end{aligned} \tag{3.11}$$

Note that $\forall x \in (\beta, 1]$, $g(a,x) \le g(a,\beta)$. Thus if we would like $f(x)$ to approximate $\log x$ to within a multiplicative factor $\gamma > 1$ in the sense of Eq. (2.5), it suffices to choose $a \in (0,1)$ such that

$$g(a,\beta) \le \mathrm{e}^{a \log 1/\beta} \le \gamma \,,$$

which translates to requiring that

$$a \le \frac{\log \gamma}{\log 1/\beta}. \tag{3.12}$$

Choosing such an a, we have a γ-multiplicative approximation of the logarithm for $x \in (\beta, 1]$, i.e.,

$$\frac{1}{\gamma} \cdot \log 1/x \le \log 1/x \le f(x) \le \gamma \cdot \log 1/x. \tag{3.13}$$

It is worth remarking that this is actually *stronger* than a γ-multiplicative approximation, by virtue of being one-sided; the lower bound on the approximation is actually as strong as

$\log 1/x$, without the scaling by $1/\gamma$. We will use this stronger inequality in our calculations below and freely replace it with the weaker version where required.

For $i \in B_\beta$, if we replace the logarithm of p_i with the function f, we get a multiplicative approximation of $H_{\mathbf{p}}(B)$. Indeed by Eq. (3.13), $\forall p_i \in (\beta, 1]$

$$\log \frac{1}{p_i} \leq f(p_i) \leq \gamma \cdot \log \frac{1}{p_i}. \tag{3.14}$$

Multiplying by p_i and summing over $i \in B_\beta$, we have

$$\sum_{i \in B} p_i \log \frac{1}{p_i} \leq \sum_{i \in B} p_i f(p_i) \leq \gamma \cdot \sum_{i \in B} p_i \log \frac{1}{p_i}, \tag{3.15}$$

and so we have

$$H_{\mathbf{p}}(B) \leq \sum_{i \in B} p_i f(p_i) \leq \gamma \cdot H_{\mathbf{p}}(B)\ . \tag{3.16}$$

We hence see that in fact for any $B \subseteq [n]$, $\sum_{i \in B} p_i f(p_i)$ is a good γ-multiplicative approximation to $H_{\mathbf{p}}(B)$. Next, we look at how to estimate the sum over $p_i f(p_i)$ by using the quantum singular value transformations (QSVT) technique [GSL+19] of implementing functions of block encoded matrices on quantum computers (see also [CGJ19; SBJ19]).

3.2.2. Using QSVT to estimate power sums

Defining the following power sums with exponent a,

$$F_\pm = \sum_{i \in B} p_i^{1 \pm a} = \sum_{i \in B} p_i f_\pm(p_i), \tag{3.17}$$

we see that our estimator for $H_{\mathbf{p}}(B)$ in Eq. (3.16) takes the form

$$F = -\frac{F_+ - F_-}{2a}. \tag{3.18}$$

This suggests a strategy of estimating $F_\pm$ separately and combining them to obtain F. Ideally, we would have liked to prepare two states, which have squared amplitude equal to $F_\pm$ on a subspace flagged by $|0\rangle$ in the ancilla. One possible form such states might take is

$$\sum_i \sqrt{p_i} f_\pm(\sqrt{p_i}) \, |\psi_i\rangle \, |i\rangle \, |0\rangle + |\text{junk}\rangle \, |1\rangle\ .$$

Since we cannot directly implement arbitrary power functions $x^{\pm a}$, we use the standard technique of implementing the quantum singular value transformation corresponding to polynomial approximations $\tilde{f}_\pm(x)$ of $f_\pm(x)$ over the domain specified by B; recalling that we have a block encoding of P whose singular value spectrum encodes $\sqrt{p_i}$, we only need to work with the domain $x \in [\sqrt{\beta}, 1]$.

We discuss the details behind constructing and implementing the polynomials $\tilde{f}_\pm(x)$ in Appendix B. Intuitively, polynomial approximations using Taylor series give an exponential convergence in the degree of the approximating polynomial for smooth functions, i.e., the degree of the approximating polynomial only needs to grow as $\log 1/\epsilon$. Since the query complexity of QSVT depends on the degree of the polynomial being implemented, this in conjunction with what we noted above about the domain of approximation being $[\sqrt{\beta}, 1]$ leads us to expect the net query complexity of estimating $F_\pm$ to grow as $\mathcal{O}\left(1/\sqrt{\beta}\right)$. We will show below that this is indeed the case.

Algorithm 2 HEAVYENTROPY(β) – Estimate $H_{\mathbf{p}}(B)$ to multiplicative factor $\gamma > 1$.

1: input $\leftarrow U_{\mathbf{p}} |0\rangle |0\rangle = \sum_{i\in[n]} \sqrt{p_i} |\phi_i\rangle |i\rangle |0\rangle$ ▷ Input state for QSVT

2: $\ldots \xrightarrow{\text{QSVT}(P,\ \tilde{f}_\pm)} \sum_{i\in[n]} \sqrt{p_i}\tilde{f}_\pm(\sqrt{p_i}) |\phi_i\rangle |i\rangle |0\rangle + |\text{junk}\rangle |1\rangle$ ▷ QSVT - Appendix B

3: $\ldots \xrightarrow{\text{QSVE}(P,\ m)} \sum_{i\in B} \sqrt{p_i}\tilde{f}_\pm(\sqrt{p_i}) |\phi_i\rangle |i\rangle |0\rangle |0\rangle_{\text{flag}} + |\text{junk}\rangle |1\rangle_{\text{flag}}$

▷ flag $= 0 \iff p_i \geq \beta$

4: Pick ϵ_3 as in Eq. (3.26) and let

$\tilde{F}_\pm = \text{QAE}(\text{flag} = |0\rangle, \epsilon_3)$ ▷ $\implies$ QAE cost $= \mathcal{O}\left(\frac{1}{\epsilon_3}\right)$

5: **return** $\tilde{F} = -\frac{2\tilde{F}_+ - 2\sqrt{\gamma}\tilde{F}_-}{2a}$

Some difficulties: In constructing polynomial approximations on our subdomain $[\sqrt{\beta}, 1]$ of interest, we need to admit a small interval $[\sqrt{\beta}/2, \sqrt{\beta}]$ where we allow the polynomial to vary before falling to low values on the rest of the domain $[-1, 1]$. It is in accordance with this intuition that we have the guarantees $|f_\pm(x)| \leq 1$ on $[0, 1]$, and $|f(x)| \leq \epsilon$ on $[0, \sqrt{\beta}/2)$ in Appendix B. If we naïvely try to use the simple state produced by applying the QSVT for $\tilde{f}_\pm$ as in step 2 of Algorithm 2 to estimate $F_\pm$, we see that the amplitude of the part of the state flagged by zero is actually given by

$$\sum_{i=1}^{n} p_i \tilde{f}_\pm^2(\sqrt{p_i}) = \sum_{i\in B_\beta} p_i \tilde{f}_\pm^2(\sqrt{p_i}) + \sum_{i\in[\beta/2,\beta]} p_i \tilde{f}_\pm^2(\sqrt{p_i}) + \sum_{i\in[0,\beta/2]} p_i \tilde{f}_\pm^2(\sqrt{p_i}). \tag{3.19}$$

In particular, the second term on the rhs above is undesirable and in addition not easy to control. While we can upper bound the error caused by this term, we cannot manage it well without increasing the degree of the polynomial and hence incurring overheads in the query complexity.

