

Quantum Computational Complexity of Matrix Functions

Santiago Cifuentes¹, Samson Wang^{2,*}, Thais L. Silva³, Mario Berta^{4,5} and Leandro Aolita³

¹*Instituto de Ciencias de la Computación, UBA-CONICET, Argentina*

²*Institute for Quantum Information and Matter, Caltech, USA*

³*Quantum Research Center, Technology Innovation Institute, Abu Dhabi, United Arab Emirates*

⁴*Institute for Quantum Information, RWTH Aachen University, Germany*

⁵*Department of Computing, Imperial College London, United Kingdom*



(Received 8 September 2025; revised 25 November 2025; accepted 2 March 2026; published 18 June 2026)

We investigate the dividing line between classical and quantum computational power in estimating properties of matrix functions. More precisely, we study the computational complexity of two primitive problems: given a function f and a Hermitian matrix A , we compute a matrix element of $f(A)$ or a local measurement on $f(A)|0\rangle^{\otimes n}$, with $|0\rangle^{\otimes n}$ being an n -qubit reference state vector, in both cases up to additive approximation error. We consider four functions—monomials, Chebyshev polynomials, the time evolution function, and the inverse function—and probe the complexity across a broad landscape covering different problem input regimes. Namely, we consider two types of matrix inputs (sparse and Pauli access), matrix properties (norm and sparsity), the approximation error, and function-specific parameters. We identify BQP-COMPLETE forms of both problems for each function and then toggle the problem parameters to easier regimes to see where hardness remains, or where the problem becomes classically easy. As part of our results, we make concrete a hierarchy of hardness across the functions; in parameter regimes where we have classically efficient algorithms for monomials, all three other functions remain robustly BQP-HARD, or hard under usual computational complexity assumptions. In identifying classically easy regimes, among others, we show that for any polynomial of degree $\text{poly}(n)$, both problems can be efficiently classically simulated when A has $\mathcal{O}(\log n)$ nonzero coefficients in the Pauli basis. This contrasts with the fact that the problems are BQP-COMPLETE in the sparse access model even for constant row sparsity, whereas the stated Pauli access efficiently constructs sparse access with row sparsity $\mathcal{O}(\log n)$. Our work provides a catalog of efficient quantum and classical algorithms for fundamental linear-algebra tasks.

DOI: [10.1103/g5x4-jcsz](https://doi.org/10.1103/g5x4-jcsz)

I. INTRODUCTION

A. Setting and motivation

In which problems from matrix algebra can we expect strong (i.e., super-polynomial) quantum speedups? This question lies at the heart of understanding the computational complexity of matrix functions and has motivated previous results proving quantum advantage in specific settings [1–3]. Moreover, the emergence of the quantum singular value transformation (QSVT) [4,5] and related techniques [6–9] has brought new depth to this

investigation. These are powerful abstract frameworks for approximating matrix functions via matrix polynomials, allowing for the implementation of any smooth-enough matrix function with polylogarithmic runtime in the dimension of the matrix. This provides a unified perspective on synthesizing many quantum algorithms, as matrix polynomials can be thought of as building blocks for constructing other interesting matrix transformations on a quantum computer. However, not all efficient quantum algorithms for matrix polynomials lead to a super-polynomial quantum speedup. Indeed, some have classically efficient counterparts, and dequantization results have been proven for QSVT under certain assumptions and settings [10,11]. In this work, we aim to characterize this dividing line in terms of computational complexity, identifying when the use of quantum algorithms for matrix functions can genuinely yield a strong quantum speedup. Apart from matrix polynomials, we also discuss the hardness of two more concrete problems: matrix inversion and

*Contact author: samsonwang@outlook.com

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

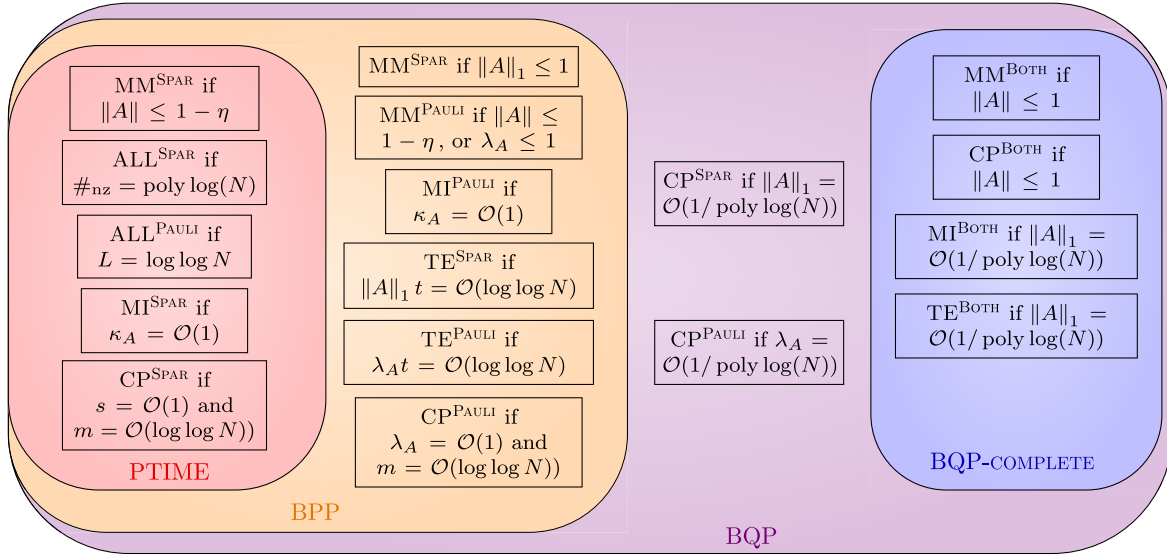


FIG. 1. Diagram indicating the complexity of the studied problems. We use the acronyms MM (matrix monomial), CP (Chebyshev polynomials), MI (matrix inversion), TE (time evolution), and ALL (all the previous functions). The superscript indicates the access model: SPAR (Sparse), PAULI (Pauli), or BOTH (the problem belongs to the complexity class for both access models). We denote $\|A\|$ as the operator (or spectral) norm of A , $\|A\|_1$ as its induced 1-norm, λ_A as the vector ℓ_1 -norm of its Pauli coefficients, L as the number of Pauli terms, $\#_{\text{nz}}$ as its number of nonzero elements in the computational basis, κ_A as its condition number, t as the evolution time, and $\eta > 0$ is an $\mathcal{O}(1)$ number. Although the *local measurement problem* appears, at first sight, a more natural task for quantum algorithms than the *matrix element problem*, interestingly, we find an almost-complete match between the two problems for almost all settings studied (see Table I). The only potentially discrepant cases are CP for inverse polynomially small matrix norms (purple region) and TE for constant time (blue region), for which our hardness proof works only for the matrix element problem. All other results sketched in the figure hold for both problems. When not indicated, it is assumed that $\|A\| \leq 1$ and all other problem parameters (sparsity, inverse precision, and problem-specific parameters) scale polynomially in the input size (which is polylogarithmic in the matrix dimension). Hence, for instance, MM^{SPAR} indicates both the matrix element and local measurement problems for $f_m(A) = A^m$, for m polynomial in the input size, where A is given through the sparse access model.

time evolution, which themselves can also be considered building blocks for synthesizing other matrix functions [8, 12–18].

We study two primitive problems for matrix algebra. First, given a Hermitian matrix A and function f , we ask for one matrix element $[f(A)]_{ij}$. We refer to this as the *matrix element problem*, which can be considered as an elemental matrix algebra task. Second, we consider a task, which we refer to as the *local measurement problem*, that at first sight may appear more native to quantum approaches than the previous one: Performing a local measurement on a state acted on by $f(A)$ without normalization. That is, we ask for the value $\langle 0|^{\otimes n} f(A)^\dagger (|0\rangle\langle 0| \otimes \mathbb{1}_{N/2}) f(A) |0\rangle^{\otimes n}$, where $|0\rangle\langle 0|$ is a projector acting on the first qubit and $\mathbb{1}_{N/2}$ is the identity matrix on the remaining $n - 1$ qubits, for a total system dimension $N = 2^n$. Immediate questions arise: *how do the complexities of these two problems relate to each other? And how do they depend on the input models assumed?* For both these problems, we derive hardness results for several different settings and problem-parameter regimes (see Fig. 1).

The complexity of matrix function problems can depend on various factors, which we tune individually to

investigate their effect on the hardness of the problems. The concrete properties that we study are

- The access model to the matrix.
- The matrix normalization.
- The matrix sparsity.
- The desired precision.
- The type of function.

Let us now briefly discuss and motivate each of these.

Access model. A standard matrix access model that we investigate is the so-called *sparse access model* (Definition 2), where the nonzero matrix entries of A in the computational basis are accessible via an efficiently computable function. This is commonly studied both constructively for quantum algorithms as well as for computational complexity analysis. Without an efficient structure for matrix entries, a generic quantum data structure of size $\Omega(N)$ would be required, even for row- and column-sparse matrices. That is, $\Omega(N)$ gates are required to instantiate the access model for a generic $N \times N$ row- or column-sparse matrix [19]. Sparse access should then be seen as one way of imposing efficient access to large matrices. By

“efficient,” we mean that there is a polylogarithmic-sized circuit in both depth and width able to provide access to the entries.

Another reasonable type of access one might ask for, particularly in physically-motivated problems, is classical access to the nonzero coefficients of A in the Pauli basis (Definition 3). This access model can be considered as a concrete special case of sparse access, and it can only be efficient generically if there are polylogarithmically many nonzero coefficients. This condition is commonly encountered in chemistry, material science, or many-body physics applications. Alternatively, the nonzero Pauli coefficients can often have some underlying structure enabling efficient description, as has been recently considered in a line of work on randomized quantum algorithms [20–23]. The main question we ask concerning these access models is does providing a matrix in Pauli access change the complexity of these problems compared to sparse access in the computational basis? Answering this question should shed light on whether hardness results for matrix algebra problems in one model hold any relevancy for hardness in the other one. For instance, if a problem’s complexity is unchanged by the access model, this suggests a form of complexity robustness with respect to the matrix representation.

Sparsity. To a certain extent, the sparser the matrix, the easier it is to perform matrix algebra—both for deterministic and randomized classical algorithms. Moreover, as discussed above, it is common to ask for efficient sparse access models to allow for scalable quantum algorithms. Thus, it is pertinent to ask whether quantum algorithms become classically simulatable at some sparsity level (both in the sparse and Pauli access models). However, in the absence of further conditions, we cannot expect efficient classical algorithms for general matrix functions even for $\mathcal{O}(1)$ -row-sparse matrices (in the computational basis) without further constraints—for example, the problem of computing a local measurement for $f(A) = A^{-1}$ is well known to be BQP-COMPLETE for $\mathcal{O}(1)$ -sparse matrices [2]. In our work we start with the case of $\mathcal{O}(\text{poly log}(N))$ many nonzero coefficients per row or $\mathcal{O}(\text{poly log}(N))$ nonzero coefficients in the Pauli basis (the maximal amounts that still allow efficient representation of matrices with generic coefficients), and investigate the change in complexity when we tune the sparsity down with additional constraints.

Matrix normalization. For many problems, the complexity of the classical algorithms solving them depends on norms different from the operator norm [24,25], and thus considering different normalization conditions for A can affect such dequantization results. Here, we consider normalization conditions based mostly on three different matrix norms: the operator norm $\|A\|$; the larger induced 1-norm $\|A\|_1$ (for the sparse access model) and the vector ℓ_1 -norm of the Pauli coefficients (for the Pauli access

model). We consider the classes of problems where each of these norms is upper bounded by 1 and also investigate how the complexity changes when stronger bounds are put on these norms.

Error parameter. For each function of interest, we consider two regimes of additive precision for the desired estimations: $1/\varepsilon = \mathcal{O}(\text{poly log}(N))$ and $1/\varepsilon = \mathcal{O}(1)$. Recent work [11] shows that changing from the former to the latter can make certain BQP-COMPLETE problems turn classically efficiently solvable, under access assumptions. Moreover, understanding the impact of the precision parameter in a problem’s complexity is a central question regarding the Quantum PCP Conjecture [26].

Type of function. We study two classes of polynomials. First, we study monomials. This is arguably the simplest instance of a polynomial, and thus potentially the most amenable one to classical approaches. Second, we study Chebyshev polynomials. This is a powerful class of polynomials in numerical approximation theory [27] and has been widely studied in the context of quantum algorithms [4,6,28–30]. Finally, we also consider time evolution, also known as Hamiltonian simulation [i.e., $f_t(A) = e^{-iAt}$], and the inverse function. Moreover, for each function family, we characterize the complexity of subclasses given by restricted regimes of relevant parameters (the polynomial degree, evolution time, or matrix condition number).

The rest of our paper is structured as follows: we discuss related work in Sec. IB, we present a condensed form of our results in Sec. II along with a summary in Table I and Fig. 1, and the remainder of the manuscript is dedicated to elucidating our results in full detail with proofs.

B. Related work

In Ref. [24], Montanaro and Shao study the query complexity (i.e., number of queries to an oracle for A) for the matrix element problem for both classical and quantum algorithms for matrices of bounded operator norm $\|A\| \leq 1$. For classical algorithms, they give query complexity lower bounds that grow exponentially in the degree of the target polynomial, assuming the input matrix has bounded operator norm and the estimation error satisfies $1/\varepsilon = \mathcal{O}(\text{poly log}(N))$ [33]. This contrasts with (essentially matching) upper and lower bounds for quantum algorithms, which are linear in the polynomial degree. Our work can be seen as a complementary investigation where, instead of query complexity, we tackle computational complexity. A central difference between these two notions is that, with computational complexity, the access model must also be computationally efficient. In contrast, with query complexity, computational hardness can (in theory) be hidden inside the oracle to A . Another distinction from Ref. [24] is that there the central parameter probed is the degree of the target polynomial, whereas here

TABLE I. Results for the matrix element and local measurement problems. Unless explicitly stated we assume that $\|A\| \leq 1$ and $1/\varepsilon, s, L, m, t, \kappa_A, \lambda_A = \mathcal{O}(\text{poly log}(N))$ where applicable, which can be considered problem inputs. All results shown hold for both problems, but for the one with a superscript a, our proof for the local measurement problem requires an additional mild assumption (see C.2 for details). All the hardness results with the superscript b hold even for the easier problem where precision ε is fixed. The results with existing references can be obtained in some (partial) form via the cited prior art, and we extend the claims through additional observations. In turn, the results with the symbol $\mathfrak{F}$ are to our knowledge not published anywhere but we believe them to be folklore knowledge in the community. We prove hardness and classical simulability results for four classes of matrix functions: monomials, Chebyshev polynomials $T_m(A)$, the complex exponential (time evolution), and the inverse function. In addition, certain results for classical simulability are generalized for general polynomials (see Summary 3). The symbol $\#_{\text{nz}}$ stands for “number of nonzero elements in the computational basis.” The rest of the notation in the table is the same as in Fig. 1. We remark that the stronger the condition on the norm (smaller norm), generally the easier the problem.

$f(A)$	Access	$\ A\ \leq c$	$\ A\ _1$ or $\lambda_A \leq k$	Additional classically efficient cases	
				Super-sparse matrices	Problem-specific cases
A^m	Sparse	BQP-COMPLETE for $c = 1$ (M.1 & [1])	BPP for $k = 1$ (M.2.3 & [31], M.3.2 & [22])	$\#_{\text{nz}} = \mathcal{O}(\text{poly log}(N))$ (M.4.1, $\mathfrak{F}$)	$s = 1$, or $s = \mathcal{O}(1)$ and $m = \mathcal{O}(\log \log N)$ (M.2.1 & [11]); $\ A\ \leq 1 - \eta$, s or $\varepsilon^{-1} = \mathcal{O}(1)$ (M.2.2)
	Pauli			$L = \mathcal{O}(\log \log N)$ (M.4.2)	$\ A\ \leq 1 - \eta$, $\lambda_A = \mathcal{O}(1)$ (M.3.2)
$T_m(A)$	Sparse	BQP-COMPLETE ^b for $c = 1$ (C.1, $\mathfrak{F}$)	Classically hard ^a for $k = \mathcal{O}(1/\text{poly log}(N))$ if BPP $\neq$ BQP (C.2, $\mathfrak{F}$)	$\#_{\text{nz}} = \mathcal{O}(\text{poly log}(N))$ (C.3.2, $\mathfrak{F}$)	s or $\ A\ _1 = \mathcal{O}(1)$ and $m = \mathcal{O}(\log \log N)$ (C.3.1)
	Pauli			$L = \mathcal{O}(\log \log N)$ (C.3.2)	$m = \mathcal{O}(\log \log N)$ and $\lambda_A = \mathcal{O}(1)$ (C.3.1); $\lambda_A = \mathcal{O}(1/(m^2 \log^{1.5}(N)))$ and $1/\varepsilon = \mathcal{O}(1)$ (C.3.3)
A^{-1}	Sparse	BQP-COMPLETE ^b for $c, k = \mathcal{O}(1/\text{poly log}(N))$ (I.1.1–I.1.2 & [2])		$\#_{\text{nz}} = \mathcal{O}(\text{poly log}(N))$ (P.1, $\mathfrak{F}$)	s or $\ A\ _1 = \mathcal{O}(1)$ and $\kappa_A = \mathcal{O}(1)$ (I.2)
	Pauli			$L = \mathcal{O}(\log \log N)$ (P.2)	$\lambda_A = \mathcal{O}(1)$ and $\kappa_A = \mathcal{O}(1)$ (I.2); $\lambda_A = \tilde{\mathcal{O}}(1/(\kappa_A^2 \log^{1.5}(N)))$ and $1/\varepsilon = \tilde{\mathcal{O}}(1)$ (P.3)
e^{-iAt}	Sparse	BQP-COMPLETE ^b for $c, k = \mathcal{O}(1/\text{poly log}(N))$ (T.1.1–T.1.3 & [3,32])		same as A^{-1}	$\ A\ t = \mathcal{O}(\log \log N)$, $s = \mathcal{O}(1)$ (T.2); $\ A\ _1 t = \mathcal{O}(\log \log N)$ (T.2)
	Pauli				$\lambda_A t = \mathcal{O}(\log \log N)$ (T.2)

we toggle various additional problem parameters including properties of the matrix.

Another related line is the significant body of literature on so-called quantum-inspired or dequantization algorithms [10,11,25,34–37]. Here, classical algorithms are emboldened with a particular kind of access to vectors and matrices which mimics quantum query access, usually called “sample and query access models.” While such work is hugely informative about the feasibility of generic superpolynomial speedup for matrix problems given oracular query access, we stress again, our setting differs in that we work with access models that are efficiently instantiable.

In turn, we highlight a recent work by Gharibian and Le Gall [11], which provides a quantum-inspired algorithm

for the Guided Local Hamiltonian problem [38] for constant precision. This is relevant to our discussions for two reasons. First, the authors establish a core classical subroutine for sparse-matrix monomials which has exponential-in-degree runtime in general but is efficient for 1-sparse matrices or for monomials of logarithmic degree (see Lemma 6). We complement these findings by discovering other restricted settings that result in a classically efficiently solvable problem (see Table I). Second, their result provides an example of a matrix problem that is classically efficient when the precision is fixed (in the different setting of sample and query access to the guiding vector). Efficient classical constant-precision algorithms for ground-state problems have also been found in other contexts [39]. Yet, to our knowledge, the realm

of constant precision remains mostly unexplored for more general matrix algebra tasks.