Our approach: It may in principle be possible to modify the approximating polynomial sufficiently with only polylogarithmic overheads in the degree, but we do not explore this route; instead we once again invoke QSVE as a tool to split the heavy and light weight subspaces. We may repeat the arguments of the previous section, with the slight modification of flagging the heavy elements with $|0\rangle$ in a single qubit ancilla, say by applying a unitary comparator circuit to check if the estimated singular value $q_i > \sqrt{\beta}$.

Correctness: Thus, generating states of the form given in step 3 of Algorithm 2, we see that their amplitudes in the flagged subspace now exactly square to

$$\tilde{F}_{\pm} = \sum_{i \in B} p_i \tilde{f}_{\pm}^2(\sqrt{p_i}), \tag{3.20}$$

and taking into account the normalisation factors corresponding to $\tilde{f}_{\pm}$, we define

$$\tilde{F} = -\frac{2\tilde{F}_{+} - 2\sqrt{\gamma}\tilde{F}_{-}}{2a}, \tag{3.21}$$

where in particular, by the guarantees on the polynomial approximations Appendix B we know that $\forall x \in [\sqrt{\beta}, 1]$

$$\begin{aligned} \left| \tilde{f}_{-}(x) - \frac{\sqrt{\beta}^a}{2} x^{-a} \right| &\leq \epsilon_2 \\ \left| \tilde{f}_{+}(x) - \frac{x^a}{2} \right| &\leq \epsilon_2. \end{aligned} \tag{3.22}$$

Recalling that

$$a = \frac{\log \gamma}{\log 1/\beta},$$

we see that the normalisation factor for f_{-} is given by

$$\beta^{a/2} = \mathrm{e}^{\log \beta \cdot \frac{\log \gamma}{-2 \log \beta}} = \frac{1}{\sqrt{\gamma}}.$$

From these approximation guarantees we immediately see that

$$\begin{aligned} \left| F_{\pm} - \tilde{F}_{\pm} \right| &\leq n\epsilon_2 \\ \left| F - \tilde{F} \right| &\leq \frac{\sqrt{\gamma} n \epsilon_2}{a} \end{aligned} \tag{3.23}$$

Furthermore, the QAE steps for $\tilde{F}_{\pm}$ are performed to some precision ϵ_3, yielding $\hat{F}_{\pm}$ which satisfy

$$\left| \hat{F}_{\pm} - \tilde{F}_{\pm} \right| \leq \epsilon_3, \tag{3.24}$$

and so we also have the analogous quantity $\hat{F}$ such that

$$\left|\hat{F} - \tilde{F}\right| \leq \frac{2\sqrt{\gamma}\epsilon_3}{a} \tag{3.25}$$

As indicated by the above expressions for the error, let us then choose the precision of the polynomial approximation and QAE steps such that the errors above are of constant order, i.e.,

$$\begin{aligned} \epsilon_2 &= \frac{a}{2n\sqrt{\gamma}} \cdot \epsilon = \frac{\log \gamma}{2\sqrt{\gamma} n \log 1/\beta} \cdot \epsilon \\ \epsilon_3 &= \frac{a}{4\sqrt{\gamma}} \cdot \epsilon = \frac{\log \gamma}{4\sqrt{\gamma} \log 1/\beta} \cdot \epsilon. \end{aligned} \tag{3.26}$$

This ensures that $\hat{F}$ is a good ϵ-additive approximation to F, since

$$\left|F - \hat{F}\right| \leq \left|F - \tilde{F}\right| + \left|\tilde{F} - \hat{F}\right| \leq \epsilon. \tag{3.27}$$

Since this means $F - \epsilon \leq \hat{F} \leq F + \epsilon$, recalling that F is a good γ-multiplicative approximation of $H_{\mathbf{p}}(B)$ as in Eq. (3.16), we arrive at the following multiplicative guarantee with some additive slack on our estimate $\hat{F} \equiv \tilde{H}_{\mathbf{p}}(B)$

$$\frac{H_{\mathbf{p}}(B)}{\gamma} - \epsilon \leq \hat{F} \leq \gamma H_{\mathbf{p}}(B) + \epsilon. \tag{3.28}$$

Complexity: The total query complexity of Algorithm 2 is the sum of the complexity of QSVT and QSVE corresponding to the polynomial approximations of $f(x) = x^{\pm a}$, multiplied by the complexity of QAE. The former two are respectively given by the degrees of $\tilde{f}_{\pm}$ from Appendix B and $\mathcal{O}(1/\sqrt{\beta})$, both of which scale as $\widetilde{\mathcal{O}}(1/\beta)$, while the latter is the inverse of ϵ_3. Thus, the net complexity is bounded by

$$\begin{aligned} &\left[\mathcal{O}\left(n^{1/2\gamma^2}\right) + \mathcal{O}\left(n^{1/2\gamma^2} \log \frac{n \log n}{\epsilon \log \gamma}\right)\right] \cdot \mathcal{O}\left(\frac{\log n}{\epsilon \log \gamma}\right) \\ &\qquad = \widetilde{\mathcal{O}}\left(\frac{n^{1/2\gamma^2} \log^2 n}{\epsilon \log \gamma}\right). \end{aligned} \tag{3.29}$$

The normalisation α of the block encoding: First, we essentially repeat the discussion at the end of Section 3.1, giving a factor of α in the complexity of the QSVE step in Algorithm 2.

In addition, we have to also account for the fact that the power functions are now implemented on the spectrum of P/α (for more details, see [GSL+19, Theorem 56]). This has two effects: firstly, the domain over which we construct the polynomial approximations $\tilde{f}_{\pm}$ should now be $[\sqrt{\beta}/\alpha, 1]$, which increases their degree by a factor of α. Secondly, we

obtain the power sums F_+ and F_- are divided and multiplied respectively by an extra α^a factor, which means we need to choose ϵ_2 and ϵ_3 to be smaller by this factor.

Since $\alpha = \frac{\log \gamma}{\log 1/\beta}$, for polynomially scaling normalisations $\alpha = n^c$ and $\beta = n^{-1/\gamma^2}$ we see that α^a scales as $\gamma^{c\gamma^2} = \mathcal{O}(1)$ for our purposes.

Hence since the complexities of the QSVT and QSVE steps add, we see once again that the net overhead in complexity is a factor of α.

3.3. Combining the heavy and light estimates to form $\tilde{H}$

Finally, we analyse the errors in and combine the estimates of $H_{\mathbf{p}}(B)$ and $H_{\mathbf{p}}(S)$ from Algorithms 1 and 2 to get an estimate of $H(\mathbf{p})$. Thus far, we have estimated the entropy

Algorithm 3 Approximating $H(\mathbf{p})$ to multiplicative precision $\gamma > 1$

1: $\gamma' \leftarrow \gamma \cdot \sqrt{\frac{\log n/\gamma^2}{\lceil \log n/\gamma^2 \rceil}}$ ▷ Scale approximation factor down

2: $\beta \leftarrow n^{-1/\gamma'^2}$ ▷ Choose threshold that splits heavy & light elements

3: $\tilde{H}_{\mathbf{p}}(B) \leftarrow \textsc{HeavyEntropy}(\beta)$ ▷ $\frac{H_{\mathbf{p}}(B)}{\gamma} - \epsilon \leq \tilde{H}_{\mathbf{p}}(B) \leq \gamma H_{\mathbf{p}}(B) + \epsilon$

4: $\tilde{w}_{\mathbf{p}}(S) \leftarrow \textsc{LightWeight}(\beta)$ ▷ $|\tilde{\mathrm{w}}_{\mathbf{p}}(S) - \mathrm{w}_{\mathbf{p}}(S)| \leq \epsilon_1$

5: **return** $\tilde{H}_{\mathbf{p}}(B) + \frac{\tilde{w}_{\mathbf{p}}(S) \log n}{\gamma'}$

of the heavy elements to multiplicative precision, viz Eq. (3.28), and we have estimated the total weight of the lightweight elements to additive precision Eq. (3.5). The estimate that our algorithm outputs for $H(\mathbf{p}) := H_{\mathbf{p}}([n])$ is the quantity $\tilde{H}$ defined by

$$\tilde{H} = \tilde{H}_{\mathbf{p}}(B) + \frac{\tilde{\mathrm{w}}_{\mathbf{p}}(S) \log n}{\gamma}. \tag{3.30}$$

Upper bounding $\tilde{H}$: Using the upper bounds on $\hat{F}$ and $\tilde{\mathrm{w}}_{\mathbf{p}}(S)$, we see that

$$\tilde{H} \leq \gamma H_{\mathbf{p}}(B) + \epsilon + \frac{\mathrm{w}_{\mathbf{p}}(S) + \epsilon_1}{\gamma} \cdot \log n. \tag{3.31}$$

The lower bound on $H_{\mathbf{p}}(S)$ in Lemma 2 along with the choice $\beta = n^{-1/\gamma^2}$ implies that

$$\frac{\mathrm{w}_{\mathbf{p}}(S) \log n}{\gamma} \leq \gamma H_{\mathbf{p}}(S). \tag{3.32}$$

Recalling that $\epsilon_1 = \epsilon/\log^2 n$, we thus have

$$\begin{aligned}\tilde{H} &\leq \gamma H_{\mathbf{p}}(B) + \gamma H_{\mathbf{p}}(S) + 2\epsilon \\ &\leq \gamma H(\mathbf{p}) + 2\epsilon \\ &\leq (1 + 2\epsilon)\,\gamma H(\mathbf{p}),\end{aligned} \tag{3.33}$$

where on the last line we assume that $H(\mathbf{p}) \geq \frac{1}{\gamma}$, i.e., that the entropy of the input distribution is bounded away from zero, with a small increase in the approximation factor. This is a reasonable assumption to make, since no algorithm can output a good γ-multiplicative approximation of *all* distributions (see Section 4.1 for more details).

Lower bounding $\tilde{H}$: This time using the lower bounds on $\hat{F}$ and $\tilde{\mathrm{w}}_{\mathbf{p}}(S)$, we see that

$$\tilde{H} \geq H_{\mathbf{p}}(B) - \epsilon + \frac{\mathrm{w}_{\mathbf{p}}(S) - \epsilon_1}{\gamma} \cdot \log n. \tag{3.34}$$

The upper bound on $H_{\mathbf{p}}(S)$ in Lemma 2 implies that

$$\frac{\mathrm{w}_{\mathbf{p}}(S) \log n}{\gamma} \geq \frac{H_{\mathbf{p}}(S) - \frac{1}{e}}{\gamma}. \tag{3.35}$$

Thus we have that

$$\begin{aligned}\tilde{H} &\geq H_{\mathbf{p}}(B) + \frac{H_{\mathbf{p}}(S)}{\gamma} - 2\epsilon - \frac{1}{e\gamma} \\ &\geq \frac{H(\mathbf{p})}{\gamma} - 2\epsilon - \frac{1}{\gamma} \\ &\geq \frac{H(\mathbf{p})}{(1 + 2\epsilon)\,\gamma},\end{aligned} \tag{3.36}$$

where to go from the second to the third line we assume that $H(\mathbf{p}) \geq 3\gamma + 1/2\epsilon \geq \frac{1}{\gamma}$, i.e., as before, that the entropy of the input distribution is bounded away from zero, with a small increase in the approximation factor.

This shows that Algorithm 3 outputs a $(1 + 2\epsilon)\gamma$-multiplicative approximation of $H(\mathbf{p})$ and thus establishes its correctness.

Complexity: Finally, the query complexity of Algorithm 3 is the sum of the complexities of steps (3) and (4). From the complexities Eqs. (3.7) and (3.29) of Algorithms 1 and 2 respectively, we have a net query complexity that scales as

$$\mathcal{O}\left(\frac{\alpha n^{1/2\gamma^2} \log^2 n}{\epsilon}\right) + \widetilde{\mathcal{O}}\left(\frac{\alpha n^{1/2\gamma^2} \log^2 n}{\epsilon \log \gamma}\right) = \widetilde{\mathcal{O}}\left(\frac{\alpha n^{1/2\gamma^2} \log^2 n}{\epsilon \log \gamma}\right), \tag{3.37}$$

which completes our proof of Lemma 1.

3.4. Proving Theorems 1 and 2

For classical probability distributions accessed via a purified quantum query oracle $U_{\mathbf{p}}$ as in Eq. (2.10), we can construct projected unitary encodings of the kind required in Lemma 1 with $\alpha = 1$ (see Appendix A). This immediately furnishes a proof of Theorem 1.

Similarly, for an arbitrary n-dimensional quantum density matrix accessed via a purified quantum query oracle U_ρ as in Eq. (2.9), we can construct a projected unitary encoding with $\alpha = \sqrt{n}$ (see Appendix A). Since $S(\rho)$ is equal to the Shannon entropy of the spectrum of ρ, we can plug this encoding into Lemma 1 to obtain a proof of Theorem 2.

The key difference between the case of classical distributions and quantum mixed states is that for the former, we know that the purified access oracle produces a superposition over computational basis states in the second register, and we can use this knowledge to our advantage. On the other hand, for the latter case we do not *a priori* know the basis in which the quantum state is diagonal, which reflects in the fact that we do not know the states $|\psi_i\rangle$ and $|\phi_i\rangle$ appearing in Eq. (2.9) beforehand.

4. Lower bounds

Batu et al. [BDK+02] proved that even if we restrict to distributions with entropy $H(\mathbf{p}) \geq \log n/\gamma^2$, any algorithm that estimates $H(\mathbf{p})$ within a multiplicative $\gamma > 1$ requires $\Omega(n^{1/2\gamma^2}) = \mathcal{O}(\sqrt{n})$ samples. By arguing about the fingerprints of samples drawn from the unknown distribution, for small approximation factors $\gamma \in (1, \sqrt{2})$ they were able to show a stronger lower bound of $\Omega(n^{2/(5\gamma^2-2)})$ samples, which is $o(n^{2/3})$, even when the input is known to be a distribution with $H(\mathbf{p}) \geq \frac{5\log n}{10\gamma^2-4}$. This was later improved by Valiant [Val11] to $\Omega(n^{1/\gamma^2-o(1)})$, which showed the original upper bound of Batu et al. to be essentially tight.