Finally, let us summarize a few additional previous results on the hardness of matrix functions that directly fall within our setting. These are contextualized in Table I among our results. In the sparse access model, Janzing and Wocjan showed that estimating matrix elements of matrix monomials is BQP-COMplete for inverse error and monomial power scaling polylogarithmically with N , for A normalized by its operator norm [1]. However, the same problem was shown to be classically easy by Apers *et al.* when the matrix normalization is strengthened to the induced 1-norm, i.e., when the norm is decreased from $\|A\| \leq 1$ to $\|A\|_1 \leq 1$ [24,31]. We observe that the proof from Ref. [1] shows that relaxing the normalization condition from $\|A\|_1 \leq 1$ to $\|A\|_1 \leq 2$ brings back the BQP-completeness of the problem. This reflects a sharp transition in hardness and evidences, thereby, a form of tightness of the algorithm from Ref. [31], since relaxing its hypothesis to $\|A\|_1 = \mathcal{O}(1)$ breaks it without any possibility of fixing it (assuming $\text{BPP} \neq \text{BQP}$). An efficient classical algorithm analogous to that of Apers *et al.* for the Pauli access models (both for deterministic and random sampling access) was presented by Wang *et al.* [22], when A is normalized by its ℓ_1 -norm of the Pauli coefficients.

BQP-completeness of the normalized-state version of our local measurement problem was shown for $f(A) = A^{-1}$ by Harrow, Hassadim, and Lloyd (HHL) in their seminal paper [2], for the sparse access model, operator-norm matrix normalization, and constant precision $\varepsilon = \Omega(1)$. Employing their construction, we can prove BQP-completeness also for the non-normalized case under the same assumption of $1/\varepsilon = \mathcal{O}(1)$. In addition, we characterize the change in complexity of matrix inversion over the different settings we probe, i.e., we also look at the matrix element estimation problem (using a similar construction to that for monomials from Ref. [1]), sparse access versus Pauli access models, and different norms and sparsity regimes.

The universality of time evolution, meaning the capability of encoding any quantum circuit in a local Hamiltonian time evolution, is known from the original work by Feynman [32]. In particular, the circuit associated with any BQP problem can be mapped into a Hamiltonian and probabilistically implemented. Building upon Feynman's construction, Nagaj showed that a polynomial-sized circuit can be implemented using time evolution for a 3-local (sparse) Hamiltonian with $t = \text{poly}(n)$ [3]. This result naturally translates to the BQP-completeness of the local measurement problem with $f_t(A) = e^{-iAt}$. Here, we include a BQP-completeness proof using a slightly different Hamiltonian and extend the result to the matrix entry problem.

II. RESULTS

A. Basic definitions

In order to provide a precise summary of our results, we first need to define a few basic concepts and contextualize their use. We denote $\|A\| = \max_{\mathbf{x}: \|\mathbf{x}\|_2=1} \|A\mathbf{x}\|_2$ as the operator (or spectral) norm of A and $\|A\|_1 = \max_{\mathbf{x}: \|\mathbf{x}\|_1=1} \|A\mathbf{x}\|_1 = \max_{1 \leq j \leq N} \sum_{i=1}^N |A_{ij}|$ its induced 1-norm. Finally, for $A = \sum_{\ell} a_{\ell} P_{\ell}$ decomposed in the Pauli basis, we denote $\lambda_A = \sum_{\ell} |a_{\ell}|$ which we call the Pauli norm of A . We will use the symbol s to denote sparsity, L to denote number of Pauli terms, and ε to denote precision. We start by recalling the notion of function of a matrix:

Definition 1 (Function of a matrix (eigenvalue transformation)). Let $A \in \mathbb{C}^{N \times N}$ be a Hermitian matrix diagonalized as $A = S\Lambda S^{-1}$, for Λ real and diagonal and S unitary, and $f: \mathbb{R} \rightarrow \mathbb{C}$ some univariate function. Then $f(A) = Sf(\Lambda)S^{-1}$, where $f(\Lambda)$ is obtained by applying f to each diagonal element of Λ while leaving the off-diagonal ones untouched.

The two central tasks we study are computing entries of $f(A)$ or overlaps between a local measurement operator and a state vector transformed under $f(A)$, for different functions f . We now provide their definitions [40].

Problem I (Matrix element problem). Let $\{f_m\}_{m \in \mathbb{N}}$ be a family of functions. Given access to a Hermitian matrix $A \in \mathbb{C}^{N \times N}$ with bounded norm (such as $\|A\| \leq 1$ or $\|A\|_1 \leq 1$), two indices $i, j \in [N]$, a precision $\varepsilon > 0$ and a natural number $m \in \mathbb{N}$, compute an ε -approximation of $\langle i | f_m(A) | j \rangle$.

Problem II (Local measurement problem). Let $\{f_m\}_{m \in \mathbb{N}}$ be a family of functions. Given access to a Hermitian matrix $A \in \mathbb{C}^{N \times N}$ with bounded norm (such as $\|A\| \leq 1$ or $\|A\|_1 \leq 1$), a precision $\varepsilon > 0$ and a natural number $m \in \mathbb{N}$, compute an ε -approximation of $\langle 0 |^{\otimes n} f_m(A)^{\dagger} | 0 \rangle \langle 0 | \otimes \mathbb{1}_{N/2} f_m(A) | 0 \rangle^{\otimes n}$, where $|0\rangle\langle 0|$ is a single-qubit rank-1 projector and $\mathbb{1}_{N/2}$ is the $N/2 \times N/2$ identity matrix.

We note that we could consider more general matrices than Hermitian matrices, thus considering the more general singular value transformation [4,5] rather than the eigenvalue transform. Nevertheless, we restrict to this simpler setting because through a portion of our results we are interested in proving BQP-hardness. Hence, such hardness results will necessarily extend to more general cases if we restrict to more constrained formulations for the problems. In fact, our proofs of hardness will even hold for the further restricted class of real-symmetric matrices. Importantly, both above problems are efficiently simulable by the QSVT [4,5] and other methods for quantum eigenvalue transformations [8,22,41]. The problems can be considered natural and elemental applications of these algorithms

where one asks for a scalar classical output. Thus, the statements we make about either problem are informative on probing advantage for potential applications of the QSVT and related methods.

We study whether the matrix representation affects the difficulty of the problems. As discussed above, the two models we will consider are *sparse access* and *Pauli access*, defined as follows.

Definition 2 (Sparse access). A matrix $A \in \mathbb{C}^{N \times N}$ is a s -sparse matrix if it has at most s nonzero entries per row and column. In addition, if $s = \mathcal{O}(\text{poly log}(N))$, we refer to A simply as a sparse matrix. We say that we have classical sparse access to A if the following two conditions are satisfied:

- (i) We have efficiently computable functions $h_r, h_c : [N] \times [s] \rightarrow [N]$ such that $h_r(i, k)$ is the index of the k th nonzero entry of the i th row of A and $h_c(l, j)$ is the index of the l th nonzero entry of the j th column
- (ii) Given any $i, j \in [N]$, we can efficiently compute the entry A_{ij} .

Meanwhile, we say we have quantum sparse access to A if we have the following oracles:

$$\begin{aligned} O_{\text{row}} : |i\rangle|k\rangle &\rightarrow |i\rangle|h_r(i, k)\rangle, \\ O_{\text{col}} : |l\rangle|j\rangle &\rightarrow |h_c(l, j)\rangle|j\rangle, \\ O_A : |i\rangle|j\rangle|0\rangle^{\otimes b} &\rightarrow |i\rangle|j\rangle|A_{ij}\rangle, \end{aligned} \quad (1)$$

for $i, j \in [N], k, l \in [s]$.

We further note that efficient classical sparse access automatically implies efficient quantum sparse access. The quantum access can be granted via efficient boolean circuits for arithmetic operations [42], which can be mapped to reversible circuits [43], or via more modern quantum arithmetic circuits [44,45]. The soundness of this model was recently highlighted in Ref. [46], where the sparse access oracles are explicitly constructed for physically relevant Hamiltonian matrices.

Definition 3 (Pauli query access and Pauli-sparseness). Consider the decomposition of $A \in \mathbb{C}^{N \times N}$ as

$$A = \sum_{\ell=1}^L a_{\ell} P_{\ell}, \quad (2)$$

where each P_{ℓ} is a multi-qubit Pauli matrix (tensor product of single-qubit Paulis) and $a_{\ell} \in \mathbb{C}$. Then, Pauli query access consists of efficient classical access to the coefficients $\{a_{\ell}\}_{\ell \in [L]}$ and the *Pauli norm* $\lambda_A = \sum_{\ell=1}^L |a_{\ell}|$. Moreover, whenever $L, \lambda_A = \mathcal{O}(\text{poly log}(N))$ we say that the matrix is *Pauli-sparse*.

Importantly, we stress again that Pauli access is a special case of classical sparse access that can be natural for many problems. For example, qubit-mapped quantum chemistry or solid-state electronic Hamiltonians in second quantization are most commonly modeled in the Pauli basis [47]. Likewise, physical models of spin systems adopt a Pauli representation, and combinatorial optimization problems can also be encoded as spin Hamiltonian problems [48]. From a computational complexity perspective, the Pauli access model offers a fruitful ground since, on the one hand, there are Hamiltonians expressible as Eq. (2) which encode instances of QMA-COMplete problems even if each Pauli term in the sum is 2-local [49]. On the other hand, the algebraic properties of the Pauli group enable efficient classical algorithms—such as stabilizer circuit simulation [50,51], Pauli propagation [52–55], stabilizer-rank decompositions [56], and ZX-calculus [57]—to simulate certain quantum circuits. To the authors' knowledge, no efficient algorithm is known to compute the Pauli coefficients of a matrix from its computational basis representation, even under the promise that the matrix admits a sparse Pauli representation [58–60].

Finally, for QSVT algorithms, both sparse and Pauli access models are relevant because they are some of the few classical data accesses that are directly amenable to efficiently building quantum data accesses as block-encodings (via linear combinations of unitaries for the Pauli access) [6].

B. Summary of results

We now summarize our results. In this section we give informal versions of our results, quoted alongside prior results in the literature which also fall into our setting. We also point the reader to Table I which provides a further condensed visual summary. Full, formal statements along with proofs can be found in Sec. IV. Let us first highlight our main technical contributions. After summarizing our results we provide a summary of our conceptual contributions as a complement in the Discussion (Sec. II C).

- (a) We provide a simple-to-test scalar formula that determines whether entry estimation is BQP-hard for a family of functions $\{f_m\}_{m \in \mathbb{N}}$ and constant-sparse constant-local real-symmetric matrix (Eq. (3)).
- (b) We demonstrate an efficient classical algorithm for to return *any* polynomial-degree matrix polynomial for matrices that are sparse enough in the Pauli basis (P.2 / Theorem 6).
- (c) We give classical algorithms for polynomials and functions of matrices with suppressed norms, in a number of different settings (M.2.2 & M.3.1/Theorem 2 & Proposition 3), (P.3 / Theorem

8, [C.3.1](#) / Proposition 9), ([T.2.1-2](#) / Props. 16 & 17), and ([I.2](#) / Proposition 12).

In what follows, all hardness results quoted are applicable for real-symmetric matrices, whereas results for classical algorithms are applicable for general complex matrices. Following the convention stated in the caption of Table [I](#), results accompanied by a reference can be obtained from previous work, although in all cases we extend them with additional observations. If we believe that a certain result is somewhat folklore, we highlight this with an $\mathfrak{F}$ symbol. We start with the monomial function—the most elemental polynomial.

Summary 1 (Results for matrix monomials). Instantiate Problems I and II with $f_m(x) = x^m$. Unless explicitly stated, set m , $1/\varepsilon = \mathcal{O}(\text{poly log}(N))$, $\|A\| \leq 1$, and let A be either sparse or Pauli-sparse [i.e., either s or $L, \lambda_A = \mathcal{O}(\text{poly log}(N))$] depending on the access model. Then:

- (M.1) Problem I is BQP-COMPLETE when the input matrix A is given through either the sparse access model (Theorem 1 [\[1\]](#)) or the Pauli access model (Proposition 1), as is Problem II in both access models (Proposition 2). For the sparse access model this holds even for a choice of $s = \mathcal{O}(1)$. Moreover, in both cases, the result holds even if we add the condition that $\|A\|_1 \leq 2$ and A is 5-local.
- (M.2) Both problems can be solved efficiently classically in time $\mathcal{O}(\text{poly log}(N))$ whenever A is given through the sparse access model and satisfies either
 - (1) $s = 1$, or $s = \mathcal{O}(1)$ and $m = \mathcal{O}(\log \log N)$ (exact algorithm, Lemma 6 [\[11\]](#)), or
 - (2) $\|A\| \leq 1 - \eta$ with $\eta = \Omega(1)$, and either $s = \mathcal{O}(1)$ or $\varepsilon^{-1} = \mathcal{O}(1)$ (Theorem 2), or
 - (3) $\|A\|_1 \leq 1$ (Proposition 4 [\[31\]](#)).
- (M.3) Both problems can be solved efficiently classically in time $\mathcal{O}(\text{poly log}(N))$ whenever A is given through the Pauli access model and satisfies either:
 - (1) $\|A\| \leq 1 - \eta$, $\lambda_A = \mathcal{O}(1)$, and $\eta = \Omega(1)$ (Proposition 3), or
 - (2) $\lambda_A \leq 1$ (Proposition 5 [\[22\]](#))
- (M.4) Both problems can be solved efficiently classically in time $\mathcal{O}(\text{poly log}(N))$ if A either
 - (1) Has at most $\mathcal{O}(\text{poly log}(N))$ nonzero entries in the computational basis (Proposition 6 [\[3\]](#)), or
 - (2) Is given through the Pauli access model and $L = \mathcal{O}(\log \log N)$ (Theorem 3).

As mentioned in Sec. [1B](#), Problem I has already been considered for monomials $f_m(x) = x^m$ in Ref. [\[1\]](#), where it was shown that if A is given through sparse access and satisfies $\|A\| \leq 1$, the problem is BQP-COMPLETE. We show an analogous result for the Pauli Access model, and even when A is assumed to be Pauli-sparse. We also show that these two results hold for the local measurement problem too. In turn, when the norm condition $\|A\| \leq 1$ is strengthened to $\|A\| \leq 1 - \eta$, for any fixed $\eta > 0$, both problems become classically easy for both access models given a sparsity assumption. The same has been known to be true in the absence of an additional sparsity assumption for the alternative norm assumptions $\|A\|_1 \leq 1$ or $\lambda_A \leq 1$, which was shown in Refs. [\[31\]](#) and [\[22\]](#), respectively. Finally, we find the problem is classically easy to solve exactly if the matrix is made sparse enough in the Pauli basis.

The fact that a strengthening of the norm condition $\|A\| \leq 1$ to $\|A\| \leq 1 - \eta$ allows one to develop efficient classical algorithms suggests that the quantum advantage is related to handling the higher end of the spectrum, i.e., the eigenvectors with eigenvalues close to one. We note that the condition $\|A\| \leq 1 - \eta$ was also studied in Ref. [\[24\]](#) as a sufficient condition to construct a non-normalized quantum block encoding starting from sparse query access [\[61\]](#). This indicates that for certain problems, specific use of a block encoding could shroud superpolynomial quantum advantage.

We observe that the classical algorithm from [M.2.2](#) cannot be extended to matrices with $\|A\|_1 = \mathcal{O}(1)$ since the hardness results holds even when $\|A\|_1 \leq 2$ and $s = 4$ (Theorem 1 and Ref. [\[1\]](#)). This reflects a second regime in which there is a sharp transition in hardness with respect to the norm parameter. Explicitly, the Monte Carlo algorithm underlying [M.2.2](#) scales as $\mathcal{O}(\exp(2\delta m))$ for $\|A\|_1 = (1 + \delta)$. Thus, it is exponential for any matrix with induced 1-norm a constant δ above 1. We can understand this intuitively as a sign problem: when the induced 1-norm exceeds 1 the matrix entries of A are too oscillatory to classically keep track of efficiently. Meanwhile, quantum computers can efficiently process matrices with greater fluctuations: for instance, unitaries on n -qubits can have induced 1-norm as large as $2^{n/2}$. The fact that the regime of BQP-completeness is found for a norm value of 2 originates from the fact that we simulate 2-qubit gates, and we leave it as an open question as to whether or not it can be tightened to smaller norm values.

A first look at [\(M.1\)](#) indicates that for a certain variation of the problem, the BQP-hardness is robust with respect to the choice of access model, namely for the parameter settings of Ref. [\[62\]](#). However, result [\(M.4\)](#) shows a concrete difference in complexity between the Pauli and sparse access models in the regime when the sparsity is $o(\text{poly log}(N))$. As discussed, in the sparse access model constant row sparsity is in general BQP-HARD. In contrast, here we see that if A has structure in the Pauli basis

with $L = \mathcal{O}(\log \log(N))$, implying super-constant sparsity $s = \mathcal{O}(\log \log(N))$ (and total number of nonzero computational basis elements $\mathcal{O}(LN)$), the problem is classically easy. This demonstrates that Pauli structure makes problems easier than other efficiently computable instantiations of sparse accesses. The difference regarding the complexities between the access models for our classical algorithms (we demand $\text{poly} \log(N)$ nonzero entries for the sparse access model but $\mathcal{O}(\log \log N)$ for the Pauli one) is due to the fact that products between canonical projectors $|i\rangle\langle j|$ mostly vanish, while those between Pauli matrices do not.

Result (M.1) is obtained by observing that the reduction from Ref. [62] builds a matrix that is sparse also in the Pauli representation. It can also be seen that the matrix built in this reduction has 1-norm upper bounded by 2. We generalize these techniques to demonstrate a simple criterion for BQP-hardness for any function: entry estimation for any class of matrix functions parameterized by parameter m is BQP-HARD for inverse error $1/\varepsilon = \mathcal{O}(1/k)$ if its odd component f_m^o satisfies the condition

$$\frac{1}{M} \left| f_m^o(1) + 2 \sum_{\ell=1}^{\frac{M-1}{2}} f_m^o(\cos(2\pi\ell/M)) \right| \geq k, \quad (3)$$

for any $M = \mathcal{O}(\text{poly} \log(N))$ and the considered class of m [see Proof of Theorem 1 and Eq. (22) for more detailed discussion].

Meanwhile, for M.2.2 and M.3.1, we develop an efficient classical algorithm under the condition $\|A\| \leq 1 - \eta$ by observing that A^m tends to zero fast as m increases. In turn, to prove results M.2.3 and M.3.2, respectively based on the conditions $\|A\|_1 \leq 1$ and $\lambda_A \leq 1$, we use classical algorithms that are particular cases of algorithms from Refs. [24] and [22], which rely on Monte Carlo sampling and Markov-chain Monte Carlo. Finally, the results in (M.4) are proven by observing that, under the given sparsity conditions, A^m belongs to a low-dimensional subspace and can thus be efficiently represented explicitly for any monomial degree m . This result will allow us to simulate any polynomial of polynomial degree $\mathcal{O}(\text{poly} \log(N))$ (see Summary 3).