The intuition behind proving lower bounds is to notice that estimating the entropy to a suitable multiplicative factor can suffice to distinguish between a given pair of distributions $\mathbf{p}$ and $\tilde{\mathbf{p}}$. Recall that any γ-approximation algorithm must output an estimate $\tilde{H}$ such that

$$\begin{aligned} \frac{H(\mathbf{p})}{\gamma} \leq \tilde{H}(\mathbf{p}) \leq \gamma H(\mathbf{p}) \\ \frac{H(\tilde{\mathbf{p}})}{\gamma} \leq \tilde{H}(\tilde{\mathbf{p}}) \leq \gamma H(\tilde{\mathbf{p}}). \end{aligned} \tag{4.1}$$

If the ratio of entropies is larger than γ^2, then we have that

$$\tilde{H}(\tilde{\mathbf{p}}) \leq \gamma H(\tilde{\mathbf{p}}) \leq \frac{H(\mathbf{p})}{\gamma} \leq \tilde{H}(\mathbf{p}), \tag{4.2}$$

and so γ-estimating H will allow us to distinguish $\mathbf{p}$ and $\tilde{\mathbf{p}}$.

In this section we prove lower bounds similar to those of [BDK+02] but for the case of quantum algorithms that output a γ-multiplicative estimate of the Shannon entropy of a classical distribution. We prove our bounds for the quantum frequency vector model. Recalling that this model is capable of emulating both the purified access model and the model that quantumly queries a classical list of samples, these lower bounds carry over to both these models as well. The technique we use for this is a reduction from the collision problem to multiplicatively approximating entropy.

We also prove what is perhaps the first non-trivial lower bound for the vanilla quantum samples model by showing a reduction from the promise problem of distinguishing two classical distributions in Hellinger distance to γ-multiplicative approximation of entropy.

4.1. Distributions with non-zero entropy

We first show that no algorithm working with input models (i)-(iv) that makes only polynomially many queries can estimate the Shannon entropy of all distributions over $[n]$ to multiplicative precision. Consider $\mathbf{p} = (1-\epsilon, \frac{\epsilon}{n-1}, \ldots, \frac{\epsilon}{n-1})$ and $\tilde{\mathbf{p}} = (1, 0, \ldots, 0)$. The Hellinger distance between $\mathbf{p}$ and $\tilde{\mathbf{p}}$ is given by

$$\begin{aligned} d_H(\mathbf{p}, \tilde{\mathbf{p}}) &= \sqrt{\frac{1}{2}\sum_i (\sqrt{p_i} - \sqrt{\tilde{p}_i})^2} \\ &= \sqrt{1 - \sqrt{1-\epsilon}}. \end{aligned} \tag{4.3}$$

The binomial theorem tells us that for $|\epsilon| \leq 1$ and $\beta \in \mathbb{R}$,

$$(1-\epsilon)^\beta = 1 + \sum_{k=1}^{\infty} \frac{(\beta)(\beta-1)\ldots(\beta-k)}{k!}(-\epsilon)^k. \tag{4.4}$$

Therefore $1 - \epsilon < (1-\epsilon)^{1/2} < 1 - \epsilon/2$, and we have

$$\sqrt{\epsilon} \geq d_H(\mathbf{p}, \tilde{\mathbf{p}}) \geq \sqrt{\frac{\epsilon}{2}}, \tag{4.5}$$

i.e., $d_H(\mathbf{p}, \tilde{\mathbf{p}}) = \Theta(\sqrt{\epsilon})$.

We also have $H(\tilde{\mathbf{p}}) = 0$ and

$$\begin{aligned} H(\mathbf{p}) &= -(1-\epsilon)\log(1-\epsilon) - \epsilon \log \epsilon + \epsilon \log(n-1) \\ &= \Omega(\epsilon \log n), \end{aligned} \tag{4.6}$$

since $h(\epsilon) := -(1-\epsilon)\log(1-\epsilon) - \epsilon \log \epsilon \in [0, \log 2]$ is the binary entropy. Thus any algorithm that outputs an approximation for H to a multiplicative factor γ must output exactly 0 on input $\tilde{\mathbf{p}}$ and at least $\frac{\epsilon}{\gamma} \log n$ on input $\mathbf{p}$.

From [Bel19], we know that distinguishing $\mathbf{p}$ and $\tilde{\mathbf{p}}$ has query complexity

$$\Theta\left(\frac{1}{d_H(\mathbf{p},\tilde{\mathbf{p}})}\right),$$

in any of the four input models (i)-(iv). In particular, any algorithm requires $\Omega\left(1/\sqrt{\epsilon}\right)$ queries to distinguish $\mathbf{p}$ from $\tilde{\mathbf{p}}$. Picking $\epsilon = n^{-k}$, for $\forall k > 0$, shows that no n^k-query algorithm can distinguish $\mathbf{p}$ and $\tilde{\mathbf{p}}$, whence no such algorithm can output a good multiplicative approximation of the entropy for arbitrary input distributions.

4.2. General sub-logarithmic lower bounds

Leaning on the lower bound in [Bel19] for the promise problem of distinguishing two probability distributions, we also get a weak lower bound on the query complexity of entropy estimation for any input model, and in particular, for the vanilla quantum samples model.

Consider the distributions $\mathbf{p} = (1-\epsilon, \frac{\epsilon}{n-1}, \ldots, \frac{\epsilon}{n-1})$ and $\tilde{\mathbf{p}} = (1-\epsilon, \epsilon, 0, \ldots, 0)$ with

$$\begin{aligned} H(\mathbf{p}) &= \Omega\left(\epsilon \log n\right) \\ H(\tilde{\mathbf{p}}) &= h(\epsilon) \leq \log 2, \end{aligned} \tag{4.7}$$

so that the ratio of entropies is

$$\frac{H(\mathbf{p})}{H(\tilde{\mathbf{p}})} \geq 1 + \frac{\epsilon \log(n-1)}{\log 2} = \Omega(\epsilon \log n). \tag{4.8}$$

The Hellinger distance between these two distributions is given by

$$\begin{aligned} d_H(\mathbf{p},\tilde{\mathbf{p}}) &= \sqrt{\frac{1}{2}\left(\sqrt{\frac{\epsilon}{n-1}} - \sqrt{\epsilon}\right)^2 + \sum_{i=3}^{n} \frac{\epsilon}{n-1}} \\ &= \sqrt{\epsilon\left(1 - \frac{1}{\sqrt{n-1}}\right)} \\ &\leq \sqrt{\epsilon}, \end{aligned} \tag{4.9}$$

so that the inverse of the Hellinger distance is of order $\Omega(1/\sqrt{\epsilon})$. If we now make the choice

$$\epsilon = \frac{\gamma^2}{\log n}, \tag{4.10}$$

Belovs' query lower bound for distinguishing $\mathbf{p}$ and $\tilde{\mathbf{p}}$ translates into a

$$\Omega\left(\frac{\sqrt{\log n}}{\gamma}\right) \tag{4.11}$$

lower bound for γ-estimating $H(\mathbf{p})$.

4.3. Polynomial lower bounds in the frequency vector and purified access models

For $\gamma > 1$, consider the domain $[N]$ of size $N = n \cdot n^{1/\gamma^2}$, and consider the uniform distribution $\mathbf{p}$ on $[N]$, and a family of uniform distributions on subsets $S \subset [N]$ of size $|S| = n^{1/\gamma^2}$.

For any such distribution $\tilde{\mathbf{p}}$, an input vector of length N in the frequency vector input model represents an r-to-1 function $f : [N] \to S$, where $r = N/|S| = n$. On the other hand for inputs of this length $\mathbf{p}$ corresponds to a 1-to-1 function since each label in $[N]$ must occur exactly once in the input string.