For Chebyshev polynomials we prove the following:

Summary 2 (Results for matrix Chebyshev polynomials). Instantiate Problems I and II with $f_m(x) = T_m(x)$, with T_m the Chebyshev polynomial of the first kind and degree $m \in \mathbb{N}$. Unless explicitly stated set $m, 1/\varepsilon = \mathcal{O}(\text{poly} \log(N))$, $\|A\| \leq 1$, and let A be sparse or Pauli-sparse depending on the access model (that is, s or L , $\lambda_A = \mathcal{O}(\text{poly} \log(N))$). Then,

(C.1) The problems are BQP-COMPLETE whenever the matrix A is given through either access model (Theorem 4 [3] and Proposition 7). For the sparse

access model this holds even for a choice of $s = \mathcal{O}(1)$. Moreover, in both cases, the result holds even if we add the conditions that $1/\varepsilon = \mathcal{O}(1)$, $\|A\|_1 \leq 2$ and A is 5-local.

(C.2) If $\text{BPP} \neq \text{BQP}$ both problems cannot be solved classically in polynomial time when the input matrix A is either

- (1) Given through the sparse access model and satisfies $\|A\|_1 = \mathcal{O}(1/\text{poly} \log(N))$ (Proposition 8 [3]).
- (2) Given through the Pauli access model, is Pauli-sparse and satisfies $\lambda_A = \mathcal{O}(1/\text{poly} \log(N))$ (Proposition 8).
- (*) The statement for Problem II requires the mild additional assumption that estimating the “1” measurement outcome is no harder than estimating the “0” measurement outcome (see Remark 2).

(C.3) Both problems can be solved classically in polynomial time via

- (1) An exact algorithm in the sparse access model whenever $s = \mathcal{O}(1)$ and $m = \mathcal{O}(\log \log N)$, a randomized algorithm in the sparse access model whenever $\|A\|_1 = \mathcal{O}(1)$ and $m = \mathcal{O}(\log \log N)$ or a randomized algorithm for the Pauli access model if $\lambda_A = \mathcal{O}(1)$ and $m = \mathcal{O}(\log \log N)$ (Proposition 9).
- (2) An exact algorithm in the sparse access model when A has at most $\text{poly} \log(N)$ nonzero entries in the computational basis (consequence of Proposition 6 [3]) or an exact algorithm in the Pauli access model whenever $L = \mathcal{O}(\log \log N)$ (consequence of Theorem 3).
- (3) A randomized algorithm in the Pauli access model when $\lambda_A = \mathcal{O}(1/(m^2 \log^{1.5}(N)))$ and the required precision is constant (Theorem 8).

The results from Summary 2 show a sharp contrast between monomials and the Chebyshev polynomials. We first see indication of this as for the latter the problem remains BQP-HARD even if we ask for a fixed precision $\varepsilon \leq \frac{1}{3}$. Furthermore, strengthening the normalization condition beyond $\|A\| \leq 1$ does not allow for polynomial-time classical algorithms (under the common assumption that $\text{BQP} \neq \text{BPP}$), which is a concrete separation from monomials for which we showed an efficient algorithm.

Result (C.1) is obtained again by using the clock construction of Ref. [62], but noting that the Chebyshev polynomials enhance the argument from that reduction. Specifically, denoting the clock matrix as H , the remarkably convenient fact that the eigenvalues of H happen to

coincide precisely with the extrema of T_M allows us to show that the problem is BQP-HARD to *constant* error.

The intractability results (C.2) are obtained for Problem I by exploiting the relationship between Hamiltonian Simulation and the Chebyshev polynomials expressed through the Anger-Jacobi expansion (see Definition A1). This is achieved through a polynomial-time Turing reduction (rather than a Karp reduction), and thus we do not claim BQP-completeness for the problem [63] under the constraints $\|A\|_1 \leq 1$ or $\lambda_A \leq 1$. As for Problem II, in Remark 2 we show that a mild generalization—namely, additionally estimating the probability of the “1” outcome in the local measurement on the non-normalized state—reduces to it the entry estimation problem for Chebyshev polynomials. Hence, under reasonable assumptions, we can expect estimating local measurements to be just as hard. Similar observations to Remark 2 hold also for other real functions of Hermitian matrices, including monomials, and the real exponential (relevant for imaginary-time evolution)

The classical algorithm of Result C.3.1 is obtained by observing that computing monomials is easy under the given hypothesis, and the error propagation due to the coefficients of the Chebyshev polynomials can be controlled because they can be bounded as $\mathcal{O}(4^m)$ for the m th polynomial. Finally, the results in C.3.2 are direct consequences of Proposition 6 and Theorem 3 (which we further generalize in the next Summary), while C.3.3 is due to a general algorithm based on importance-sampling sketching on the Pauli coefficients, which we also elucidate in the next summary.

The hardness results for monomials (and Chebyshev polynomials) are of course inherited by general polynomials. As for classical feasibility, for general polynomials, we can extend the ideas from Result (M.4) and obtain the following classical algorithms:

Summary 3 (Classical eigenvalue transform). Instantiate Problems I and II with any degree- d polynomial.

- (P.1) If A has $\mathcal{O}(\text{poly log}(N))$ nonzero matrix elements in the computational basis both problems can be classically solved exactly in $\mathcal{O}(d^2 \cdot \text{poly log}(N))$ time (Proposition 6 [3]).
- (P.2) If A has $L = \mathcal{O}(\log \log(N))$ nonzero coefficients in the Pauli basis [implying that A is $\mathcal{O}(\log \log(N))$ -sparse] both problems can be classically solved exactly in $\mathcal{O}(d^2 \cdot \text{poly log}(N))$ time (Theorem 6).
- (P.3) Further suppose the polynomial has magnitude at most 1 on the domain $[-1, 1]$ and $\|A\| \leq 1$. If A has $L = \mathcal{O}(\text{poly log}(N))$ nonzero coefficients in the Pauli basis (or efficiently representable structure in the Pauli basis), both problems can be classically solved to constant error in $\mathcal{O}(d^2 \cdot \text{poly log}(N))$ time, for $\lambda_A = \mathcal{O}(1/(d^2 \log^{1.5}(N)))$ (Theorem 8).

Results (P.1) and (P.2) are direct extensions of Result (M.4). They provide classical algorithms for the eigenvalue transform for any polynomial, with runtimes that are polynomially equivalent to those of their corresponding quantum algorithms. Importantly, these techniques are also straightforwardly transferable to the quantum singular value transform for general (non-Hermitian) matrices for even functions. They demonstrate that no superpolynomial quantum advantage in the problem size is possible when considering matrices that are super sparse, particularly when given in the Pauli basis. Note that these results can be easily extended to general functions that can be efficiently approximated by polynomials, including the time evolution function and inverse function. This also highlights that the problems’ inputs must be incompressible beyond some level of Pauli sparsity if we wish to preserve quantum advantage. Moreover, in Proposition 7 we demonstrate that the setting of (P.2) can be relaxed to matrices which are *approximately sparse* in the Pauli basis, that is, dominated by $\mathcal{O}(\log \log(N))$ coefficients. In addition, we give an auxiliary physical perspective in Proposition 18 that demonstrates that the corresponding states in this problem have bounded entanglement.

Result (P.3) comes from a two-stage algorithm. First, we perform importance sampling on Pauli coefficients to obtain a “sketched” description of A in the Pauli basis. We then apply the algorithm of (M.4). We see that under the standard condition $d = \text{poly log}(N)$, it would suffice to ask for $\lambda_A \leq c$ for some $c = \mathcal{O}(1/\text{poly log}(N))$ to guarantee an efficient classical algorithm. We saw in Result C.2 that we should not expect much more from a classical algorithm, as solving the Chebyshev problem to nonconstant precision for $\lambda_A \leq \mathcal{O}(1/\text{poly log}(N))$ is hard under complexity-theoretic assumptions. We will see in the following two summaries that for the inverse and time evolution functions this setting can even be shown to be BQP-COMPLETE. We remark that there is still a regime on which efficient classical algorithms for computing the Chebyshev polynomials may still be possible: namely, whenever $\|A\|_1 \leq 1$ and the precision is constant.

We now consider the paradigmatic problem of time evolution, defined through the complex exponential function. For this, we prove the following:

Summary 4 (Results for time evolution). Instantiate Problems I and II with $f_t(x) = \exp(ixt)$. Unless explicitly stated, set $t, 1/\varepsilon = \mathcal{O}(\text{poly log}(N))$, $\|A\| \leq 1$, and let A be sparse or Pauli-sparse depending on the access model (that is, s or $L = \mathcal{O}(\text{poly log}(N))$). Then, it holds that

- (T.1) The problems are BQP-COMPLETE for some fixed precision $1/\varepsilon = \mathcal{O}(1)$ (Propositions 13 and 14) whenever the matrix A ,

- (1) Is given through the sparse access model and satisfies $\|A\|_1 \leq 1$ or
 - (2) Is given through the Pauli Access model satisfies $\lambda_A \leq 1$ and is Pauli-sparse.
 - (3) Has even more strongly contracted norm $\|A\|_1 = \mathcal{O}(1/\text{poly log}(N))$ (hence $\|A\| = \mathcal{O}(1/\text{poly log}(N))$ also) in sparse access or $\lambda_A = \mathcal{O}(1/\text{poly log}(N))$ in Pauli access (Proposition 15).
- (T.2) The problems can be solved efficiently classically in $\mathcal{O}(\text{poly log}(N))$ time via
- (1) A randomized algorithm whenever $\|A\|_1 t = \mathcal{O}(\log \log N)$ in the sparse access model or $\lambda_A t = \mathcal{O}(\log \log N)$ in the Pauli access model (Proposition 16).
 - (2) A deterministic algorithm whenever $\|A\| t = \mathcal{O}(\log \log N)$ and A is $\mathcal{O}(1)$ -sparse (Proposition 17).
 - (3) A deterministic algorithm in the Pauli access model whenever $L = \mathcal{O}(\log \log N)$ (Theorem 6).
- (I.1) The problems are BQP-COMPLETE (Theorem 5 and Proposition 11 [2]) for a fixed precision whenever the matrix A is either
 - (1) Given through the sparse access model and satisfies $\|A\|_1 = \mathcal{O}(1/\text{poly log}(N))$ (hence $\|A\| = \mathcal{O}(1/\text{poly log}(N))$ also) or
 - (2) Given through the Pauli access model, satisfies $\lambda_A = \mathcal{O}(1/\text{poly log}(N))$ and is Pauli-sparse.
 - (I.2) The problems can be solved classically in polynomial time
 - (1) For the sparse access model whenever $s, m = \mathcal{O}(1)$ (exactly) or $\|A\|_1 \leq 1, m = \mathcal{O}(1)$ (approximately) (Proposition 12).
 - (2) For the Pauli access model whenever $\lambda_A, m = \mathcal{O}(1)$ (Proposition 12).

Similar to Chebyshev polynomials, here hardness holds robustly in a variety of settings, even when matrix norms are small and precision is a constant. As discussed previously, measurement of time-evolved states can be considered the canonical BQP-COMPLETE problem, dating back to the landmark proposal of Feynman [64]. To show BQP-completeness for the entry estimation variant of the problem for (T.1), we use a Hamiltonian clock construction of Ref. [65]. Result (T.2) arises due to an application of the classical algorithms of Refs. [31] and [22] and our generalizations thereof to the local measurement problem (M.2.1 and M.3) applied to the truncated Taylor expansion strategy of Ref. [66].

We also consider the inverse function. To frame it into the format of Problems I and II, we consider the family of functions given by

$$\text{inv}_m(x) = \begin{cases} \frac{1}{x} & \text{if } |x| \geq \frac{1}{m}, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

The family inv_m coincides with the inverse function for any A whose condition number $\kappa_A = \|A\| \|A^{-1}\|$ is upper bounded by m , and thus we will assume that the input matrix A satisfies $\kappa_A \leq m$.

Summary 5 (Results for matrix inversion). Instantiate Problems I and II with $f_m(x) = \text{inv}_m(x)$. Unless explicitly stated, set $m, 1/\varepsilon = \mathcal{O}(\text{poly log}(N))$, $\|A\| \leq 1$, and let A be sparse or Pauli-sparse depending on the access model (that is, s or $L = \mathcal{O}(\text{poly log}(N))$). Then, it holds that

The hardness part of Results (I.1) is proven, once again, by exploiting the construction from Ref. [62] in the case of entry estimation and by using the construction of Ref. [2] for the local measurement problem with a small tweak to hone the hardness result to Hermitian (real symmetric) matrices. These results are analogous to (C.1) and (T.1), since it shows that even a strengthening of the condition $\|A\| \leq 1$ does not lead to efficient classical algorithms for the case of the inverse function. This intractability result holds even for a constant-precision level, but, unlike the case of Chebyshev polynomials, this is less surprising since we are considering additive precision and the inverse function commutes with scalar multiplication [67].

Result (I.2) is deduced by employing a low-degree polynomial approximation of the inverse function in the range $[-1, -(1/m)] \cup [1/m, 1]$, which was developed in Ref. [4] and by approximating that polynomial with the classical algorithms in Results (M.2) and (M.3). We note that the classically solvable case is not immediately evident because, even when m is fixed, the precision is still a parameter of the problem. Therefore, we cannot pick a fixed polynomial approximation of the inverse function (we need to pick an ε -approximation of the inverse function). Meanwhile, for the case of monomials and the Chebyshev polynomials, whenever m is fixed the resulting function is a polynomial, and those can be computed exactly and straightforwardly classically if the matrix is sparse or Pauli-sparse (Lemma 6).

C. Discussion

The ultimate goal of quantum algorithm development is to identify problems of practical interest for which there is a large quantum advantage in terms of computational resources, if possible quantified in an end-to-end fashion, accounting for all necessary steps and not just

specific subroutines. Focusing on rigorous worst-case estimates, we have aimed at identifying what mathematical structure needs to be present in the input data and problem parameters for any significant quantum advantage to be *in principle* available. Concretely, we looked at the complexity of matrix functions for the matrix element problem (Problem I) and the local measurement problem (Problem II)—which gives a basic but, in our opinion, comprehensive framework for analyzing the potential of proposed quantum algorithms in different regimes of interest. Analyzing the impact of tuning a variety of problem-relevant parameters, as summarized in Fig. 1 and Table I, we emphasize the following takeaways around classical simulability and hardness.

First, besides being useful as building blocks for approximating more general functions, as we argued before, monomials and Chebyshev polynomials also directly arise in classical/quantum walks [68] as well as in quantum Krylov methods [69–72]. For monomials and Chebyshev polynomials especially, previous results [1] along with ours (see Results M.1, C.1, and C.2) show that it is unlikely that certain quantum Krylov methods are classically simulable in desired parameter regimes.

Second, we identified a natural special case of sparse access which can render a subclass of otherwise BQP-complete problems classically easy: Pauli access (P.2/Theorems 3 & 6). This highlights the importance of specifying precisely the access model when pondering quantum advantage, since a more specific model can be used to develop efficient classical algorithms under regimes for which the same problem might be hard in worst-case in other models.

Third, we made concrete a hierarchy of hardness for matrix functions. Specifically, in the sparse access model, we show hardness for Chebyshev polynomials, time evolution, and matrix inversion for $\|A\| \leq 1/\text{poly log}(N)$ and $s = \mathcal{O}(1)$ (e.g., see C.2/Theorem 8 for Chebyshev polynomials). Meanwhile, this regime is classically easy for monomials up to even constant-suppressed norm (M.2.2 / Theorem 2); and analogous statements hold true in the Pauli access model (see [M.3.1 / Proposition 3] for classical algorithm). Similarly, we have also proven a concrete complexity separation in the regime $s = \mathcal{O}(\text{poly log}(N))$ as we actually show hardness for the same three functions under the even stronger normalization condition on the induced 1-norm $\|A\|_1 \leq 1/\text{poly log}(N)$, which contrasts with classically easy randomized algorithms for monomials that efficiently process $\mathcal{O}(\text{poly log}(N))$ -sparse matrices whenever $\|A\|_1 \leq 1$ [M.2.3 / Proposition 4 [31]] and, again, with analogous statement for Pauli access [M.3.2/ Proposition 5 [22]].

Fourth, we stress that, in the sparse access model, the results from Ref. [1] tell us that our known capabilities for

randomized classical algorithms for monomials are (essentially) tight in dependence on $\|A\|_1$, in that the problem under the condition $\|A\|_1 \leq 1$ is in BPP but whenever $\|A\|_1 \leq 2$ is BQP-COMplete. We extend this also to the local measurement problem [M.1 / Proposition 2].

Fifth, so far we find no concrete evidence in our settings of a difference in complexity between local measurement and matrix entry estimation—we argue that the one gap in Table I can be filled by a slightly modified definition of the local measurement problem in Remark 2. One can also ask whether there is a concrete complexity separation between the normalized and unnormalized local measurement problem. While we do not observe a conclusive separation, so far we demonstrate that standard techniques yield BQP-hardness for monomials for *constant* error for the normalized local measurement problem (Proposition A1, Appendix A), whereas in our results we show BQP-hardness for *inverse-polynomial* error for the unnormalized version [M.1 / Proposition 1].

Last, we note that all of our classical algorithms for matrices given in Pauli basis actually allow for processing of arbitrary matrices with rank super-polylogarithmic in N . Thus, in this sense, “high rank” does not guarantee quantum advantage, particularly when there is enough structure in Pauli basis. However, we also often exploit this structure for quantum algorithms, so care should be taken. Overall, we reinforce the idea that very specific matrix structure is needed for potential significant quantum speedup. However, there are also still plentiful and significant gaps in the complexity landscape to further explore. Namely, some open questions motivated by our work are as follows: Can we find concrete differences in complexity between the local measurement and the entry estimation problems? One might then also study the normalized versions of the former. Further, are our classical Pauli algorithms for monomials tight in the way that the algorithms in the sparse access model are in the regime of constant sparsity? That is, can efficient classical algorithms be extended to $\lambda_A \geq 1$; or can the BQP-completeness result be strengthened to $\lambda_A = \mathcal{O}(1)$ from $\lambda_A = \mathcal{O}(\text{poly log}(N))$? More generally, we have seen a recurring theme through our results: that structure in the Pauli basis can be a powerful tool for classical algorithms. We leave it open whether there can be other interesting methods to exploit Pauli structure for classical algorithms.

Note added. In the version 3 arXiv update, Ref. [24] discusses a new idea to prove BQP-hardness of the entry estimation problem for matrix polynomials with $\|A\| \leq 1$ in the sparse access model. At present, there is a gap in the proof strategy, as only a proof of existence of a circuit-to-matrix mapping is known. It remains to be seen if there is an *efficient* mapping for their approach, which is needed to complete a proof of BQP-hardness.

III. TECHNICAL BACKGROUND

In this section, we present some standard definitions and results from the literature that will be useful for establishing our results in the next section. We also elucidate some generalizations thereof, which will be needed to consider the local measurement problem.

A. BQP and computational complexity

For completeness, we start by recalling the definitions of BQP, BQP-HARD, and BQP-COMLETE. Then, function approximation results and classical algorithms are recalled.

Definition 4 (Promise problems and BQP). A promise problem Π is given by two disjoint subsets of the set of binary strings $\Pi_{\text{yes}}, \Pi_{\text{no}} \subseteq \{0, 1\}^*$ that represent the set of positive and negative instances of Π , respectively [73].

BQP is the class of promise problems that can be solved by a uniform family of poly-sized quantum circuits [74]. More precisely, a promise problem $\Pi = (\Pi_{\text{yes}}, \Pi_{\text{no}})$ is in BQP whenever there is a family of circuits $\{C_n\}_{n \in \mathbb{N}}$ [where C_n act on $r(n) = \text{poly}(n)$ qubits and has $\text{poly}(n)$ gates], a classical algorithm able to compute a description for C_n in time $\mathcal{O}(\text{poly}(n))$ and these circuits satisfy that given $x \in \{0, 1\}^n$ it is the case that

$$C_n|x\rangle|0\rangle^{\otimes r(n)-n} = \alpha_{x,0}|0\rangle|\psi_{x,0}\rangle + \alpha_{x,1}|1\rangle|\psi_{x,1}\rangle, \quad (5)$$

where $|\psi_{x,0}\rangle, |\psi_{x,1}\rangle$ are $r(n) - 1$ qubit states and

- (1) If $x \in \Pi_{\text{yes}}$ it holds that $|\alpha_{x,1}|^2 \geq \frac{2}{3}$.
- (2) If $x \in \Pi_{\text{no}}$ it holds that $|\alpha_{x,1}|^2 \leq \frac{1}{3}$.

In other words, a promise problem is within the class BQP if there exists a family of efficient quantum algorithms that solves the problem with high probability (at least $2/3$ for all problem instances). This probabilistic aspect is important to accommodate the intrinsic random nature of quantum measurements.

We will also need the notions of reductions, hardness, and completeness.

Definition 5 (Karp reductions, BQP-HARD, and BQP-COMLETE). A promise problem $\Pi = (\Pi_{\text{yes}}, \Pi_{\text{no}})$ is Karp-reducible to the promise problem $\Pi' = (\Pi'_{\text{yes}}, \Pi'_{\text{no}})$ whenever there is a classical polynomial-time algorithm $\mathcal{A}$ such that

- (1) If $x \in \Pi_{\text{yes}}$ it holds that $\mathcal{A}(x) \in \Pi'_{\text{yes}}$.
- (2) If $x \in \Pi_{\text{no}}$ it holds that $\mathcal{A}(x) \in \Pi'_{\text{no}}$.

A promise problem Π' is BQP-HARD whenever all promise problems $\Pi \in \text{BQP}$ are Karp-reducible to Π' . If a promise problem Π is both BQP and BQP-HARD then it is BQP-COMLETE.

With these, it is straightforward to prove that there is at least one BQP-COMLETE problem.

Observation 1. The following problem is (naturally) BQP-COMLETE:

Problem: BQPCIRCUITSIMULATION

Input: An n -bit string x and a circuit $C = U_T \dots U_1$, with $T = \text{poly}(n)$ and each U_i acting nontrivially on at most 3 qubits (and, therefore, admitting a $\mathcal{O}(1)$ classical description), that acts on $r = \text{poly}(n) \geq n$ qubits as $C|x\rangle|0\rangle^{\otimes r-n} = \alpha_{x,0}|0\rangle|\psi_{x,0}\rangle + \alpha_{x,1}|1\rangle|\psi_{x,1}\rangle$, where $|\psi_{x,0}\rangle, |\psi_{x,1}\rangle, \alpha_{x,1}$, and $\alpha_{x,0}$ are all unknown except for the promise that either $|\alpha_{x,1}|^2 \geq \frac{2}{3}$ or $|\alpha_{x,1}|^2 \leq \frac{1}{3}$.

Output: Decide whether $|\alpha_{x,1}|^2 \geq \frac{2}{3}$ or rather $|\alpha_{x,1}|^2 \leq \frac{1}{3}$.

B. Hamiltonian simulation for sparse and Pauli access

We will also invoke known connections between our quantum access models and another important oracle for quantum algorithms. Specifically, sparse and Pauli access models both can be employed to simulate the time evolution e^{-iAt} .

Lemma 1 (Hamiltonian simulation for Sparse and Pauli access). Given sparse access to s -sparse $A \in \mathbb{C}^{N \times N}$ it is possible to construct a circuit U such that $\|e^{-iAt} - U\| \leq \alpha$ in time polynomial in $(1/\alpha, t, s, \|A\|, \log N)$ [75,76].

Similarly, given Pauli access to A with Pauli norm λ_A and number of Pauli matrices L it is possible to construct a circuit U such that $\|e^{-iAt} - U\| \leq \alpha$ in time polynomial in $(\log \frac{1}{\alpha}, t, \log N, L, \lambda_A)$ [6,66].

C. Polynomial approximations

Now, we introduce some general and well-known techniques for approximating the inverse function and the complex exponential with polynomials. These techniques are relevant because most of our classically simulable results will be based on approximating these functions by low-degree polynomials and using classically efficient algorithms to compute monomials or Chebyshev polynomials.

Lemma 2 (Polynomial approximation of $1/x$ ([28, Lemmas 17 and 19])). The function

$$g(x) = \frac{1 - (1 - x^2)^b}{x}$$

ε -approximates the function [77] $f(x) = 1/x$ in the domain $[-1, -(1/\kappa)] \cup [1/\kappa, 1]$ for any $b \geq \kappa^2 \log(\kappa/\varepsilon)$. Moreover, a polynomial of degree $\mathcal{O}(\kappa \log(\kappa^2/\varepsilon))$ that ε -approximates $f(x)$ can be obtained from $g(x)$.

Lemma 3 (Polynomial approximation of e^{ixt} ([4, Lemmas 57 and 59])). Let $t \in \mathbb{R} \setminus \{0\}$ and $\varepsilon \in (0, 1/e)$.

Then, the polynomial $J_0(t) + 2 \sum_{k=1}^R t^k J_k(t) T_k(x)$ of degree $R = \Theta(t + \log(1/\varepsilon)/\log(e + \log(1/\varepsilon)/t))$ is a 2ε -approximation to the function e^{itx} , where $J_k(t)$ are Bessel functions and $T_k(x)$ Chebyshev polynomials of the first kind.

D. Quantum algorithms for matrix function estimation

In Ref. [62], the authors present an algorithm to compute $\langle j | f(A) | j \rangle$ based on phase estimation. Briefly, computing $\langle j | f(A) | j \rangle$ amounts to computing $\sum_l f(\theta_l) |\langle j | u_l \rangle|^2$ where the state $|u_l\rangle$ represents the eigenvector of A with corresponding eigenvalue θ_l . Phase estimation allows sampling of the eigenvalue θ_l with probability $|\langle j | u_l \rangle|^2$, and, therefore, we obtain an estimator of $\langle j | f(A) | j \rangle$ as long as we can control: (1) the error propagation that occurs when applying f to the approximated eigenvalue computed by phase estimation and (2) the maximum error that may happen with some probability if phase estimation completely fails. Condition (1) can be achieved by bounding the Lipschitz constant K_f of f , while (2) is obtained by bounding $\|f\|_\infty$. This strategy is formalized below:

Lemma 4 (Quantum algorithm for entry estimation (Janzing/Wocjan [62, Lemma 2])). Let $A \in \mathbb{C}^{N \times N}$ be a Hermitian matrix such that $\|A\| \leq 1$ and let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ be a function satisfying $|f(x) - f(y)| \leq K_f |x - y|$ for all $x, y \in I$, where K_f is a constant. Let a circuit U be given such that $\|U - \exp(iA)\| \leq \alpha$ using resources that scale polynomially in $\log(N)$ and $1/\alpha$. Then, given a state $|\psi\rangle$ whose decomposition into A -eigenvectors contains eigenvalues only in the interval I we can estimate $\langle \psi | f(A) | \psi \rangle$ up to error $\varepsilon(\|f\|_\infty + K_f)$ with probability at least $1 - \delta$ with time and space resources polynomial in $\log(N)$, $1/\varepsilon$ and $\log(1/\delta)$.

For matrices given by the access models of Definitions 2 and 3, Lemma 1 gives efficient ways of constructing the operator U . Therefore, in these cases, for any function satisfying $\|f\|_\infty, K_f = \mathcal{O}(\text{poly log}(N))$, this algorithm runs in polynomial time if also $1/\varepsilon = \mathcal{O}(\text{poly log}(N))$. For instance, when $f_m(x) = x^m$ it is the case that $K_{f_m} = m$ and $\|f_m\|_\infty = 1$, and the BQP algorithm follows. Since this algorithm computes $\langle \psi | f(A) | \psi \rangle$ for any $|\psi\rangle$, it is easy to estimate any $\langle i | f(A) | j \rangle$ observing that these nondiagonal terms can be expressed as a sum of “diagonal” ones in different bases (Lemma A2).

We demonstrate a quantum algorithm for the local measurement version of the problem.

Lemma 5 (Quantum algorithm for normalized local measurement). Consider a function $f : I \rightarrow [-f_{\max}, f_{\max}]$ which satisfies $|f(x) - f(y)| \leq K_f |x - y|$ for all $x, y \in I$, and where the smallest discontinuity in I is of size b . The normalized local measurement $\langle f(A) | (|0\rangle\langle 0| \otimes$

$\mathbb{1}_{N/2}) | f(A) \rangle$ (denoting $|f(A)\rangle = f(A)|0\rangle/\|f(A)|0\rangle\|$) can be solved for matrix A with spectrum contained in I , given in sparse or Pauli access, to additive error $\varepsilon \leq \|f(A)|0\rangle\|b/2$ with cost polynomial in $(K_f, 1/\|f(A)|0\rangle\|, \log N, 1/\varepsilon)$.

Proof. Denote the eigendecomposition of A as $A = \sum_i \theta_i |u_i\rangle\langle u_i|$ and write the zero state in this basis as $|0\rangle = \sum_i \beta_i |u_i\rangle$. Let us now consider the following sequence of operations:

$$\begin{aligned} |0\rangle|0\rangle|0\rangle &= \sum_i \beta_i |u_i\rangle|0\rangle|0\rangle \xrightarrow{\text{QPE}} \sum_i \beta_i |u_i\rangle|\tilde{\theta}_i\rangle|0\rangle \\ &\xrightarrow{\text{C-R, QPE}^{-1}} \sum_i \beta_i |u_i\rangle \left(\frac{f(\tilde{\theta}_i)}{f_{\max}} |0\rangle + \sqrt{1 - \frac{f^2(\tilde{\theta}_i)}{f_{\max}^2}} |1\rangle \right) \end{aligned} \quad (6)$$

$$\xrightarrow{\text{AA}} \sum_i \frac{f(\tilde{\theta}_i)}{\|f(A)|0\rangle\|} \beta_i |u_i\rangle \approx \frac{1}{\|f(A)|0\rangle\|} f(A)|0\rangle, \quad (7)$$

where QPE denotes quantum phase estimation to an error $|\tilde{\theta}_i - \theta_i| \leq \varepsilon'$ for all i (with cost $\mathcal{O}(1/\varepsilon')$ [78–80]), C-R denotes a rotation of the third register controlled on the second register, and AA denotes amplitude amplification [81] by factor $f_{\max}/\|f(A)|0\rangle\|$. We remark that $\tilde{\theta}_i$ may fall outside of I by ε' —this can be resolved by extending f to be a total function whose value outside of $x \notin I$ corresponds to $f(y)$ of $y \in I$ closest to x . We then should only consider $\varepsilon' \leq b/(2K_f)$. The Lipschitz condition ensures that the output state is an approximation of $f(A)|0\rangle/\|f(A)|0\rangle\|$ with additive error $\varepsilon'K_f/\|f(A)|0\rangle\|$ in ℓ_2 -norm. The total cost in cumulative Hamiltonian simulation time of QPE + AA is $f_{\max}/(\|f(A)|0\rangle\|\varepsilon')$, which can be simulated with linear cost using the algorithm of [82]. The stated result can be checked to follow by choice of $\varepsilon' = \varepsilon\|f(A)|0\rangle\|/(2K_f)$. ■

E. Classical algorithms for matrix functions

We start by stating a lemma on sparse-matrix multiplication. This same idea was used to classically compute matrix polynomials in Ref. [11].

Lemma 6 (Classical algorithms for matrix powers). Given sparse access to a $N \times N$ s -sparse matrix A , for any indices (i, j) it is possible to compute $[A^m]_{ij}$ exactly in time $\mathcal{O}(s^m)$ classically. Similarly, $\langle i | A^{m_1} \pi A^{m_2} | j \rangle$ can be computed exactly in time $\mathcal{O}(s^{m_1+m_2})$ classically.

Proof. The first statement is possible by using a matrix multiplication algorithm recursively,

- (1) If $m = 1$ then return $A_{i,j}$.
- (2) Else, find the ℓ nonzero entries of the i th row of A , and name their positions $k_1, \dots, k_\ell$, where $\ell \leq s$. Then compute recursively the entries $\{(k_1, j), \dots, (k_\ell, j)\}$ of A^{m-1} , and return $A_{i,k_1} A_{k_1,j}^{m-1} + \dots + A_{i,k_\ell} A_{k_\ell,j}^{m-1}$.

Let $R(m)$ be the runtime for this algorithm given the degree m . Then $R(m) = sR(m-1) + s$ and, therefore, $R(m) = \mathcal{O}(s^m)$. We can also use this same idea for $\langle i|A^{m_1}\pi A^{m_2}|j\rangle$, namely,

- (1) If $m_1 = m_2 = 0$ we return $\langle i|\pi|j\rangle = \langle i|(|0\rangle\langle 0| \otimes \mathbb{1}_{N/2})|j\rangle$ which can be computed straightforwardly.
- (2) If $m_2 > 0$ we note that

$$[A^{m_1}\pi A^{m_2}]_{i,j} = [A^{m_1}\pi A^{m_2-1}]_{i,t_1} A_{t_1,j} + \dots + [A^{m_1}\pi A^{m_2-1}]_{i,t_p} A_{t_p,j}, \quad (8)$$

where $t_1, \dots, t_p$ with $p \leq s$ are the row indices of the nonzero entries of column j of A .

- (3) If $m_2 = 0$ and $m_1 > 0$ we apply an analogous strategy.

This algorithm has complexity $\mathcal{O}(s^{m_1+m_2})$. ■

The inspiration to consider different normalization factors comes from recent classical algorithms that allow computing $p(A)$ for any polynomial p and whose complexity depends on $\|A\|_1$, the Pauli norm λ_A and the coefficients of the polynomial p . Roughly, these algorithms work by considering the matrix A as a description of a Markov chain, and thus, Monte Carlo techniques can estimate any entry of A^m .

Lemma 7 (Classical sampling algorithm for classical sparse access ([24, Proposition 5.5], [31, Lemma 3.4])). Let $f(x) = \sum_{r=0}^m \alpha_r x^r$. Then, there is an algorithm that, given sparse access to an s -sparse matrix A and two indices i, j , computes an ε -approximation of $\langle i|f(A)|j\rangle$ with probability at least $1 - \delta$ in time

$$\mathcal{O}\left(\frac{ms}{\varepsilon^2} \|f(\|A\|_1 x)\|_{\ell_1}^2 \log\left(\frac{1}{\delta}\right)\right),$$

where we denote $\|f(bx)\|_{\ell_1} = \sum_{r=0}^m |\alpha_r b^r|$.

Lemma 8 (Classical sampling algorithm for Pauli access (adapted from Ref. [22, Proposition 3])). Let $f(x) = \sum_{r=0}^m \alpha_r x^r$ and assume ℓ_1 -sampling access to the Pauli coefficients of $A = \sum_{\ell} a_{\ell} P_{\ell}$. That is, suppose $\lambda_A = \sum_{\ell=1}^L |a_{\ell}|$ is known and there is an efficient sampler who returns the tuple $(\ell, \text{sign}(a_{\ell}))$ with probability $\frac{|a_{\ell}|}{\lambda_A}$. Then there is an algorithm that, given two indices (i, j) ,

computes an ε -additive approximation of $\langle i|f(A)|j\rangle$ with probability at least $1 - \delta$ in time

$$\mathcal{O}\left(\frac{m \log(N)}{\varepsilon^2} \|f(\lambda_A x)\|_{\ell_1}^2 \log\left(\frac{1}{\delta}\right)\right),$$

where we denote $\|f(bx)\|_{\ell_1} = \sum_{r=0}^m |\alpha_r b^r|$. If no sampling access is available, it can be provided in $\mathcal{O}(L)$ time as a preprocessing step starting from Pauli access.

The above two algorithms can be adapted to the local measurement problem:

Lemma 9 (Classical sampling algorithm for both access models and local measurement). Given $f(x) = \sum_{r=0}^m \alpha_r x^r$, there exist classical algorithms that yield ε -additive approximations to $\langle i|f(A)\pi f(A)|i\rangle$ for $\pi = |0\rangle\langle 0| \otimes \mathbb{1}_{N/2}$ with probability at least $1 - \delta$ in time

$$\begin{aligned} &\tilde{\mathcal{O}}\left(\frac{ms}{\varepsilon^2} \|f(\|A\|_1 x)\|_{\ell_1}^4 \log\left(\frac{1}{\delta}\right)\right) \quad \text{and} \\ &\mathcal{O}\left(\frac{m \log(N)}{\varepsilon^2} \|f(\lambda_A x)\|_{\ell_1}^4 \log\left(\frac{1}{\delta}\right)\right), \end{aligned} \quad (9)$$

for A given in classical sparse access or Pauli access, respectively.

Proof sketch. The central primitive we need is a method to compute quantities of the form $\langle i|A^a \pi A^b|i\rangle$ —with this, the local measurement of any $f(A)$ which is a probabilistic combination of monomials can be efficiently returned by sampling over such quantities. This idea can then be extended to general linear combinations. For example, let us discuss obtaining $\langle i|A^a \pi A^b|i\rangle$ in Pauli access. We note that π is the convex sum of two Pauli strings, and thus we observe that $\langle i|A^a \pi A^b|i\rangle$ can be written as a linear combination of terms of the form $\langle i|P_{\ell_1} \dots P_{\ell_a} P_{\pi} P'_{\ell'_1} \dots P'_{\ell'_b}|i\rangle$, where each P_{χ} denotes some Pauli string. Considering the linear combination as a normalized probability distribution, we can sample from these terms (along with any accompanying phase) with variance upper bounded by $(\lambda_A^a \lambda_A^b)^2$. Each such term costs $\mathcal{O}((a+b) \log N) = \mathcal{O}(m \log N)$ time to explicitly and exactly evaluate. We provide a full proof in Appendix A 2. ■

Together, Lemmas 7, 8, 9 immediately imply (as a special case) efficient classical algorithms for monomials when problem parameters scale polylogarithmically ($m, 1/\varepsilon, s$ or $L = \mathcal{O}(\text{poly log}(N))$), when $\|A\|_1 \leq 1$ or $\lambda_A \leq 1$ for sparse access or Pauli ℓ_1 -sampling access [obtainable in $\mathcal{O}(L)$ preprocessing time], respectively.

IV. DETAILED STATEMENTS AND MAIN PROOFS

In this section, we study the complexity of different instantiations of Problems I and II. Each matrix function studied is presented in a separate subsection that begins by defining a promise problem form of the respective problem.

For brevity, we will use the following macros to refer to the different access models for a given matrix A :

- (a) SPARSEACCESS: sparse access to A , as per Definition 2, assuming that A is s -sparse.
- (b) PAULIACCESS: Pauli query access to A as per Definition 3, assuming that A is Pauli-sparse.
- (c) AMODEL: a placeholder for any access model when defining the problems.

We will use the symbol $b(A)$ as placeholder notation for any norm of A .

A. Monomials

In this section we present in detail our hardness results and classical algorithms for Problems I and II when $f_m(x) = x^m$, i.e., the family of matrix powers or monomials. Let us start with hardness results. To this end we adapt the formal definition of the promise problem from Ref. [1, Definition 4.1], where the authors introduce the DIAGONALENTREYESTIMATIONSPARSEACCESS problem. It consists in computing the value $[A^m]_{j,j}$, given a sparse and real-symmetric matrix A satisfying $\|A\| \leq 1$, an integer j and a power m . They prove this problem is BQP-COMPLETE when A is given in sparse access. We define an analogous problem, which will also allow us to analyze Pauli access.