Note then that the ratio of Shannon entropies is

$$\frac{H(\mathbf{p})}{H(\tilde{\mathbf{p}})} = \gamma^2 + 1 > \gamma^2, \tag{4.12}$$

so that estimating H to multiplicative precision γ will enable us to distinguish the two distributions, and by extension, the two corresponding functions in the frequency vector input model. [AS04] show that distinguishing a 1-to-1 function from an r-to-1 functions requires $\Omega\left((N/r)^{1/3}\right)$ queries to the input function oracle, where N is the size of the domain of the functions. This for us translates to a lower bound of

$$\Omega\left(\left(\frac{N}{r}\right)^{1/3}\right) = \Omega\left(n^{1/3\gamma^2}\right).$$

Recalling from Section 2.2 that any algorithm in the purified query access model implies an algorithm with the same complexity in the frequency vector model, and the fact that classical distributions are automatically examples of density matrices that are diagonal in the computational basis, we see that this lower bound applies to the estimation of both Shannon and von Neumann entropies, and to Models (i), (ii), and (iv).

5. Conclusions and outlook

In this paper, we initiated the investigation of quantum algorithms of sublinear query complexity for the task of γ-multiplicative approximation of both Shannon and von Neumann entropies. Our algorithm for probability distributions achieves a quadratic quantum speedup over classical algorithms, whilst our algorithm for mixed states indicates that it may be possible to estimate other global properties of quantum states with sublinear query complexity in their dimension.

Our results throw some light on the interesting question of the relation between the four input models discussed in Section 2.2, which was first raised by [Bel19]. In particular, the sub-logarithmic lower bound we obtain for the quantum samples model shows that there are problems that cannot be solved in this model with complexity independent

of the dimension, where to our knowledge no such non-trivial problems were previously known for this model. It still remains open whether or not in the quantum samples model (which is strictly stronger than the general purified access models that we study), stronger speedups are possible for entropy estimation and similar tasks.

An immediate question left open by our work is to tighten the lower bounds we obtain, both for Shannon and von Neumann entropies. For the latter, the quantum polynomial method [BKT18] applied to the frequency vector model might yield better bounds. On the other hand, the strong intuition that quantum algorithms typically achieve quadratic speedups indicates that the upper bounds we obtain are tight up to polylogarithmic factors. A potential way to improve on these polylogarithmic factors may be to refine the approximation of $\log x$ by constructing functions such as $x^{2a} + x^a - x^{-a} - x^{-2a}$, reminiscent of symmetric Laurent series.

Our methods can also be extended to other information quantities such as Renyi and Tsallis entropies, and Kullback-Leibler and other divergence measures. Tight bounds on the complexity of multiplicative approximation of these quantities for both probability distributions and mixed states appear within reach, and we hope to report results in this direction in future work.

We close by remarking that this line of work has close and interesting connections to distributional property testing, a rich and active field in classical complexity theory, offering exciting avenues for investigation in quantum complexity theory.

Acknowledgements

We thank Tugkan Batu and Clément Canonne for insightful discussions.

References

[AR20] Scott Aaronson and Patrick Rall. "Quantum approximate counting, simplified", pages 24–32. Society for Industrial and Applied Mathematics, 2020 (page 38).

[AS04] Scott Aaronson and Yaoyun Shi. "Quantum lower bounds for the collision and the element distinctness problems". *J. ACM*, 51(4):595–605, July 2004 (pages 6, 28).

[AIS+19] Jayadev Acharya, Ibrahim Issa, Nirmal V. Shende, and Aaron B. Wagner. "Measuring Quantum Entropy". *IEEE International Symposium on Information Theory - Proceedings*, 2019-July:3012–3016, 2019. arXiv: 1711.00814 (pages 7, 8, 11).

[AK20] Koji Azuma and Go Kato. *Second law of black hole thermodynamics*, 2020. eprint: arXiv:2001.02897 (page 3).

[AS18] Koji Azuma and Sathyawageeswar Subramanian. *Do black holes store negative entropy?*, 2018. eprint: arXiv:1807.06753 (page 3).

[BDK+02] Tuğkan Batu, Sanjoy Dasgupta, Ravi Kumar, and Ronitt Rubinfeld. "The complexity of approximating the entropy". *Proceedings of the Annual IEEE Conference on Computational Complexity*:17, 2002 (pages 3, 4, 7, 13, 14, 25, 26).

[Bek73] Jacob D. Bekenstein. "Black holes and entropy". *Physical Review D*, 7(8):2333–2346, 1973 (page 3).

[Bel19] Aleksandrs Belovs. "Quantum algorithms for classical probability distributions". *Leibniz International Proceedings in Informatics, LIPIcs*, 144:1–14, 2019. arXiv: 1904.02192 (pages 5, 6, 10, 27, 28).

[BHM+02] Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. "Quantum amplitude amplification and estimation". *Quantum Computation and Information, Contemporary Mathematics*, 305:53–74, 2002. arXiv: 0005055 [quant-ph] (pages 12, 38).

[BHH11] Sergey Bravyi, Aram W. Harrow, and Avinatan Hassidim. "Quantum algorithms for testing properties of distributions". *IEEE Transactions on Information Theory*, 57(6):3971–3981, 2011. arXiv: arXiv:0907.3920v1 (page 5).

[BKT18] Mark Bun, Robin Kothari, and Justin Thaler. "The polynomial method strikes back: Tight quantum query bounds via dual polynomials". *Proceedings of the Annual ACM Symposium on Theory of Computing*, (Focs):722–734, 2018. arXiv: 1710.09079 (pages 7, 29).

[CGJ19] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. "The power of block-encoded matrix powers: Improved regression techniques via faster Hamiltonian simulation". *Leibniz International Proceedings in Informatics, LIPIcs*, 132, 2019. arXiv: 1804.01973 (pages 12, 15, 17, 19, 35, 38).

[CFM+10] Sourav Chakraborty, Eldar Fischer, Arie Matsliah, and Ronald de Wolf. "New Results on Quantum Property Testing". Kamal Lodaya and Meena Mahajan, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2010)*, volume 8 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 145–156, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2010 (page 5).

[Don16] Xi Dong. "The gravity dual of rényi entropy". *Nature Communications*, 7(1), 2016 (page 3).

[GL20] András Gilyén and Tongyang Li. "Distributional property testing in a quantum world". *Leibniz International Proceedings in Informatics, LIPIcs*, 151:1–18, February 2020. arXiv: 1902.00814 (pages 5–8, 13, 17, 33).

[GSL+19] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. "Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics". *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, pages 193–204, Phoenix, AZ, USA, 2019. eprint: 1806.01838 (pages 12, 13, 15, 19, 22, 33, 35, 37, 38).

[HGK+10] Matthew B. Hastings, Iván González, Ann B. Kallin, and Roger G. Melko. "Measuring renyi entanglement entropy in quantum Monte Carlo simulations". *Physical Review Letters*, 104(15):157201, 2010 (page 8).

[HOW06] Michał Horodecki, Jonathan Oppenheim, and Andreas Winter. "Quantum state merging and negative information". *Communications in Mathematical Physics*, 269(1):107–136, October 2006 (page 3).

[IMP+15] Rajibul Islam, Ruichao Ma, Philipp M. Preiss, M. Eric Tai, Alexander Lukin, Matthew Rispoli, and Markus Greiner. "Measuring entanglement entropy in a quantum many-body system". *Nature*, 528(7580):77–83, 2015 (page 8).

[JVH+15] Jiantao Jiao, Kartik Venkat, Yanjun Han, and Tsachy Weissman. "Minimax estimation of functionals of discrete distributions". *IEEE Transactions on Information Theory*, 61(5):2835–2885, 2015 (pages 3, 7).