Problem: $\text{MONOMIAL}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with norm $b(A) \leq 1$ and accessible through AMODEL, a positive integer m , index j , a precision ε and a threshold g , such that $m, 1/\varepsilon = \mathcal{O}(\text{poly log}(N))$, $g = \mathcal{O}(1)$. b is an upper bound on $\|A\|$, m and $\frac{1}{\varepsilon}$ are poly log (N) .

Output: YES if $[A^m]_{j,j} \geq g + \varepsilon$, NO if $[A^m]_{j,j} \leq g - \varepsilon$.

We also define the local measurement version of the problem.

Problem: $\text{LM-MONOMIAL}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with norm $b(A) \leq 1$ and accessible through AMODEL, a positive integer m , a precision ε and a threshold g , such that $m, 1/\varepsilon = \mathcal{O}(\text{poly log}(N))$, $g = \mathcal{O}(1)$. b is an upper bound on $\|A\|$, m and $\frac{1}{\varepsilon}$ are poly log (N) .

Output: Let $\pi = |0\rangle\langle 0| \otimes \mathbb{1}_{N/2}$ and $r = \langle 0|A^m\pi A^m|0\rangle$. Then, answer YES if $r \geq g + \varepsilon$ and NO if $r \leq g - \varepsilon$.

Note that, as in Ref. [1], we consider a more restricted class of matrices for the promise problems to show hardness: real-symmetric matrices, rather than Hermitian

matrices. First, we recount and adapt the main result and proof of Ref. [1] in a way that will be relevant for our discussions.

Theorem 1. The problem $\text{MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ is BQP-COMPLETE. The hardness result continues to hold even under the stronger conditions that the matrix A is 5-local, 4-sparse, satisfies $\|A\|_1 \leq 2$, and is real symmetric.

Proof. Regarding membership in BQP, the algorithm described in Lemma 4 can be employed to compute $\langle j | f_m(A) | j \rangle$ for any $j \in [N]$ when $f_m(x) = x^m$ in polynomial time using Lemma 1 and observing that $\|f_m\|_{\infty}^{[-1,1]} = 1$ and $K_{f_m} \leq m$. Thus, $\text{MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ is in BQP. Moreover, any nondiagonal entry can be expressed as a sum of diagonal terms on a different basis (see Lemma A2).

To prove hardness, consider $C = U_T \dots U_1$ as the r -qubit input circuit to BQPCIRCUITSIMULATION (Observation 1). The reduction we are going to show defines a sparse Hermitian matrix, for which it is possible to construct sparse access. Moreover, there will be a diagonal element of a monomial of that matrix such that the circuit accepts $|x\rangle$ if that entry contains a value above some threshold. We are free to pick the gate set from which C is composed; it will turn out that if the gates $U_1, \dots, U_T$ are assumed to be either Hadamard or Toffoli gates (both having real entries and forming a universal set of gates), then the reduction defines a real-symmetric matrix, which is a special case of a Hermitian matrix. From here on, and in all proofs to follow, we assume this decomposition. It will be useful to consider a new circuit C' obtained from C , defined as $C' = U_1^\dagger \dots U_T^\dagger (Z \otimes \mathbb{1}^{r-1}) U_T \dots U_1 =: V_{M-1} \dots V_0$ with $M = 2T + 1$ (see also Fig. 2).

We will use a unary clock to keep track of the computation steps, where state $|\text{step}_k\rangle = |0\rangle^{\otimes k} |1\rangle |0\rangle^{\otimes (M-k-1)}$ denotes the k th computation step. Thus, the k th clock transition (k to $k+1$) can be described by the operator $\mathcal{T}_k = \mathbb{1}^{\otimes k} \otimes |01\rangle\langle 10| \otimes \mathbb{1}^{\otimes (M-k-2)}$ for $k < M-1$, and $\mathcal{T}_{M-1} = |1\rangle\langle 0| \otimes \mathbb{1}^{\otimes (M-2)} \otimes |0\rangle\langle 1|$. It holds that

$$\mathcal{T}_\ell |\text{step}_k\rangle = \begin{cases} |\text{step}_{k+1}\rangle & k = \ell, \\ 0 & \text{otherwise,} \end{cases} \quad (10)$$

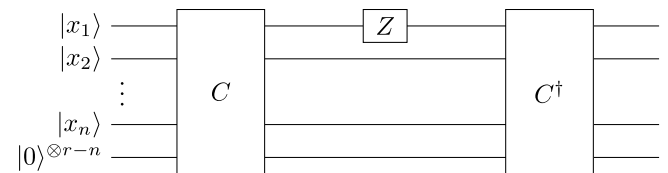


FIG. 2. Circuit extension C' of the input circuit C used in Ref. [62]. Note that the amplitude $\langle x | \langle 0 | C' | x \rangle | 0 \rangle$ is linearly related to the measurement probability that decides the problem. In turn, the estimation of such amplitude reduces to estimating an element of a monomial of a sparse matrix A , defined in Eq. (15).

(where $+$ is understood modulo M) and that, for any d ,

$$\prod_{k=0}^{M-1} \mathcal{T}_{k+d} = |\text{step}_d\rangle\langle\text{step}_d|, \quad (11)$$

where the addition $+$ is understood modulo M .

Let $|s_x\rangle = |\text{step}_0\rangle|\mathbf{x}\rangle|0\rangle^{r-n}$ with $|\mathbf{x}\rangle|0\rangle^{r-n}$ the input bit-string to BQPCIRCUITSIMULATION. The circuit C' operates on $|\mathbf{x}\rangle|0\rangle^{r-n}$ as the identity $\mathbb{1}$ if $|\alpha_{x,0}|^2 = 1$, while if $|\alpha_{x,1}|^2 = 1$ it behaves as $-\mathbb{1}$. We define

$$W = \sum_{\ell=0}^{M-1} \mathcal{T}_\ell \otimes V_\ell. \quad (12)$$

One can check that, due to Eq. (11),

$$W^M = \sum_{\ell=0}^{M-1} |\text{step}_\ell\rangle\langle\text{step}_\ell| \otimes V_{(\ell+M+1)} \dots V_\ell, \quad (13)$$

and, because $(C')^2 = \mathbb{1}$, we have that $(W^M)^2 = \sum_{\ell=0}^{M-1} |\text{step}_\ell\rangle\langle\text{step}_\ell| \otimes \mathbb{1}^{\otimes r}$. This implies that $(W^M)^2$ behaves as the identity in the subspace spanned by the set $\{|\text{step}_\ell\rangle\langle\text{step}_\ell| \otimes \mathbb{1}^{\otimes r}\}_{0 \leq \ell \leq M-1}$, and from now on we restrict all our analysis to that subspace. Note that W^M only has ± 1 eigenvalues there, and, therefore, let $\mathcal{S}^\pm$ be the W^M -invariant subspaces associated with the projectors $Q^\pm = \frac{1}{2}(\mathbb{1} \pm W^M)$.

We observe that the action of W over $\mathcal{S}^\pm$ is isomorphic to a cyclic shift (with a phase shift in the case of $-$). Starting from $Q^\pm|s_x\rangle$, the cycle travels across the vectors $W^\ell Q^\pm|s_x\rangle$ for $\ell = 0, \dots, M-1$. Using this property, we show in Lemma A5 that the eigenvalues of W take values $e^{i2\pi\ell/M}$ (for eigenstates in $\mathcal{S}^+$) and $e^{i\pi(2\ell+1)/M}$ (for eigenstates in $\mathcal{S}^-$). Moreover, we can evaluate the overlap

$$\omega_\ell^+ = \langle s_x | P_\ell^+ | s_x \rangle = \frac{|\alpha_{x,0}|^2}{M}, \quad (14)$$

where P_ℓ^+ is the projector onto $\mathcal{P}_\ell^+$, the eigenspace of W corresponding to eigenvalue $e^{i2\pi\ell/M}$. Similarly, $\omega_\ell^- = \langle s_x | P_\ell^- | s_x \rangle = |\alpha_{x,1}|^2/M$, where P_ℓ^- is the projector onto $\mathcal{P}_\ell^-$, the eigenspace of W corresponding to eigenvalue $e^{i\pi(2\ell+1)/M}$.

Now, consider the Hermitian (real-symmetric) matrix

$$A = \frac{W + W^\dagger}{2}, \quad (15)$$

where W is defined in Eq. (12) we note that A is sparse with sparsity $s = 4$ due to the fact that each V_ℓ is a either a Hadamard or Toffoli gate, and also 5-local because the clock transitions are 2-local and each circuit gate is at most 3-local. We can also check that $\|A\|_1 \leq 2$. Note also that each eigenvector $|\psi_\ell^+\rangle \in \mathcal{S}^+$ of W with eigenvalue $e^{i2\pi\ell/M}$ is also an eigenvector of $W^\dagger$, but with eigenvalue $e^{-i2\pi\ell/M}$, and the same happens for the eigenvectors $|\psi_\ell^-\rangle \in \mathcal{S}^-$. Therefore, $A|\psi_\ell^+\rangle = (e^{i2\pi\ell/M}|\psi_\ell^+\rangle + e^{-i2\pi\ell/M}|\psi_\ell^+\rangle)/2 = \cos(2\pi\ell/M)|\psi_\ell^+\rangle$ and similarly $A|\psi_\ell^-\rangle = \cos((\pi(2\ell+1))/M)|\psi_\ell^-\rangle$. We denote the eigenvalues of A as $\theta_\ell^+ = \cos(2\pi\ell/M)$ (corresponding to $\mathcal{P}_\ell^+$) and $\theta_\ell^- = \cos((\pi(2\ell+1))/M)$ (corresponding to $\mathcal{P}_\ell^-$) for $\ell = 0, \dots, M-1$. Some properties of these eigenvalues will be useful. First, $\theta_0^+ = 1$ and θ_ℓ^+ and $\theta_{M-\ell}^+$ coincide for $\ell = 1, \dots, \frac{M-1}{2}$ since $\cos(2\pi\ell/M) = \cos(2\pi(M-\ell)/M)$. Second, $\theta_{(M-1)/2}^- = -1$ and $\theta_\ell^- = \theta_{M-\ell-1}^-$ for $\ell = 0, \dots, (M-1)/2 - 1$ since $\cos(\pi(2\ell+1)/M) = \cos(\pi(2(M-\ell-1)+1)/M)$. Lastly, we also observe that $\theta_{(M-1)/2-\ell}^- = -\theta_\ell^+$ for $\ell = 0, \dots, (M-1)/2$. These properties yield the spectral decomposition

$$A = P_0^+ - P_{\frac{M-1}{2}}^- + \sum_{l=1}^{\frac{M-1}{2}} \theta_l^+ (P_l^+ + P_{M-\ell}^+ - P_{\frac{M-1}{2}-\ell}^- - P_{\frac{M-1}{2}+\ell}^-). \quad (16)$$

Denote $|j\rangle = |s_x\rangle$, with j the integer whose binary representation is s_x . Then, for a function $f_m(A)$

$$\begin{aligned} [f_m(A)]_{j,j} &= f_m(1)\omega_0^+ + f_m(-1)\omega_{\frac{M-1}{2}}^- + \sum_{l=1}^{\frac{M-1}{2}} f_m(\theta_l^+) (\omega_\ell^+ + \omega_{M-\ell}^+) + f_m(-\theta_\ell^+) (\omega_{\frac{M-1}{2}-\ell}^- + \omega_{\frac{M-1}{2}+\ell}^-) \\ &= \frac{|\alpha_{x,0}|^2}{M} \left[f_m(1) + 2 \sum_{l=1}^{\frac{M-1}{2}} f_m(\theta_l^+) \right] + \frac{|\alpha_{x,1}|^2}{M} \left[f_m(-1) + 2 \sum_{l=1}^{\frac{M-1}{2}} f_m(-\theta_l^+) \right], \end{aligned} \quad (17)$$

where we employed Eqs. (16) and (14), and the fact that the projectors are orthogonal. Thus, for $f_m(A) = A^m$, with m odd, we can explicitly write

$$[A^m]_{j,j} = \frac{(1 - 2|\alpha_{x,1}|^2)}{M} \left[1 + 2 \sum_{l=1}^{\frac{M-1}{2}} (\theta_l^+)^m \right] \\ := (1 - 2|\alpha_{x,1}|^2) E_0, \quad (18)$$

where we have used the fact that $1 - |\alpha_{x,1}|^2 = |\alpha_{x,0}|^2$ and denoted $E_0 = (1/M)(1 + 2 \sum (\theta_l^+)^m)$.

Note that if $|E_0|$ is not too small, then by computing $[A^m]_{j,j}$ one can recover the acceptance probability $|\alpha_{x,1}|^2$ of the original circuit C . Precisely, observe that

$$E_0 \geq \frac{1}{M} \left(1 + 2 \cdot \frac{M-1}{2} \left(\theta_{\frac{M-1}{2}}^+ \right)^m \right) \geq \frac{1}{M} + \left(\theta_{\frac{M-1}{2}}^+ \right)^m, \quad (19)$$

where the first inequality follows by observing that the eigenvalues are enumerated in decreasing order. Since, $\theta_{\frac{M-1}{2}}^+ = \cos(\pi(M-1)/M) < 0$ and m is odd, we need to take m big enough for the right-hand side of Eq. (19) to be sufficiently positive. One can check that by picking $m = M^3$ (which is odd) we can ensure that $E_0 > 3/4M$. Finally, using Eq. (18) this implies that if $|\alpha_{x,1}|^2 \leq \frac{1}{3}$ then $[A^m]_{j,j} \geq E_0/3 > 1/4M$, and whenever $|\alpha_{x,1}|^2 \geq \frac{2}{3}$ then $[A^m]_{j,j} \leq -(E_0/3) < -(1/4M)$. Thus, we can decide which of the two cases holds for $|\alpha_{x,1}|^2$ by computing a $\frac{1}{4M}$ -approximation of $[A^m]_{j,j}$, where we recall that $M = \mathcal{O}(\text{poly log}(N))$.

The Karp mapping goes as

$$(C = U_T \dots U_1, \mathbf{x}) \rightarrow \left(-A, |j\rangle = |\text{step}_0\rangle |\mathbf{x}\rangle |0\rangle^{\otimes r-n}, m = (2T+1)^3, g = 0, \varepsilon = \frac{1}{4(2T+1)} \right). \quad (20)$$

This can be computed in polynomial time, and, according to our previous arguments, it is correct, i.e., it maps positive (negative) instances of the first problem to positive (negative) instances of the second one. ■

We kept the presentation of the proof general up to Eq. (17), which reads for any function f that

$$[f_m(A)]_{j,j} = \frac{1}{M} \left[f_m(1) + 2 \sum_{l=1}^{\frac{M-1}{2}} f_m(\theta_l^+) \right] - \frac{|\alpha_{x,1}|^2}{M} \left[f_m(1) - f_m(-1) + 2 \sum_{l=1}^{\frac{M-1}{2}} (f_m(\theta_l^+) - f_m(-\theta_l^+)) \right]. \quad (21)$$

The first term is independent of the output of the quantum circuit being simulated and can be evaluated exactly efficiently. We see that the proof strategy for hardness we presented above actually shows that entry estimation of any family of matrix functions f_m is BQP-HARD for real-symmetric matrices A ; $\|A\|_1 \leq 2$ that are 5-local, 4-sparse, for inverse error $1/\varepsilon = \mathcal{O}(1/k)$, if the condition

$$\frac{1}{M} \left| f_m^o(\theta_0^+) + 2 \sum_{l=1}^{\frac{M-1}{2}} f_m^o(\theta_l^+) \right| \geq k; \\ \theta_\ell^+ = \cos\left(\frac{2\pi\ell}{M}\right) \quad \forall \ell \in \left[\frac{m-1}{2}\right] \quad (22)$$

is satisfied for some m , where f_m^o denotes the odd contribution of f_m , and for any $M = \mathcal{O}(\text{poly log}(N))$

(the even contribution cancels). Roughly, this inequality states that for entry estimation to be BQP-HARD for f_m it is sufficient that f_m^o varies fast enough in some subinterval of $[-1, 1]$. If it is not true then each term $f_m^o(\cos(2\pi\ell/M))$ with $0 \leq \ell \leq (M-1)/4$ would cancel out its “almost” opposite term $f_m^o(\cos(2\pi((M-1)/2 - \ell)/M)) = f_m^o(-\cos(\pi - 2\pi((M-1)/2 - \ell)/M)) = -f_m^o(\cos(2\pi(\ell+1)/M))$. The same condition applies to the Pauli access model, as we will see in Proposition 1.

We remark that while quantum algorithms can also find off-diagonal matrix entries (for instance, use Lemma A2 in Appendix A to write off-diagonal entries as a linear combination of diagonal entries in some other basis), and our classical algorithms will also be able to manage general matrix entries, the above proof shows hardness even for the restricted problem of computing a diagonal entry. Moreover, it turns out the same hardness result can

be also shown when restricting the problem to strictly off-diagonal entries, and this idea was also shown in Ref. [1]. Similar tricks can be used in the forthcoming BQP-completeness results for Chebyshev polynomials and the inverse function.

Remark 1 (Hardness for off-diagonal entries). The variant of the problem MONOMIAL that computes a strictly off-diagonal matrix entry $[A^m]_{ij}$ for $i \neq j$ can also be shown to be BQP-complete. Hardness can be shown by tensoring the matrix in (15) with an idempotent matrix $B = A \otimes \begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$ which satisfies the property $B^m = A^m \otimes \begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$. Thus, diagonal entries of A^m are encoded into off-diagonal entries of B^m .

The arguments from the proof of Theorem 1 can be adapted to also work for the Pauli query access model.

Proposition 1. The problem $\text{MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ is BQP-COMPLETE. As in Theorem 1, the hardness results holds even if A is 5-local, $\|A\|_1 \leq 2$, and for real symmetric A .

Proof. The BQP membership of $\text{MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ again follows from a direct application of Lemma 4 alongside Lemma 1, noting that Lemma 1 is also applicable for Pauli access when $L, \lambda = \mathcal{O}(\text{poly log } (N))$.

The BQP-hardness is shown by providing Pauli-sparse query access to the matrix $W = \sum_{\ell=0}^{M-1} \mathcal{T}_\ell \otimes V_\ell$ from the proof of Theorem 1 (from which one can build the Pauli access to $A = (W + W^\dagger)/2$). By Lemma A11, we know that each term $\mathcal{T}_\ell$ can be written with $\mathcal{O}(1)$ Pauli terms and Pauli norm 1. Also, by Lemma A9 each V_ℓ can be decomposed in $\mathcal{O}(1)$ Pauli matrices with Pauli norm $\mathcal{O}(1)$. Finally, by Lemma A10 we conclude that each term $\mathcal{T}_\ell \otimes V_\ell$ can be written with $\mathcal{O}(1)$ Pauli terms and $\mathcal{O}(1)$ Pauli norm, and thus the Pauli decomposition of A has norm $\lambda_A = \mathcal{O}(M)$ and $L = \mathcal{O}(M)$ terms, which are $\mathcal{O}(\text{poly log } (N))$ for polynomial-sized circuits. ■

We also obtain an analogous result for the local measurement version of the problem.

Proposition 2. The problems $\text{LM-MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ and $\text{LM-MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ are BQP-COMPLETE. The hardness results hold even if the matrix A is 5-local, 4-sparse, satisfies $\|A\|_1 \leq 2$, and is real symmetric A .

Proof. To show inclusion for both access models we use a two-stage algorithm. First, we use Lemma 4 to evaluate $\|A^m|0\rangle\|^2 = |\langle 0|A^{2m}|0\rangle|$ to additive error $\varepsilon/3$. If this value is $\leq g + \varepsilon/2$ then we output NO as we can guarantee that $\langle 0|A^m \pi A^m|0\rangle \leq g + 5\varepsilon/6$ by Hölder's tracial matrix inequality. If the value is otherwise $> g + \varepsilon/2$

then we can guarantee that $\|A^m|0\rangle\|^2 > g + \varepsilon/6 = \Omega(\varepsilon)$ and we use Lemma 5 to evaluate $\langle 0|A^m \pi A^m|0\rangle / \|A^m|0\rangle\|^2$ to additive error $\varepsilon/3$, with runtime $\mathcal{O}(m/\varepsilon \|A^m|0\rangle\|^2) = \mathcal{O}(m/\varepsilon^2) = \mathcal{O}(\text{poly log } (N))$, where we have used the fact that the Lipschitz constant of x^m is m . Multiplying the two outputs (additive estimates for $\|A^m|0\rangle\|^2$ and $\langle 0|A^m \pi A^m|0\rangle / \|A^m|0\rangle\|^2$) together gives the desired quantity to additive error $2\varepsilon/3 + \varepsilon^2/3 \leq \varepsilon$ (assuming $\varepsilon \leq 1$, else we can rescale the errors appropriately).

To show hardness, we will simulate any circuit with T gates on r qubits $C = U_T \cdots U_1$. As in previous proofs, we assume this circuit is built wholly from Hadamard and Toffoli gates. Recall we would like to approximate $|\alpha_{x,1}|^2$ in Eq. (5). We will consider the measurement $\pi = |1\rangle\langle 1| \otimes \mathbb{1}$ for convenience, which is an arbitrary choice by adding a final gate in the circuit. Consider the sequence of unitaries $C' = V_M \cdots V_1 = U_1^\dagger \cdots U_T^\dagger C_{\text{NOT}} \mathbb{1} \cdots \mathbb{1} C_{\text{NOT}} U_T \cdots U_1$, where we have padded the sequence with $(T+1)$ identities and two CNOT gates, so that $M = 3(T+1)$. C' acts on $r+1$ qubits, with $U_T \cdots U_1$ acting on the lower r registers, and C_{NOT} acting on the first two registers, controlled on the second one. We note that $C_{\text{NOT}}(|1\rangle\langle 1| \otimes \mathbb{1})C_{\text{NOT}} = |10\rangle\langle 10| + |01\rangle\langle 01|$. Given the first register being in the zero state, the C_{NOT} gate toggles the measurement to be on/off on the second register. This, in turn, implies that for any input state $|\mathbf{x}\rangle \in \mathbb{C}^{2^r}$, we have

$$\begin{aligned} & \langle 0, \mathbf{x} | V_1^\dagger \cdots V_\ell^\dagger \pi^{(r+1)} V_\ell \cdots V_1 | 0, \mathbf{x} \rangle \\ &= \begin{cases} \langle \mathbf{x} | C \pi^{(r)} C | \mathbf{x} \rangle = |\alpha_{x,1}|^2 & \text{if } T+1 \leq \ell \leq 2T+2, \\ 0 & \text{otherwise,} \end{cases} \end{aligned} \quad (23)$$

where we denote $\pi^{(m)} = |1\rangle\langle 1| \otimes \mathbb{1}^{(m)}$ where $\mathbb{1}^{(m)}$ is the m -qubit identity matrix

Let us now consider the operator

$$A = \frac{1}{2} \sum_{\ell=0}^{M-1} \left(\mathcal{T}_\ell \otimes V_{\ell+1} + \mathcal{T}_\ell^\dagger \otimes V_{\ell+1}^\dagger \right). \quad (24)$$

Similar to Eq. (15), we note that A in Eq. (24) is also a sparse matrix with sparsity $s = 4 = \mathcal{O}(1)$ (the gates we simulate have sparsity at most 2) and 5-local. When successive powers of A are applied to the initial state $|\text{step}_0\rangle \otimes |0, \mathbf{x}\rangle$, where the first register is the “clock,” a classical random walk is performed over the following M quantum states:

$$|\text{step}_\ell\rangle \otimes V^{(\ell)}|0, \mathbf{x}\rangle = \begin{cases} |\text{step}_\ell\rangle \otimes |0, \mathbf{x}\rangle & \text{for } \ell = 0 \\ |\text{step}_\ell\rangle \otimes U_\ell \cdots U_1|0, \mathbf{x}\rangle & \text{for } 1 \leq \ell \leq T \\ |\text{step}_\ell\rangle \otimes C_{\text{NOT}} U_T \cdots U_1|0, \mathbf{x}\rangle & \text{for } T+1 \leq \ell \leq 2T+2 \\ |\text{step}_\ell\rangle \otimes U_{M-\ell} \cdots U_1|0, \mathbf{x}\rangle & \text{for } 2T+3 \leq \ell \leq M, \end{cases} \quad (25)$$

where we have denoted $V^{(\ell)} = V_\ell \cdots V_1$. We stress from this equation that the state in the latter $(n+1)$ registers is wholly determined by the state of the clock register; it is agnostic to the path taken.

From Eq. (23) we see that only amplitudes corresponding to $T+1 \leq \ell \leq 2T+2$ contribute a nonzero measurement probability, and for the choice of input state $|\mathbf{x}\rangle = |0\rangle^{\otimes r}$ the measurement probability is exactly $|\alpha_{0,1}|^2$ as desired. Considering $\mathbf{x} = \mathbf{0}$ is sufficient since any other input state can be prepared with r additional gates, which we absorb into our definition of C . All that remains is to evaluate the amplitude corresponding to $T+1 \leq \ell \leq 2T+2$ for a given value of matrix power t .

Random walks on a 1D chain are well studied and known to be rapidly mixing. The random walk in question is represented by the M -component probability distribution $\mathbf{p}_m$ (here, the power m of A labels the random-walk iteration), which approaches the uniform distribution $\mathbf{u} = \{\frac{1}{M}, \dots, \frac{1}{M}\}$ as

$$\|\mathbf{p}_m - \mathbf{u}\|_1 \leq \frac{1}{2} \exp\left(-\frac{\pi^2}{2} \frac{m}{M^2}\right), \quad (26)$$

for any iteration $m \geq M^2 \geq 49$ (e.g., see [83, Theorem 2.3]). Explicit evaluation thus gives

$$\langle 0|A^m \pi^{(r)} A^m|0\rangle = |\alpha_{0,1}|^2 \sum_{T+1 \leq \ell \leq 2T+2} p_m^2(\ell). \quad (27)$$

For $p_\infty = \mathbf{u}$, the last sum equals $(T+2)/M^2 \geq 1/3M$. Our final step will be to show that for finite but large enough m the ratio is still $\Omega(1/M)$, and thus any problem in BQP can be decided by solving $\text{LM-MONOMIAL}_{\|A\|}^{\text{SPAR}}$ for precision $\varepsilon = \mathcal{O}(1/M)$.

For an arbitrary distribution $\mathbf{p}_m$ such that $\|\mathbf{p}_m - \mathbf{u}\|_1 = \varepsilon$, the sum in Eq. (27) is minimized when p_m takes uniform values $\frac{1}{M} - \frac{\varepsilon}{2(T+2)}$ across all $T+1 \leq \ell \leq 2T+2$. Thus, we can bound the sum as

$$\sum_{T+1 \leq \ell \leq 2T+2} p_m^2(\ell) \geq \sum_{T+1 \leq \ell \leq 2T+2} \left(\frac{1}{M} - \frac{\varepsilon}{2(T+2)}\right)^2 \quad (28)$$

$$\geq \frac{(T+2)(2 - \frac{\varepsilon}{T+2})^2}{4M^2} \quad (29)$$

$$> \frac{1}{6M}, \quad (30)$$

where the last inequality is true for any $\varepsilon \leq 1$. From Eq. (26), we see that it is thus sufficient to take $m = \mathcal{O}(M^2)$, which scales as $\mathcal{O}(\text{poly log}(N))$ because so does the target circuit size T . This ensures that one can estimate $|\alpha_{0,1}|^2$ up to constant precision via Eq. (27) by estimating the amplitude $\langle 0|A^m \pi^{(r)} A^m|0\rangle$ up to precision $\varepsilon = \mathcal{O}(1/\text{poly log}(N))$.

Note that the matrix A consists of $\mathcal{O}(M) = \mathcal{O}(\text{poly log}(N))$ nonzero entries and can be instantiated via sparse access efficiently. Moreover, it is Pauli-sparse as computational basis entries and constant-dimension unitaries have efficient Pauli decompositions (Lemmas A9 and A11 respectively), and Pauli norms are multiplicative (Lemma A10). Thus, hardness holds in the Pauli access model as well. ■

We may also ask whether $\text{LM-MONOMIAL}_{\|A\|}^{\text{AMODEL}}$ becomes harder when we consider its “normalized form” (that is the quantity expected in the output is a normalized measurement result of quantum states $A^m|0\rangle/\|A^m|0\rangle\|$). We provide some indication this could be the case via Proposition A1 in Appendix A2: using exactly the same construction, one can show that the normalized problem is BQP-HARD even for *constant* error.

Now let us think about classical algorithms. Theorem 1 and Proposition 2 imply that both problems $\text{MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ and $\text{LM-MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ are BQP-COMPLETE if the inverse precision scales polynomially with the input size and the matrix A satisfies $\|A\| \leq 1$. If we strengthen the second condition then they become classically solvable:

Theorem 2. Let $\eta : \mathbb{N} \rightarrow \mathbb{R}$ and assume that for $A \in \mathbb{C}^{N \times N}$ it holds that $\|A\| \leq 1 - \eta(N)$ [84]. Then, the problem $\text{MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ can be solved classically in time $\mathcal{O}\left((1/\varepsilon)^{-\log(s)/\log(1-\eta)}\right)$ for any value of m . Whenever $\eta = \Omega(1)$ and either s or $\varepsilon^{-1} = \mathcal{O}(1)$ this algorithm runs in polynomial time. $\text{LM-MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ can be solved using similar ideas with polynomially equivalent complexity.

Proof. Observe that if $\|A\| \leq 1 - \eta$ it holds that

$$\begin{aligned} [A^m]_{j,j} &= |\langle j | A^m | j \rangle| = \left| \sum_{\lambda} \lambda^m |\langle j | \lambda \rangle|^2 \right| \leq \sum_{\lambda} |\lambda|^m |\langle j | \lambda \rangle|^2 \\ &\leq \sum_{\lambda} (1 - \eta)^m |\langle j | \lambda \rangle|^2 = (1 - \eta)^m. \end{aligned} \quad (31)$$

Therefore, whenever $m > \log \varepsilon / \log(1 - \eta)$ is the case that 0 is an ε -approximation of $\langle i | A^m | i \rangle$. Meanwhile, if $m \leq \log \varepsilon / \log(1 - \eta)$ we can use the algorithm from Lemma 6 to compute the answer in time $\mathcal{O}(s^m) = \mathcal{O}(s^{\log(\varepsilon)/\log(1-\eta)}) = \mathcal{O}((1/\varepsilon)^{-\log(s)/\log(1-\eta)})$.

Regarding $\langle 0 | A^m \pi A^m | 0 \rangle$, we can use Lemma 6 whenever $m < \log \varepsilon / (2 \log(1 - \eta))$. Otherwise, 0 is a sufficient ε -approximation. ■

We recall that the condition $s = \mathcal{O}(1)$ alone should still yield hard problems for classical algorithms (observe that the BQP-HARD proofs from Theorem 1 and Proposition 2 rely only on 4-sparse matrices). Thus, it is the additional condition on the norm of A which allow for classical algorithms. We note that similar conditions on the norm are applied to construct quantum block-encodings on matrices [4, Theorem 30], [24, Lemma 4.5], though here $\eta = 1/\text{poly} \log(N)$ may be tolerated at a cost of only $\text{poly} \log(N)$ gate overhead.

We can obtain an analogous result for the Pauli access model.

Proposition 3. Assume the matrix $A \in \mathbb{C}^{N \times N}$ satisfies the condition $\|A\| \leq 1 - \eta$. Then, the problem $\text{MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ can be solved classically to precision ε with success probability at least $1 - \delta$ in time complexity

$$\mathcal{O} \left(\frac{\log \varepsilon}{\log(1 - \eta)} \frac{\log N}{\varepsilon^2} \left(\frac{1}{\varepsilon} \right)^{-2 \log(\lambda_A)/\log(1-\eta)} \log \left(\frac{1}{\delta} \right) \right). \quad (32)$$

Whenever $\lambda_A = \mathcal{O}(1)$, $\eta = \Omega(1)$ this algorithm works in polynomial time. Similarly, $\text{LM-MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ can also be solved classically with polynomially equivalent complexity in Pauli access.

Proof. Regarding Problem I, as in the proof of Theorem 2, whenever $m > \log \varepsilon / \log(1 - \eta)$ it is the case that 0 is an ε -approximation of $\langle i | A^m | j \rangle$. If $m \leq \log \varepsilon / \log(1 - \eta)$ use the algorithm from Lemma 8. An analogous reasoning can be applied to the Problem II employing Lemma 9 when m is small. ■

As a matter of completeness, we now briefly list the consequences of direct application of the randomized algorithms in Lemmas 7, 8, and 9. Both problems can be solved

classically for the sparse access model if $\|A\|_1 \leq 1$ using the techniques developed in Ref. [24]. Similarly, for the Pauli case, whenever $\lambda_A \leq 1$ we can employ techniques from Ref. [22].

Proposition 4. The problems $\text{MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ and $\text{LM-MONOMIAL}_{\|A\|}^{\text{SPARSEACCESS}}$ can be solved classically with probability at least $1 - \delta$ in time $\tilde{\mathcal{O}}((sm/\varepsilon^2) \|A\|_1^{2m} \log(1/\delta))$ and $\tilde{\mathcal{O}}((sm/\varepsilon^2) \|A\|_1^{4m} \log(1/\delta))$, respectively. Hence, in particular, $\text{MONOMIAL}_{\|A\|_1}^{\text{SPARSEACCESS}}$ and $\text{LM-MONOMIAL}_{\|A\|_1}^{\text{SPARSEACCESS}}$ can be solved classically in polynomial time.

Proof. For the matrix element problem, consider the algorithm from Lemma 7. In this case $f_m(x) = x^m$ and it holds that $\|f_m(\|A\|_1 x)\|_{l_1} = \|A\|_1^m$. In turn, for the local measurement problem, use the algorithm from Lemma 9 in a similar way. ■

Proposition 5. The problems $\text{MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ and $\text{LM-MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ can be solved classically in time $\mathcal{O}(m \log(N) (\lambda_A^{2m}/\varepsilon^2) \log(1/\delta))$ and $\mathcal{O}(m \log(N) \lambda_A^{4m}/\varepsilon^2 \log(1/\delta))$, respectively, both with success probability at least $1 - \delta$. In particular, $\text{MONOMIAL}_{\lambda_A}^{\text{PAULIACCESS}}$ and $\text{LM-MONOMIAL}_{\lambda_A}^{\text{PAULIACCESS}}$ can be solved classically in polynomial time.

Proof. Use the algorithms from Lemmas 8 and 9. ■

Finally, we introduce a notion of *super sparsity*, under which we can solve both problems in both access models. Roughly, a super-sparse matrix in the sparse access model has $\mathcal{O}(\text{poly} \log(N))$ nonzero entries, while a super-sparse Pauli query access only has $\mathcal{O}(\log \log N)$ nonzero coefficients.

Proposition 6. Consider an access model which lists all k nonzero entries in the computational basis via triples $\{a_{ij}, i, j\}_{(i,j) \in S}$ such that $|S| = k$ and $A = \sum_{(i,j) \in S} a_{ij} |i\rangle \langle j|$. We say that we have classical SUPER-SPARSE access to A when $k = \text{poly} \log(N)$. Under this access, both Problems I and II can be solved exactly for a monomial of power m in time complexity $\mathcal{O}(m k^3)$. Thus, $\text{MONOMIAL}_{\|A\|}^{\text{SUPER-SPARSE}}$ and $\text{LM-MONOMIAL}_{\|A\|}^{\text{SUPER-SPARSE}}$ can be solved in $\text{poly} \log(N)$ time.

Proof. For any $d \in \mathbb{N}$, A^d contains at most k^2 different projectors of the form $|i_{\ell_1}\rangle \langle j_{\ell_2}|$, with $\ell_1, \ell_2 \in [k]$. Thus, we can directly compute the coefficient $a_{i_{\ell_1} j_{\ell_2}}^{(d+1)}$ associated with each projector $|i_{\ell_1}\rangle \langle j_{\ell_2}|$ of A^{d+1} given the coefficients for

A^d , as

$$a_{i_{\ell_1} j_{\ell_2}}^{(d+1)} = \sum_{i'_\ell} a_{i_{\ell_1} i'_\ell} a_{i'_\ell j_{\ell_2}}^{(d)}. \quad (33)$$

Hence, A^m can be computed with $\mathcal{O}(mk^3)$ elementary operations. Finally, with the resulting explicit description of A^m both problems can be easily solved. ■

Theorem 3. There is an exact classical algorithm that solves $\text{MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ in time $\mathcal{O}(mL2^L \log N)$. Similarly, there is a similar algorithm that solves $\text{LM-MONOMIAL}_{\|A\|}^{\text{PAULIACCESS}}$ in time complexity $\mathcal{O}((mL2^L + 2^{2L}) \log N)$. In particular, both algorithms run in time $\text{poly log}(N)$ whenever A is super Pauli-sparse, i.e., whenever $L = \mathcal{O}(\log \log(N))$.

Proof. Given $A = \sum_{\ell=1}^L a_\ell P_{i_\ell}$, let $\mathcal{G} = \{P_{i_\ell}\}_{\ell \in [L]}$ and $\langle \mathcal{G} \rangle$ be the Pauli subgroup generated by $\mathcal{G}$ without considering global phases. Then, by Lemma A6 it holds that $|\langle \mathcal{G} \rangle| \leq 2^{L+1}$. Thus, the Pauli decomposition of A^k involves at most 2^{L+1} terms, for any $k \in \mathbb{N}$, and we can compute A^m in a bottom-up manner as follows: Given A^k we can compute $A^{k+1} = AA^k$ by computing all the $\mathcal{O}(L2^L)$ products between the nonzero terms of A and nonzero terms of A^k (each product taking $\log N$ time to evaluate). To compute A^m we need to perform this operation m times, and each step costs $\mathcal{O}(L2^L \log N)$.

Finally, given the explicit Pauli representation of A^m , computing $\langle i|A^m|j \rangle$ is straightforward by explicit sparse-matrix multiplication and takes time $\mathcal{O}(2^L \log N)$. Meanwhile, to compute $\langle 0|A^m \pi A^m|0 \rangle$, one can construct the A^m and compute each term: given Pauli decomposition $A^m = \sum_{\ell=1}^R a_\ell P_{i_\ell}$, it follows that

$$\langle 0|A^m \pi A^m|0 \rangle = \sum_{\ell_1, \ell_2=0}^R a_{\ell_1} a_{\ell_2} \langle 0|P_{i_{\ell_1}} \pi P_{i_{\ell_2}}|0 \rangle. \quad (34)$$

There are $R^2 \leq 2^{2L}$ terms in this sum, and each one can be computed in time $\mathcal{O}(\log N)$. ■

In Observation 2 we remark that Theorem 3 automatically allows exact and efficient computation of any polynomial (such as Chebyshev polynomials) whenever $L = \mathcal{O}(\log \log N)$.