[KP20] Iordanis Kerenidis and Anupam Prakash. "Quantum gradient descent for linear systems and least squares". *Physical Review A*, 101(2):1–26, 2020. arXiv: 1704.04992 (pages 12, 15, 38).

[KRS09] Robert Konig, Renato Renner, and Christian Schaffner. "The operational meaning of min- and max-entropy". *IEEE Transactions on Information Theory*, 55(9):4337–4347, 2009 (page 3).

[Laf16] Nicolas Laflorencie. "Quantum entanglement in condensed matter systems". *Physics Reports*, 646:1–59, 2016 (page 3).

[LW19] Tongyang Li and Xiaodi Wu. "Quantum Query Complexity of Entropy Estimation". *IEEE Transactions on Information Theory*, 65(5):2899–2921, 2019 (pages 5, 7, 8).

[LC17] Guang Hao Low and Isaac L. Chuang. "Optimal Hamiltonian Simulation by Quantum Signal Processing". *Physical Review Letters*, 118(1):010501, January 2017. arXiv: 1610.06546 (page 38).

[MW16] Ashley Montanaro and Ronald de Wolf. "A survey of quantum property testing". *Theory of Computing*, 2016(Graduate Surveys 7):1–81, 2016. arXiv: 1310.2035 (page 5).

[MDS+13] Martin Müller-Lennert, Frédéric Dupuis, Oleg Szehr, Serge Fehr, and Marco Tomamichel. "On quantum rényi entropies: a new generalization and some properties". *Journal of Mathematical Physics*, 54(12):122203, December 2013 (page 9).

[Neu27] J. von Neumann. "Thermodynamik quantenmechanischer gesamtheiten". ger. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch Physikalische Klasse*, 1927:273–291, 1927 (page 3).

[NC10] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2010 (page 9).

[OW15] Ryan O'Donnell and John Wright. "Quantum spectrum testing". *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '15, pages 529–538, Portland, Oregon, USA. Association for Computing Machinery, 2015 (pages 5, 11).

[Pet01] D. Petz. "Entropy, von neumann and the von neumann entropy", 2001. eprint: arXiv:math-ph/0102013 (page 3).

[Sch95] Benjamin Schumacher. "Quantum coding". *Phys. Rev. A*, 51:2738–2747, 4, 1995 (page 3).

[Sha48] C. E. Shannon. "A mathematical theory of communication". *The Bell System Technical Journal*, 27(3):379–423, 1948 (pages 3, 9).

[SBJ19] Sathyawageeswar Subramanian, Stephen Brierley, and Richard Jozsa. "Implementing smooth functions of a hermitian matrix on a quantum computer". *Journal of Physics Communications*, 3(6):065002, 2019 (page 19).

[SH21] Sathyawageeswar Subramanian and Min-Hsiu Hsieh. "Quantum algorithm for estimating renyi entropies of quantum states". *Phys. Rev. A*, 104:022428, 2, August 2021 (page 8).

[SUR+20] Yohichi Suzuki, Shumpei Uno, Rudy Raymond, Tomoki Tanaka, Tamiya Onodera, and Naoki Yamamoto. "Amplitude estimation without phase estimation". *Quantum Information Processing*, 19(2), 2020 (page 38).

[VV11] G. Valiant and P. Valiant. "The power of linear estimators". *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 403–412, 2011 (page 3).

[Val11] Paul Valiant. "Testing symmetric properties of distributions". *SIAM Journal on Computing*, 40(6):1927–1968, 2011 (pages 4, 7, 25).

[Wol19] Ronald de Wolf. *Quantum computing: lecture notes*. 2019 (page 9).

[WY16] Yihong Wu and Pengkun Yang. "Minimax rates of entropy estimation on large alphabets via best polynomial approximation". *IEEE Trans. Inf. Theor.*, 62(6):3702–3720, June 2016 (pages 3, 7).

[ZLO+07] Haiquan (Chuck) Zhao, Ashwin Lall, Mitsunori Ogihara, Oliver Spatscheck, Jia Wang, and Jun Xu. "A data streaming algorithm for estimating entropies of od flows". *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, IMC '07, pages 279–290, San Diego, California, USA. Association for Computing Machinery, 2007 (pages 8, 17).

A. Creating block encodings or projected unitary encodings from the purified access oracle

We give an overview of the methods used by [GL20] to obtain projected unitary encodings from purified access oracles in this appendix. Recall that following [GSL+19, Definition 43], we defined an (α, a, ϵ) projected unitary encoding of an operator A acting on s qubits is a unitary U acting on $a+s$ qubits, such that

$$\|A - \alpha \Pi^\dagger U \tilde{\Pi}\| \leq \epsilon, \tag{A.1}$$

where the first register consists of ancillary qubits, Π and $\tilde{\Pi}$ represent projections, i.e. $\Pi := |0\rangle^{\otimes a} \otimes \mathbb{1}_s$ is an isometry mapping $(\mathbb{C}^2)^{\otimes s} \mapsto \text{span}_{\mathbb{C}}\{|0\rangle^{\otimes a}\} \otimes (\mathbb{C}^2)^{\otimes s}$, and $\alpha, \epsilon \in (0, \infty)$. Below, we recall how to obtain such block encodings from purified access oracles to classical probability distributions and mixed states.

A.1. Classical distributions

In the case of a classical input distribution, the purified access oracle in Eq. (2.10) can be turned into a block encoding for a matrix with singular values equal to the $\sqrt{p_j}$ as follows.

We choose $\Pi := \sum_{i \in [n]} \mathbb{1} \otimes |i\rangle\langle i| \otimes |i\rangle\langle i|$, and $\tilde{\Pi} := |0\rangle\langle 0| \otimes |0\rangle\langle 0| \otimes \mathbb{1}$, where each of the three registers is of dimension n. With $W = U \otimes \mathbb{1}$, we have that

$$P = \Pi U \tilde{\Pi} = \sum_{i \in [n]} \sqrt{p_i} \, |\phi_i i i\rangle\langle 00 i| \, . \tag{A.2}$$

The right hand side above represents the singular value decomposition (SVD) of a matrix P with singular values $\sigma_i = \sqrt{p_i}$, left singular vectors $|\phi_i i i\rangle$ and right singular vectors $|00i\rangle$, i.e.

$$\begin{aligned} P \, |00i\rangle &= \sqrt{p_i} \, |\phi_i i i\rangle \\ P^\dagger \, |\phi_i i i\rangle &= \sqrt{p_i} \, |00i\rangle \, . \end{aligned}$$

Hence we see that U furnishes a $(1, \lceil \log n \rceil, 0)$ block encoding of P.

A.2. Arbitrary quantum density matrices

Classical probability distributions correspond to the special case when ρ is diagonal in the computational basis. For arbitrary density matrices, it is a little bit harder to create a projected unitary encoding that has the square roots of the eigenvalues of ρ, $\sqrt{p_i}$, as the singular values. Instead, [GL20] give a construction which has singular values $\sqrt{\frac{p_i}{n}}$.