B. Chebyshev polynomials

We consider the following problems:

Problem: $\text{CHEBYSHEV}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with norm $b(A) \leq 1$ and accessible through AMODEL, a positive integer m , index $j \in [N]$, a precision ε and a threshold g , such that $m, 1/\varepsilon, g = \mathcal{O}(\text{poly log}(N))$.

Output: YES if $[T_m(A)]_{j,j} \geq g + \varepsilon$. NO if $[T_m(A)]_{j,j} \leq g - \varepsilon$.

Problem: $\text{LM-CHEBYSHEV}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with norm $b(A) \leq 1$ and accessible through AMODEL, a positive integer m , a precision ε and a threshold g , such that $m, 1/\varepsilon, g = \mathcal{O}(\text{poly log}(N))$.

Output: Let $\pi = |0\rangle\langle 0| \otimes \mathbb{1}_{N/2}$ and $r = \langle 0|T_m(A)\pi T_m(A)|0 \rangle$. Then, answer YES if $r \geq g + \varepsilon$ and NO if $r \leq g - \varepsilon$.

The problems can be solved through the phase estimation-based algorithm from Lemma 4. Moreover, we can prove BQP-completeness when the operator norm is used as the bound condition.

Theorem 4. The problems $\text{CHEBYSHEV}_{\|A\|}^{\text{SPARSEACCESS}}$ and $\text{CHEBYSHEV}_{\|A\|}^{\text{PAULIACCESS}}$ are BQP-COMPLETE. The hardness results hold even under the stronger hypothesis of constant precision $\varepsilon = \Omega(1)$, and the matrix A satisfying $\|A\|_1 \leq 2$, being 5-local, 4-sparse, and real symmetric.

Proof. First, we show that these two problems are in BQP. By Lemmas 4 and 1 we are able to compute $[T_m(A)]_{j,j}$ efficiently if we can bound $\|T_m\|_\infty^{[-1,1]}$ and its Lipschitz constant K_{T_m} over $[-1,1]$. Clearly $\|T_m\|_\infty^{[-1,1]} = 1$, and we can bound K_{T_m} using the Mean Value Theorem as

$$\begin{aligned} |T_m(x) - T_m(y)| & \leq |T'_m(c)| \cdot |x - y| = m|U_{m-1}(c)| \cdot |x - y| \\ & \leq m(m+1)|x - y|, \end{aligned} \quad (35)$$

where we have used properties of Chebyshev polynomials detailed in Definition A1.

Now we provide the BQP-hardness proofs. Consider the matrix $A = (W + W^\dagger)/2$ from the hardness result of Theorem 1, which is 4-sparse, 5-local, Hermitian (real symmetric if we use the Hadamard + Toffoli gate set) and has operator norm bounded by 1. To prove hardness for constant precision it is enough to find a value for m such that T_m is odd and Eq. (22) is satisfied for $k = \Omega(1)$. From now on, we use the notation from the proof of Theorem 1.

Consider $m = M$. Then, T_m is an odd function ($M = 2T + 1$ is odd, and the ℓ th Chebyshev polynomial is odd if $\ell \in \mathbb{N}$ is odd), and since $T_m(\cos(\pi j/m)) = (-1)^j$ for any $j \in \mathbb{N}$ it holds that $T_m(\theta_\ell^+) = 1$, and consequently,

$$\frac{1}{M} \left(T_m(1) + 2 \sum_{l=1}^{\frac{M-1}{2}} T_m(\theta_\ell^+) \right) = 1. \quad (36)$$

It follows that we can distinguish between acceptance and rejection by picking $\varepsilon = \frac{1}{3}$. The precise Karp mapping we are considering is

$$(C = U_T \dots U_1, |\mathbf{x}\rangle) \rightarrow \left(-A, m = 2T + 1, \quad |j\rangle = |\text{step}_0\rangle |\mathbf{x}\rangle |0\rangle^{\otimes r-n}, \quad g = 0, \quad \varepsilon = \frac{1}{6} \right).$$

The matrix constructed in the reduction is the same as the one from Theorem 1, which is Pauli-sparse. Thus, from the same reasoning, the Pauli access version of the problem is also BQP-HARD. ■

We get equivalent results for the local measurement version of these problems.

Proposition 7. The problems $\text{LM-CHEBYSHEV}_{\|A\|}^{\text{SPARSEACCESS}}$ and $\text{LM-CHEBYSHEV}_{\|A\|}^{\text{PAULIACCESS}}$ are BQP-COMPLETE, even under the hypothesis of constant precision $\varepsilon = \Omega(1)$, $\|A\|_1 \leq 2$, and for real symmetric, 5-local 4-sparse A .

Proof. To show inclusion we use a two-stage algorithm in the same spirit of the proof of Proposition 2. First, we use Lemma 4 to evaluate $\|T_m(A)|0\rangle\|^2$ to additive error $\varepsilon/3$. If this value is $\leq g + \varepsilon/2$ then we output NO as we can guarantee that $\langle 0|T_m(A)\pi T_m(A)|0\rangle \leq g + 5\varepsilon/6$ by Hölder’s tracial matrix inequality. If the value is $> g + \varepsilon/2$ then we can guarantee that $\|T_m(A)|0\rangle\|^2 \geq g + \varepsilon/6 = \Omega(\varepsilon)$ and we use Lemma 5 to evaluate $\langle 0|T_m(A)\pi T_m(A)|0\rangle$ to additive error $\varepsilon/3$, with runtime $\mathcal{O}(\text{poly}(1/\varepsilon\|T_m(A)|0\rangle\|^2, m)) = \mathcal{O}(\text{poly}(1/\varepsilon^2, m)) = \mathcal{O}(\text{poly} \log(N))$, where we have used the fact that the Lipschitz constant of T^m is $\mathcal{O}(m^2)$ (see proof of Theorem 4). Multiplying the two outputs together gives the desired quantity to additive error $2\varepsilon/3 + \varepsilon^2/3 \leq \varepsilon$ (assuming $\varepsilon \leq 1$, else we can rescale the errors appropriately).

To show hardness, we use the following property of Chebyshev polynomials:

$$T_m\left(\frac{x+x^{-1}}{2}\right) = \frac{x^m + x^{-m}}{2}. \quad (37)$$

This conveys the “ballistic” property of walks performed by Chebyshev operators (see Ref. [30] for further discussion).

As in the proof of Proposition 2 for monomials we adopt a walk operator of the form

$$A = \frac{1}{2}(W + W^\dagger), \quad (38)$$

with $W = \sum_{\ell=0}^{M-1} \mathcal{T}_\ell \otimes V_{\ell+1}$, where we define the $(r+1)$ -qubit circuit $V_M \dots V_1 = U_1^\dagger \dots U_T^\dagger C_{\text{NOT}} C_{\text{NOT}} U_T \dots U_1$ where now it will not be necessary to pad the sequence with identities. Recall that $U_T \dots U_1$ is the r -qubit circuit we are simulating, and similar to the proof of the monomials we presume that the input state to the BQP problem $|\mathbf{x}\rangle$ is encoded in the first gates of the circuit, so we can take $|0\rangle$ as input. Recall also that C_{NOT} denotes a CNOT gate targeted on an otherwise untouched ancillary register which we place in the first nonclock register—this will be the register we measure. Thus, we have $M = 2T + 2$ clock register states to simulate a T -gate circuit $U_T \dots U_1$. It can be checked that $W^\dagger = W^{-1}$ and so Eq. (37) implies that $T_m(\frac{1}{2}(W + W^\dagger)) = \frac{1}{2}(W^m + (W^\dagger)^m)$. Similar to before we explicitly write the orbit of states for successive powers of W or $W^\dagger$, for any r -qubit input state $|\phi\rangle$,

$$|\text{step}_\ell\rangle \otimes V^{(\ell)}|0, \phi\rangle = \begin{cases} |\text{step}_0\rangle \otimes |0, \phi\rangle & \text{for } \ell = 0, \\ |\text{step}_\ell\rangle \otimes U_\ell \dots U_1|0, \phi\rangle & \text{for } 1 \leq \ell \leq T, \\ |\text{step}_\ell\rangle \otimes C_{\text{NOT}} U_T \dots U_1|0, \phi\rangle & \text{for } \ell = T + 1, \\ |\text{step}_\ell\rangle \otimes U_{M-\ell} \dots U_1|0, \phi\rangle & \text{for } T + 2 \leq \ell \leq M - 1. \end{cases} \quad (39)$$

We can check that

$$\begin{aligned} W^{T+1}|\text{step}_0\rangle|0, \phi\rangle &= (W^\dagger)^{T+1}|\text{step}_0\rangle|0, \phi\rangle \\ &= |\text{step}_\ell\rangle \otimes C_{\text{NOT}} U_T \dots U_1|0, \phi\rangle. \end{aligned} \quad (40)$$

Using Eq. (37) with the choice $m = T + 1$ we have

$$T_{T+1}(A)|\text{step}_0\rangle|0, \phi\rangle = \frac{1}{2}(W^{T+1} + (W^\dagger)^{T+1})|\text{step}_0\rangle|0, \phi\rangle \quad (41)$$

$$= |\text{step}_\ell\rangle \otimes C_{\text{NOT}} U_T \dots U_1|0, \phi\rangle, \quad (42)$$

which is a normalized state for which we know that measurement of the first (nonclock) register yields

$$\begin{aligned} \langle 0, \mathbf{0} | U_1^\dagger \cdots U_T^\dagger C_{\text{NOT}} \pi^{(r+1)} C_{\text{NOT}} U_T \cdots U_1 | 0, \mathbf{0} \rangle \\ = \langle \mathbf{0} | C \pi^{(r)} C | \mathbf{0} \rangle = |\alpha_{0,1}|^2, \end{aligned} \quad (43)$$

where we have denoted $\pi^{(m)} = |1\rangle\langle 1| \otimes \mathbb{1}^{\otimes(m-1)}$ and set $|\phi\rangle = |\mathbf{0}\rangle$. Thus, any BQP problem with T gates can be reduced to the Chebyshev local measurement problem with constant error and $(T+1)$ th Chebyshev polynomial. As Eq. (42) is already normalized, this demonstrates hardness both for the normalized and unnormalized problems, assuming efficient access.

Finally, we can check efficient access: the walk operator A in Eq. (38) is 4-sparse and 5-local and lends itself to efficient sparse access. It also has a Pauli decomposition of $\text{poly}(M)$ Pauli operators, with $\mathcal{O}(M)$ Pauli norm (see Lemmas A11, A8, and A10). ■

We cannot formally prove BQP-completeness for the case when $\|A\|_1 \leq 1$ nor $\lambda_A \leq 1$, but still can argue that these problems should not be solvable efficiently classically, as this would imply that $\text{BPP} = \text{BQP}$.

Proposition 8. Suppose there is a classical probabilistic algorithm $\mathcal{A}$ that, given sparse (or Pauli) access to a matrix $A \in \mathbb{C}^{N \times N}$ satisfying $\|A\|_1 \leq 1$ (or $\lambda_A \leq 1$), two indices $i, j \in [N]$, an integer m , a precision ε satisfying $m, 1/\varepsilon = \mathcal{O}(\text{poly} \log(N))$ and a number $\delta \in (0, 1)$; outputs with probability at least $1 - \delta$ an ε -approximation of $\langle i | T_m | j \rangle$ in time $\mathcal{O}(\text{poly}(\log N, m, 1/\varepsilon))$. If so, then $\text{BPP} = \text{BQP}$.

The result holds even if the algorithm $\mathcal{A}$ can only work on matrices satisfying $\|A\|_1 = \mathcal{O}(1/\text{poly} \log(N))$ or $\lambda_A = \mathcal{O}(1/\text{poly} \log(N))$.

Proof. We describe the algorithm for the sparse access case, but the treatment of the Pauli access model is equivalent. Consider the problem of computing, given a matrix A through sparse access with norm condition $\|A\|_1 \leq 1$, a ε -approximation of $\langle i | e^{iAm} | j \rangle$, which is BQP-COMplete for choice of $m, 1/\varepsilon = \mathcal{O}(\text{poly} \log(N))$ (see Proposition 13) [85]. We are going to show how to solve it efficiently classically using $\mathcal{A}$.

To ε -approximate $\langle i | e^{iAm} | j \rangle$ we can consider a $\varepsilon/2$ -approximation of the function $f(x) = e^{ixm}$ given by the Anger-Jacobi expansions (see Lemma 3). It holds that

$$\begin{aligned} \langle i | e^{iAm} | j \rangle \approx \langle i | J_0(m) \mathbb{1} | j \rangle + 2 \sum_{k=1}^R (-1)^k J_{2k}(m) T_{2k}(A) \\ + 2i \sum_{k=0}^R (-1)^k J_{2k+1}(m) T_{2k+1}(A) | j \rangle \end{aligned}$$

$$\begin{aligned} = \langle i | J_0(m) \mathbb{1} | j \rangle + 2 \sum_{k=1}^R (-1)^k J_{2k}(m) \langle i | T_{2k}(A) | j \rangle \\ + 2i \sum_{k=0}^R (-1)^k J_{2k+1}(m) \langle i | T_{2k+1}(A) | j \rangle, \end{aligned}$$

where $R = \mathcal{O}(m + \log(2/\varepsilon))$ and J_j is the Bessel function of the first kind of order j . To approximate this expression within error $\varepsilon/2$ it is enough to approximate each term $J_k(m) \langle i | T_k(A) | j \rangle$ with precision $\varepsilon/8R$ and $J_0(m)$ with precision $\varepsilon/4$.

Note that $|J_k(m)| \leq 1$ in general. In addition, since both $\|A\| \leq \|A\|_1$ and $\|A\| \leq \lambda_A$ hold, the proposition's assumptions imply that $|\langle i | T_k(A) | j \rangle| \leq 1$. Hence, we can $\varepsilon/8R$ -approximate $J_k(m) \langle i | T_k(A) | j \rangle$ using a $\varepsilon/24R$ approximation of both $J_k(m)$ and $\langle i | T_k(A) | j \rangle$ [86]. Thanks to the fact that $R = \mathcal{O}(m + \log(2/\varepsilon))$, the precision $\varepsilon/24R$ is $\mathcal{O}(1/\text{poly}(m\varepsilon))$ and we can use $\mathcal{A}$ to compute in polynomial time the $\varepsilon/24R$ -approximation of $\langle i | T_k(A) | j \rangle$, while for the term $J_k(m)$ we employ folklore techniques. Finally, by picking the probability of success as $\delta = \mathcal{O}(1/R)$ and using the Union Bound we ensure that with constant probability all approximations are correct.

The second statement of the Proposition follows straightforwardly by observing that $\langle i | e^{iAm} | j \rangle = \langle i | e^{i(A/b)bm} | j \rangle$ for any $b \in \mathbb{R} \setminus \{0\}$ and applying the same strategy as before. ■

We do not have an explicit analogous result to Proposition 8 for local measurements. However, we next provide a simple intuition that strongly suggests that the local measurement problem is expected to be at least as hard as entry estimation for Chebyshev polynomials.

Remark 2 (Connection between local measurement and entry estimation). Given the ability to compute the local measurement result for both outcomes “0” and “1” for the Chebyshev polynomial $T_m(A)$ of a Hermitian matrix A , one can compute the entry estimation problem for $T_{2m}(A)$. Thus, entry estimation of every even Chebyshev polynomial can be simulated by local measurement, if one has access to both measurement outcomes.

This can be seen from the simple identities $\langle 0 | T_m(A)^2 | 0 \rangle = \langle 0 | T_m(A) (|0\rangle\langle 0| \otimes I) T_m(A) | 0 \rangle + \langle 0 | T_m(A) (|1\rangle\langle 1| \otimes I) T_m(A) | 0 \rangle$ and $T_m(A)^2 = \frac{1}{2}(1 + T_{2m}(A))$. This argument shows that for Chebyshev polynomials we should expect the local measurement problem to be at least as hard as the entry estimation problem. This is because one naturally assumes the hardness of recovering the “1” measurement result to be no harder than that for “0.” Recall that the problem estimates the “0” outcome probability on a non-normalized state, otherwise the assumption would be trivial. Note that a remark analogous to Remark 2 can be written for monomials,

using the fact that $\langle 0|A^{2m}|0\rangle = \langle 0|A^m(|0\rangle\langle 0| \otimes \mathbb{1})A^m|0\rangle + \langle 0|A^m(|1\rangle\langle 1| \otimes \mathbb{1})A^m|0\rangle$, and similarly for the real exponential function.

Theorem 4 and Propositions 7 and 8 show that the problem of computing matrix Chebyshev polynomials is harder than the problem of computing monomials. In particular, $\text{CHEBYSHEV}_{\|A\|}^{\text{SPARSEACCESS}}$ remains BQP-COMPLETE under the restriction that $\varepsilon = \Omega(1)$, and we can argue that the problem cannot be solved classically even when considering $\|A\|_1 \leq 1$ or $\|A\| \leq 1 - \eta$ [87]. Moreover, Theorem 8 shows that under the Pauli representation computing matrix Chebyshev polynomials is hard classically, even when considering that the Pauli norm is bounded by 1.

Nonetheless, we identify a classically tractable case when the requested Chebyshev polynomial has a sufficiently low degree:

Proposition 9. The problem $\text{CHEBYSHEV}_{\|A\|}^{\text{SPARSEACCESS}}$ can be solved exactly in time $\mathcal{O}(ms^m)$, which is polynomial time whenever $s = \mathcal{O}(1)$, $m = \mathcal{O}(\log \log N)$. It can also be solved approximately classically with probability at least $(1 - \delta)$ in time

$$\mathcal{O}\left(m^3 s \frac{4^{2m}}{\varepsilon^2} \max(1, \|A\|_1^{2m}) \log\left(\frac{1}{\delta}\right)\right), \quad (44)$$

which is polynomial time if $\|A\|_1 = \mathcal{O}(1)$, $m = \mathcal{O}(\log \log N)$. Meanwhile, the problem $\text{CHEBYSHEV}_{\|A\|}^{\text{PAULIACCESS}}$ can be solved with probability at least $(1 - \delta)$ in time

$$\mathcal{O}\left(m^3 \log(N) \frac{4^{2m}}{\varepsilon^2} \max(1, \lambda_A^{2m}) \log\left(\frac{1}{\delta}\right)\right), \quad (45)$$

which is polynomial time whenever $\lambda_A = \mathcal{O}(1)$, $m = \mathcal{O}(\log \log N)$.

Proof. For the case of sparse access, using the algorithm from Lemma 6 we can compute all the monomials $\langle i|A^k|j\rangle$ for $k = 0, \dots, m$ in time $\mathcal{O}(ms^m)$. Then, multiplying by the coefficients of T_m and summing up the results takes additional time $\mathcal{O}(m)$.