Since we do not know the eigenbasis of ρ beforehand, we define the projection operators $\Pi = \mathbb{1} \otimes |0\rangle\langle 0| \otimes |0\rangle\langle 0|$ and $\tilde{\Pi} = |0\rangle\langle 0| \otimes |0\rangle\langle 0| \otimes \mathbb{1}$. As before, the third register contains

the ancilla, and the first two contain states of the system. We also need a unitary map W that prepares the maximally entangled state on the two copies of the system register:

$$|0\rangle\,|0\rangle \longmapsto \frac{1}{\sqrt{n}}\sum_{j=1}^{n}|j\rangle\,|j\rangle\,.$$

With these operators and the purified access oracle U_ρ of Eq. (2.9), we can define the unitary

$$U' = \left(\mathbb{1}\otimes U_\rho^\dagger\right)(W\otimes\mathbb{1})\,,$$

which gives rise to the following projected unitary encoding

$$\Pi U'\tilde{\Pi} = \frac{1}{\sqrt{n}}\sum_{j=1}^{n}\left|\phi_j'\right\rangle\langle 0|\otimes|0\rangle\langle 0|\otimes|0\rangle\langle\psi_j|\,,$$

where $\{\left|\phi_j'\right\rangle\}$ is a Schmidt basis for the first half of the bipartite maximally entangled state of dimension n.

This encoding has a leading normalisation factor of $\frac{1}{\sqrt{n}}$, which directly contributes a factor of $\sqrt{n}$ to the complexity of entropy estimation for quantum density matrices.

B. Implementing power functions of block encoded matrices

Polynomial functions of a matrix are defined through its singular value decomposition. An $n\times n$ matrix P has n real singular values σ_j, with a singular value decomposition in terms of its left and right singular vectors $|v_j\rangle$ and $|w_j\rangle$. An even or odd polynomial function f of such a matrix is then defined as having the same singular vectors, but with the eigenvalues $f(\sigma_j)$, as follows

$$\begin{aligned} P &= \sum_{j=1}^{n}\lambda_j\,|v_j\rangle\,\langle w_j|\,; \\ f(P) &= \begin{cases}\sum_{j=1}^{n} f(\sigma_j)\,|w_j\rangle\,\langle w_j| & \text{when } f(x)=f(-x)\\ \sum_{j=1}^{n} f(\sigma_j)\,|v_j\rangle\,\langle w_j| & \text{when } f(x)=-f(-x)\end{cases} \end{aligned} \tag{B.1}$$

The key trick in using QAE to estimate a functional $\varphi(\mathbf{p}) := \sum_i f(p_i)$ of an input vector $\mathbf{p}$ (in our case, a probability mass function) is to use the input purified access unitary that performs the map

$$U\left|0^d\right\rangle\left|0^d\right\rangle = |\psi_{\mathbf{p}}\rangle = \sum_{i=1}^{n}\sqrt{p_i}\,|\phi_i\rangle\,|\psi_i\rangle$$

where $d = \lceil \log n \rceil$, to construct a new unitary circuit which on a chosen, easy to prepare initial state performs a map of the type

$$W \left|\psi_\rho\right\rangle \left|0\right\rangle_{\text{flag}} = \sum_{i=1}^{n} \sqrt{f(p_i)} \left|\phi_i\right\rangle \left|\psi_i\right\rangle \left|0\right\rangle_{\text{flag}} + \left|...\right\rangle \left|1\right\rangle_{\text{flag}},$$

so that value of the target functional φ is encoded in the amplitude of the part of the output state marked by the $|0\rangle$ subspace of the flag register.

A projected unitary encoding U of a matrix P can be used to implement such smooth functions of the input matrix via polynomial approximations, with the following theorem.

Theorem 3 (Theorem 56, [GSL+19]). *Given an (α, a, ϵ) block encoding U of a Hermitian matrix P, for any degree m polynomial $f(x)$ that satisfies $\forall\ x \in [-1,1],\ |f(x)| < 1/2$, there exists a $(1, a+2, 4m\sqrt{\epsilon/\alpha} + \delta)$ block encoding U_f of $f(P/\alpha)$. We can construct U_p using m applications of U and $U^\dagger$, a single application of controlled-U, and $\mathcal{O}((a+1)m)$ additional 1- and 2-qubit gates. A description of the circuit of U_f can be calculated in $\mathcal{O}(\text{poly}(m, \log 1/\delta))$ time on a classical computer.*

Using Theorem 3, we can implement ϵ-approximate block encodings of power functions P^c on the part of the singular value spectrum of P that is contained in $[\delta, 1]$ for $\delta > 0$ by using polynomial approximations. The lower cutoff δ is necessary because power functions for non-integer exponents $c \in \mathbb{R}$ are not differentiable at $x = 0$. On the other hand, monomials for $c = 1, 2, \ldots$ can be implemented exactly on the entire domain $[0, 1]$.

We first note the following way [CGJ19; GSL+19] of obtaining polynomial approximations of any desired degree for positive and negative power functions over a domain $[x_0 - r - \delta, x_0 + r + \delta]$ of radius $r \in (0, 2]$ centred around a point $x_0 \in [-1, 1]$, with some wiggle room for the polynomial to vary, specified by the parameter $\delta \in (0, r]$.

Positive Power functions: Consider $f(x) = x^c$ for $c > 0$. The Taylor series expansion of f around $x_0 = 1$

$$\begin{aligned} f(1+x) &= (1+x)^c \\ &= 1 + \sum_{k=1}^{\infty} \binom{c}{k} x^k \end{aligned} \tag{B.2}$$

converges $\forall x \in [-1, 1]$, where

$$\binom{c}{k} := \frac{c(c-1)(c-2)\ldots(c-k+1)}{k!}.$$

Notice that we have

$$\begin{aligned}
1+\sum_{k=1}^{\infty}\left|\binom{c}{k}\right|(1-\delta+\delta)^k &= 1+\sum_{k=1}^{\infty}\left|\binom{c}{k}\right| \\
&= 1-\sum_{k=1}^{\infty}\binom{c}{k}(-1)^k \\
&= 2-\sum_{k=0}^{\infty}\binom{c}{k}(-1)^k \\
&= 2-f(1-1) \\
&= 2.
\end{aligned}$$

Specific to our purpose, this means we can choose $x_0 = 1$, $r = 1-\sqrt{\beta}/2$, $\delta = \sqrt{\beta}/2$, and the normalisation factor $B = 2$ for implementing positive power functions of block encodings of P with singular values $\mathbf{p} = (\sqrt{p_1}, \ldots, \sqrt{p_n})$ on the domain $[\sqrt{\beta}, 1]$ corresponding to the heavy elements $i \in B_\beta$ with probability masses $p_i \geq \beta$.

Negative Power functions: Consider $f(x) = x^{-c}$ for $c > 0$. The Taylor series expansion of f around $x_0 = 1$

$$\begin{aligned}
f(1+x) &= (1+x)^{-c} \\
&= 1+\sum_{k=1}^{\infty}\binom{-c}{k}x^k
\end{aligned} \tag{B.3}$$

converges $\forall x \in [-1, 1]$, where

$$\binom{-c}{k} := \frac{-c(-c-1)(-c-2)\ldots(-c-k+1)}{k!}.$$

With $\delta' := \frac{\delta}{2\max(1,c)}$ notice that we have

$$\begin{aligned}
1+\sum_{k=1}^{\infty}\left|\binom{-c}{k}\right|(r+\delta')^k &= 1+\sum_{k=1}^{\infty}\binom{-c}{k}(-r-\delta')^k \\
&= (1-r-\delta')^{-c} \\
&= (\delta-\delta')^{-c} \\
&= \delta^{-c}(1-\frac{\delta'}{\delta})^{-c} \\
&= \delta^{-c}(1-\frac{1}{2\max(1,c)})^{-c} \\
&= 2\delta^{-c}.
\end{aligned}$$

If we choose to normalise the original function to $\frac{\delta^c}{2}x^{-c}$, the above calculation shows that we can choose $x_0 = 1$, $r = 1-\sqrt{\beta}/2$, $\delta = \sqrt{\beta}/2$, and the normalisation factor $B = 1$

for implementing negative power functions over $[\sqrt{\beta}, 1]$. Since the case of negative power functions is more illustrative than positive ones, we state this formally below. A similar statement holds for the positive case.

Lemma 3 (Corollary 67, [GSL+19]). *Given a $(\alpha, a, 0)$ unitary block encoding U of a matrix with singular value decomposition $P = \sum_i \sigma_i |v\rangle\langle w|$, and an even polynomial $f_c(x)$ on $[-1, 1]$ that approximates the negative power function x^{-c} for $c > 0$ such that*

$$\begin{aligned} \left| f_c(x) - \frac{\delta^c}{2} x^{-c} \right| &\leq \epsilon && \forall x \in [\delta, 1] \\ |f_c(x)| &\leq 1 && \forall x \in [-1, 1] \\ m = \deg f &= \mathcal{O}\left(\frac{\max(1, c)}{\delta} \log \frac{1}{\epsilon} \right), \end{aligned}$$

we can implement a $(2/\delta^c, a + 2, \epsilon)$ block encoding U_f of the matrix polynomial $f_c(P) = \sum_i f_c(\sigma_i) |w\rangle\langle w|$ using m applications of U and $U^\dagger$, a single application of controlled-U, and $\mathcal{O}(ma)$ additional one- and two-qubit gates. Furthermore, a description of the quantum circuit U_f can be computed classically in time $\mathcal{O}(\mathrm{poly}(m, \log 1/\epsilon))$.

C. Quantum phase estimation and singular value estimation

We would like to use Quantum Phase Estimation as a subroutine to separately flag the subspaces of heavy and light elements. In essence we want to perform the map in Eq. (3.3), i.e.

$$\sum_{i \in [n]} \sqrt{p_i} |\phi_i\rangle |i\rangle \otimes |0^m\rangle \mapsto \sum_{i \in [n]} \sqrt{p_i} |\phi_i\rangle |i\rangle |q_i\rangle , \tag{C.1}$$

where $\left|\sqrt{p_i} - q_i\right| \leq 2^{-(m+1)} =: \epsilon$, and m is the number of bits of precision.

Recall that we have a block encoding U of a matrix P that represents our input distribution, where P has the singular value decomposition

$$P = \tilde{\Pi} U \Pi = \sum_{i \in [n]} \sqrt{p_i} |\phi_i i i\rangle\langle 00 i| . \tag{C.2}$$

Functions of P defined by even or odd polynomials f or $\tilde{f}$ respectively acting on the singular values then have the form

$$\begin{aligned} f(P) &:= \sum_{i \in [n]} f(\sqrt{p_i}) |00i\rangle\langle 00i| \\ f(P) &:= \sum_{i \in [n]} f(\sqrt{p_i}) |\phi_i i i\rangle\langle 00 i| . \end{aligned} \tag{C.3}$$

In principle, we can simply use the standard textbook version of the quantum phase estimation algorithm (QPE) which requires controlled-U operators and the quantum

fourier transform (QFT) in order to estimate the phases $\theta_j \in [0,1)$ of the eigenvalues $\lambda_j = \mathrm{e}^{2\pi \mathrm{i}\theta_j}$ of U. We consider $e^{2\pi \mathrm{i}Pt}$ as the input unitary, which can be implemented using the block encoding of P via Hamiltonian simulation, with query complexity to U and $U^\dagger$ bounded by

$$\mathcal{O}\left(t + \frac{\log 1/\epsilon}{\log\log 1/\epsilon}\right)$$

where ϵ is a precision parameter defined by $\forall j \in [n]$, $\left|\mathrm{e}^{2\pi \mathrm{i}\sqrt{p_j}} - \lambda_j\right| \leq \epsilon$ [LC17; GSL+19]. This condition translates to $\left|\theta_j - \sqrt{p_j}\right| \leq \frac{1}{\pi} \arcsin \frac{\sqrt{\epsilon}}{2} \leq \sqrt{\epsilon}$. This error is benign as far as we are concerned: we can choose it to be of order $1/n$ or even $1/n^2$ while incurring only a additive logarithmic overhead in the complexity. Indeed, we shall choose it to be inverse polynomial in n, so that the subsequent step that uses QPE to estimate θ_j still behaves as we expect it to — it will produce an estimate of zero whenever $\theta_j^2 \leq \beta \iff p_j < \beta$.

While the above explanation is the high level intuition, in practice things can be a bit more delicate, and we use technique of quantum singular value estimation (QSVE). Since P is not Hermitian, we consider a symmetrised version of it defined by $\hat{P} = |0\rangle\langle 1| \otimes P + |1\rangle\langle 0| \otimes P^\dagger$, which has eigenvectors $|0\rangle \otimes |\phi_i i\rangle + |1\rangle \otimes |00i\rangle$ and eigenvalues $\sqrt{p_i}$. The problem then is to perform the map in Eq. (3.3) using the block encoding of $\hat{P}$, which in turn can easily be constructed using the block encoding of P. This matches the problem addressed in [KP20; CGJ19; GSL+19], and the complexity is essentially $\widetilde{O}(1/\epsilon)$ where ϵ is the precision to which we would like to estimate the singular values.

We can then choose $m = \log \sqrt{1/\beta}$, and the query complexity of the QSVE subroutine becomes

$$\mathcal{O}\left(\frac{1}{\epsilon}\right) = \mathcal{O}\left(\frac{1}{\sqrt{\beta}}\right).$$

D. Quantum amplitude estimation

Quantum amplitude estimation (QAE) is a technique wherein quantum phase estimation (QPE) is used to estimate the amplitude of a certain basis state (more generally, of any state about which we can perform a reflection operation) in a superposition produced by applying a unitary operation U to a given input state. The QPE algorithm is applied to estimate the eigenvalues of the Grover iterate constructed from the input unitary. We also note that the most modern methods of QAE do not rely on QPE or the quantum fourier transform in an essential way [SUR+20; AR20].

Theorem 4 ([BHM+02], Theorem 12)**.** *Given a unitary U with the action*

$$U|0\rangle|0\rangle = \sqrt{p}\,|0\rangle|\phi\rangle + |\perp\rangle,$$

where $|\phi\rangle$ *is a normalised state on the system register, and* $(|0\rangle\langle 0| \otimes \mathbb{1})\,|\perp\rangle = 0$*, the quantum amplitude estimation algorithm outputs* $\tilde{p} \in [0,1]$ *satisfying*

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

with a success probability at least $8/\pi^2$*, making* M *uses of* U *and* $U^\dagger$*.*

To get an approximation $\tilde{p}$ that is correct to a constant additive precision $\epsilon \in (0, 1/2)$, we can choose $M = \lceil 2\pi \left(\frac{2\sqrt{p}}{\epsilon} + \frac{1}{\sqrt{\epsilon}} \right) \rceil = \Theta\left(\frac{\sqrt{p}}{\epsilon} + \frac{1}{\sqrt{\epsilon}} \right)$, and hence with a complexity of $\Theta(1/\epsilon)$ we can estimate p to additive precision ϵ.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il