For the other two algorithms we use Lemmas 7 and 8. Let us compute the value of $\|T_m(\lambda_A x)\|_{\ell_1}$. Since each coefficient of T_m is upper bounded by 4^m (Lemma A1), it follows that

$$\|T_m(\lambda_A x)\|_{\ell_1} \leq \sum_{k=0}^m 4^m \lambda_A^k = 4^m \sum_{k=0}^m \lambda_A^k, \quad (46)$$

If $\lambda_A = 1$ then $\|T_m(\lambda_A x)\|_{\ell_1} \leq m4^m$. Meanwhile, if $\lambda_A < 1$ we get $\|T_m(\lambda_A x)\|_{\ell_1} = \mathcal{O}(4^m)$. Finally, if $\lambda_A > 1$ we can bound the sum by $m\lambda_A^m$ and we get $\|T_m(\lambda_A x)\|_{\ell_1} \leq m(4\lambda_A)^m$. ■

Proposition 10. The problems $\text{LM-CHEBYSHEV}_{\|A\|}^{\text{SPARSEACCESS}}$ and $\text{LM-CHEBYSHEV}_{\|A\|}^{\text{PAULIACCESS}}$ can be solved in time complexities polynomially equivalent to those for the entry estimation version of the problem in Proposition 9.

Proof. Use The classical sampling algorithm of Lemma 9 and the bounds on $\|T_m(\lambda_A x)\|_{\ell_1}$ from Proposition 9. ■

C. Matrix inversion

We formalize our problems of interest as follows:

Problem: $\text{INVERSE}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with condition number κ_A , norm $b(A) \leq 1$, and accessible through AMODEL, index $j \in [N]$, a precision ε and a threshold g , such that $\kappa_A, 1/\varepsilon, g = \mathcal{O}(\text{poly log}(N))$.

Output: YES if $[A^{-1}]_{jj} \geq g + \varepsilon$ and NO if $[A^{-1}]_{jj} \leq g - \varepsilon$.

Problem: $\text{LM-INVERSE}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with condition number κ_A , norm $b(A) \leq 1$, and accessible through AMODEL, a precision ε and a threshold g , such that $\kappa, 1/\varepsilon, g = \mathcal{O}(\text{poly log}(N))$.

Output: Let $\pi = |0\rangle\langle 0| \otimes \mathbb{1}_{N/2}$ and $r = \langle 0|^{\otimes n} A^{-1} \pi A^{-1} |0\rangle^{\otimes n}$. Then, answer YES if $r \geq g + \varepsilon$ and NO if $r \leq g - \varepsilon$.

We note that our formalization for the inverse function does not coincide with the one from the HHL paper [2] since they consider the normalized version (i.e., computing $\langle 0|A^{-1} \pi A^{-1}|0\rangle / \|A^{-1}|0\rangle\|$).

We prove the hardness of $\text{INVERSE}_{\|A\|_1}^{\text{AMODEL}}$ for both access models employing again the construction from Theorem 1 and Eq. (22). In particular, we obtain the hardness result even for constant precision and under the condition that $\|A\|_1 \leq 1/\text{poly log}(N)$ for the sparse access model and the equivalent one $\lambda_A \leq 1/\text{poly log}(N)$ for the Pauli model.

Theorem 5. The problems $\text{INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$ and $\text{INVERSE}_{\lambda_A}^{\text{PAULIACCESS}}$ are BQP-COMPLETE. Both hardness results hold even under the hypothesis of constant precision $\varepsilon = \Omega(1)$, and for real symmetric, 4-sparse, 5-local A . The results hold also under the more stringent norm conditions $\|A\|_1 \leq 1/\text{poly log}(N)$ and $\lambda_A \leq 1/\text{poly log}(N)$, respectively.

Proofsketch. For both BQP membership, use the algorithm from Lemma 4 considering the function

$$f_{\kappa}(x) = \begin{cases} \frac{1}{x} & \text{for } x \in [-1, 1] \setminus \left[-\frac{1}{\kappa}, \frac{1}{\kappa}\right], \\ 0 & \text{for } x \in \left(-\frac{1}{\kappa}, \frac{1}{\kappa}\right), \end{cases} \quad (47)$$

that satisfies $K_f \leq \kappa^2$ and $\|f\|_{\infty}^{[-1, -1]} \leq \kappa$.

The BQP-HARD proof for $\text{INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$ can be obtained again in a similar fashion to the proof of BQP-hardness of monomials. More precisely, one uses Eq. (17) considering the matrix $A/2$, for A given by Eq. (15), which has the same eigenvalues but divided by two. The proof for the hardness of $\text{INVERSE}_{\lambda_A}^{\text{PAULIACCESS}}$ is identical but dividing A by λ_A . We leave the details of proving hardness to Theorem A2 in Appendix A. ■

We can prove an analogous result for the local measurement versions.

Proposition 11. The problems $\text{LM-}\text{INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$ and $\text{LM-MONOMIAL}_{\lambda_A}^{\text{PAULIACCESS}}$ are BQP-COMPLETE. Both hardness results hold under the hypothesis of constant precision, and for real symmetric, 4-sparse, 5-local A .

Proof. Regarding inclusion in BQP, the normalized version is well known to be in BQP for both access models due to the algorithm from Ref. [2] which employs a Hamiltonian simulation oracle we can instantiate efficiently. The unnormalized version can also be solved employing that algorithm, since we can compute $\langle 0|A^{-1}\pi A^{-1}|0\rangle/\|A^{-1}\| |0\rangle$ with precision $\varepsilon/\|A^{-1}|0\rangle\|^2$ and then multiply this value by $\|A^{-1}|0\rangle\|^2$ (which we can compute using the algorithms developed for matrix power). This will work in polynomial time because $\|A^{-1}|0\rangle\| = \mathcal{O}(\kappa) = \mathcal{O}(\text{poly log}(N))$.

To prove hardness, we employ the construction from Ref. [2] with some small tweaks to ensure the hypothesis condition of real-symmetric matrices. We leave the details to Appendix A, Theorem A1, where we recount the ideas of the proof from Ref. [2] for completeness. ■

The previous results also imply the BQP-completeness of the problem $\text{INVERSE}_{\|A\|}^{\text{SPARSEACCESS}}$: the BQP-hardness follows from a direct reduction from $\text{INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$, while the BQP algorithm is the same one based on Lemma 4.

Theorem 5 establishes that the problem $\text{INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$ is BQP-HARD even for a constant precision level. Moreover, by inspecting the proof it is clear that it will remain hard if we ask $\|A\| \leq 1 - \eta$ for some fixed η , as we did in Theorem 2. Nonetheless, we can obtain a classically solvable restriction if we upper bound κ :

Proposition 12 (Classical algorithm for matrix inversion with suppressed condition number). Let $\beta(\kappa_A, \varepsilon) = 2\kappa_A \log(\kappa_A^2/\varepsilon)$. Then, the problem $\text{INVERSE}_{\|A\|}^{\text{SPARSEACCESS}}$ can be solved exactly classically in time $\mathcal{O}(\beta(\kappa_A, \varepsilon)s^{\beta(\kappa_A, \varepsilon)})$, which is poly log(N) whenever $\kappa_A, s = \mathcal{O}(1)$. It can also be solved approximately classically with probability at

least $1 - \delta$ in time

$$\mathcal{O}\left(\beta(\kappa_A, \varepsilon)^3 s^{\frac{2\beta(\kappa_A, \varepsilon)}{\varepsilon^2}} \max(1, \|A\|_1^{4\beta(\kappa_A, \varepsilon)}) \log\left(\frac{1}{\delta}\right)\right),$$

which is poly log(N) if $\|A\|_1 \leq 1, \kappa_A = \mathcal{O}(1)$.

Meanwhile $\text{INVERSE}_{\|A\|}^{\text{PAULIACCESS}}$ can be solved with probability at least $1 - \delta$ in time

$$\mathcal{O}\left(\beta(\kappa_A, \varepsilon)^3 \log(N) \frac{2\beta(\kappa_A, \varepsilon)}{\varepsilon^2} \max(1, \lambda_A^{4\beta(\kappa_A, \varepsilon)}) \log\left(\frac{1}{\delta}\right)\right),$$

which is poly log(N) whenever $\kappa_A, \lambda_A = \mathcal{O}(1)$.

Finally, Problems $\text{LM-}\text{INVERSE}_{\|A\|}^{\text{SPARSEACCESS}}$ and $\text{LM-}\text{INVERSE}_{\lambda_A}^{\text{PAULIACCESS}}$ can be solved in times polynomially equivalent to the two time complexities just mentioned above.

Proof. We begin with the proof of the statements about the matrix element problem. Given ε is it possible to efficiently build an ε -approximation $P(x)$ of the function $1/x$ in the range $[-1, -(1/\kappa_A)] \cup [1/\kappa_A, 1]$ with degree upper bounded by $2\kappa_A \log(\kappa_A^2/\varepsilon) = \beta(\kappa_A, \varepsilon)$ using Lemma 2. Then, for the sparse access model, we can compute $P(A)$ exactly using the algorithm from Lemma 6 for each monomial. The final complexity is $\mathcal{O}(\beta(\kappa_A, \varepsilon)s^{2\beta(\kappa_A, \varepsilon)})$.

Regarding the two probabilistic algorithms, we can use Lemmas 7 and 8. Let us bound the value $\|P(\lambda_A x)\|_{\ell_1}$: each coefficient from $P(x)$ is upper bounded by $2^{\beta(\kappa_A, \varepsilon)}$, since they are given by the binomial expressions $\binom{\beta(\kappa_A, \varepsilon)}{i}$, and thus

$$\begin{aligned} \|P(\lambda_A x)\|_{\ell_1} &\leq \sum_{k=0}^{2\beta(\kappa_A, \varepsilon)} 2^{\beta(\kappa_A, \varepsilon)} \lambda_A^k \\ &\leq 2\beta(\kappa_A, \varepsilon) 2^{\beta(\kappa_A, \varepsilon)} \max(1, \lambda_A^{2\beta(\kappa_A, \varepsilon)}), \end{aligned} \quad (48)$$

where the last inequality follows through a reasoning analogous to the one from Proposition 8.

Finally, the algorithms for the local measurement version of the problem are obtained using Lemma 9. ■

D. Time evolution

In this section, we show our results for Problems I and II when $f_i(x) = e^{-itx}$. For this function, the former problem can be used to define a promise problem as follows:

Problem: $\text{TIMEEVOLUTION}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with norm $b(A) \leq 1$ and accessible through AMODEL, a positive real number t , integers $j, k \in [N]$, a precision ε and a threshold g , such that $t, 1/\varepsilon, g = \mathcal{O}(\text{poly log}(N))$. b is an upper bound on $\|A\|$, t and $\frac{1}{\varepsilon}$ are poly log(N).

Output: YES if $||[e^{-itA}]_{j,k}|| \geq g + \varepsilon$. NO if $||[e^{-itA}]_{j,k}|| \leq g - \varepsilon$.

Similarly, for Problem II we define:

Problem: $\text{LM-TIMEEVOLUTION}_{b(A)}^{\text{AMODEL}}$

Input: A $N \times N$ Hermitian matrix A with norm $b(A) \leq 1$ and accessible through AMODEL , a positive real number t , a precision ε and a threshold g , such that $t, 1/\varepsilon, g = \mathcal{O}(\text{poly log}(N))$. b is an upper bound on $\|A\|$, t and $\frac{1}{\varepsilon}$ are $\text{poly log}(N)$.

Output: Let $\pi = |0\rangle\langle 0| \otimes \mathbb{1}_{N/2}$ and $r = \langle 0|e^{itA}\pi e^{-itA}|0\rangle$. Then, answer YES if $r \geq g + \varepsilon$ and NO if $r \leq g - \varepsilon$.

We start by proving BQP-completeness of the problem under sparse access for the entry estimation problem. Our proof of BQP-hardness is based on a slight modification of the circuit-to-Hamiltonian mapping proposed by Peres in Ref. [65], one of the first clock constructions for circuit simulation.

Proposition 13. The problems $\text{TIMEEVOLUTION}_{\|A\|_1}^{\text{SPARSEACCESS}}$ and $\text{TIMEEVOLUTION}_{\lambda_A}^{\text{PAULIACCESS}}$ are BQP-COMplete for constant precision $\varepsilon = \Omega(1)$ and a constant choice of row sparsity ($s = 8$) and 5-locality.

Proof. We start by showing BQP membership. First, according to Lemma 1, efficient quantum algorithms to approximate the Hamiltonian simulation operator e^{-itA} as a circuit exist in both access models as long as $t, s, 1/\varepsilon = \text{poly log}(N)$ for constant $\|A\|$ (SPARSEACCESS) or constant λ_A (PAULIACCESS). This sets the stage for inclusion in BQP of $\text{TIMEEVOLUTION}_{\lambda_A}^{\text{PAULIACCESS}}$ and $\text{TIMEEVOLUTION}_{\|A\|_1}^{\text{SPARSEACCESS}}$, by noting that $\|A\| \leq \|A\|_1$. We now have to show that any matrix element $\langle j|e^{-itA}|i\rangle$ can be efficiently estimated. Notice that the real and imaginary parts of the matrix element can be estimated separately using a Hadamard test [88] on the unitary $W_j e^{-itA} W_i$, and initial state $|0^n\rangle$, where $W_{i(j)}$ is a tensor product of bit-flips that transforms the state $|0^n\rangle$ into the computational basis state $|i\rangle$ ($|j\rangle$). This requires a polynomial size circuit and polynomial number of samples, as long as the circuit implementation of e^{-itA} is polynomial and $1/\varepsilon = \text{poly log}(N)$.

To show the problems are BQP-HARD, consider a circuit $C = U_T \dots U_1$ on r qubits and a bitstring $\mathbf{x}$ as the inputs to $\text{BQPCIRCUITSIMULATION}$ (see Observation 1).

From C we can build a new circuit acting on $r' = r + 1$ qubits with corresponding unitary transformation given as $C' = (\mathbb{1} \otimes C^\dagger) (\text{CNOT}_{21} \otimes \mathbb{1}^{\otimes r-1}) (\mathbb{1} \otimes C) := V_\tau \dots V_1$, with $\tau = 2T + 1 = \text{poly log}(N)$. This circuit is the result of acting C on the original qubits, copying the computation result on the original first qubit into the additional ancilla qubit with a CNOT gate, followed by

uncomputation with $C^\dagger$. One can directly verify that

$$\begin{aligned} \langle 0|\langle \mathbf{x}|\langle 0^{r-n}|C'|0\rangle|\mathbf{x}\rangle|0^{r-n}\rangle &= \alpha_{\mathbf{x},0} \quad \text{and} \\ \langle 1|\langle \mathbf{x}|\langle 0^{r-n}|C'|0\rangle|\mathbf{x}\rangle|0^{r-n}\rangle &= \alpha_{\mathbf{x},1}. \end{aligned} \quad (49)$$

In the rest of the proof, we denote $|\mathbf{x}'\rangle = |0\rangle|\mathbf{x}\rangle$, that is the bit string $\mathbf{x}$ padded with a zero bit.

Consider the Hamiltonian working on a unary clock of dimension τ alongside $\mathbf{x}'$ defined as

$$A = \frac{1}{4\tau} \sum_{j=1}^{\tau} \sqrt{j(\tau+1-j)} \left(T_j \otimes V_j + T_j^\dagger \otimes V_j^\dagger \right), \quad (50)$$

which has row sparsity 8 and is 5-local. The 1-norm of A can be bounded as

$$\begin{aligned} \|A\|_1 &= \max_j \frac{\sqrt{j(\tau+1-j)}}{4\tau} 2 \|V_j\|_1 \\ &\leq \max_j \frac{\sqrt{j(\tau+1-j)}}{\tau} \leq \frac{1+\tau}{2\tau} \leq 1, \end{aligned} \quad (51)$$

where we used the fact that $\|V_j\|_1 \leq 2$ for a 2-qubit unitary. Let us consider that the initial state of the evolution is $|\text{step}_0\rangle|\mathbf{x}'\rangle|0\rangle^{\otimes r-n}$. Given an ansatz for the time-evolved state as $|\psi(t)\rangle = e^{-itA}|\text{step}_0\rangle|\mathbf{x}'\rangle|0\rangle^{\otimes r-n} = c_0(t)|\text{step}_0\rangle|\mathbf{x}'\rangle|0\rangle^{\otimes r-n} + \sum_{j=1}^{\tau} c_j(t)|\text{step}_j\rangle \otimes V_j \dots V_1|j\rangle|\mathbf{x}'\rangle|0\rangle^{\otimes r-n}$, the corresponding Schrodinger equation can be solved to find the time-dependent coefficients. In particular, $c_0(t) = (\cos(t/4\tau))^\tau$ while $c_\tau(t) = (i \sin(t/4\tau))^\tau$. Therefore, for $t = 2\pi\tau$ (and consequently $t = \text{poly log}(N)$) it holds that

$$e^{-2\pi\tau iA}|\text{step}_0\rangle|\mathbf{x}'\rangle|0\rangle^{\otimes r-n} = i^\tau |\text{step}_\tau\rangle C'|\mathbf{x}'\rangle|0\rangle^{\otimes r-n}. \quad (52)$$

Thus, with $f_{2\pi\tau}(A) = e^{-2\pi\tau iA}$, we have

$$\begin{aligned} \langle \text{step}_\tau | \langle 1 | \langle \mathbf{x} | \langle 0 |^{\otimes r-n} f_{2\pi\tau}(A) | \text{step}_0 \rangle | 0 \rangle | \mathbf{x} \rangle | 0 \rangle^{\otimes r-n} \\ = \langle \text{step}_\tau | \langle 1 | \langle \mathbf{x} | \langle 0 |^{\otimes r-n} (i^\tau |\text{step}_\tau\rangle C' | 0 \rangle | \mathbf{x} \rangle | 0 \rangle^{\otimes r-n}) \\ = i^\tau \alpha_{\mathbf{x},1}, \end{aligned} \quad (53)$$

where we have used Eq. (49). The Karp mapping follows with $|j\rangle = |\text{step}_0\rangle|0\rangle|\mathbf{x}\rangle|0\rangle^{\otimes r-n}$, $|k\rangle = |\text{step}_\tau\rangle|1\rangle|\mathbf{x}\rangle|0\rangle^{\otimes r-n}$, $g = 1/\sqrt{2}$, and $\varepsilon = (2 - \sqrt{3})/(2\sqrt{6})$. ■

The result for the local measurement problem follows analogously.

Proposition 14. The problems $\text{LM-TIMEEVOLUTION}_{\|A\|_1}^{\text{SPARSEACCESS}}$ and $\text{LM-TIMEEVOLUTION}_{\lambda_A}^{\text{PAULIACCESS}}$ are BQP-COMplete for constant error $\varepsilon = \Omega(1)$ and constant choice of row sparsity ($s = 8$).

Proof. The proof of inclusion for the entry estimation problem holds for local measurement, with the difference that the local observable can be estimated directly from the state transformed by e^{-iAt} without needing the SWAP test.

The proof of hardness also follows straightforwardly from the construction for entry estimation, as can be seen from Eq. (52) with the local measurement being performed in the middle register. ■

Further, we note that the norm condition can be strengthened.

Proposition 15. The problems $\text{TIMEEVOLUTION}_{\|A\|_1}^{\text{SPARSEACCESS}}$, $\text{LM-TIMEEVOLUTION}_{\|A\|_1}^{\text{SPARSEACCESS}}$, $\text{TIMEEVOLUTION}_{\lambda_A}^{\text{PAULIACCESS}}$, $\text{LM-TIMEEVOLUTION}_{\lambda_A}^{\text{PAULIACCESS}}$ are BQP-COMPLETE for constant error $\varepsilon = \Omega(1)$, a constant choice of row sparsity ($s = 8$) and $\|A\|_1, \lambda_A \leq 1/\text{poly} \log(N)$, respectively.

Proof. Inclusion follows from Lemma 1 as before, as such bounded-norm matrices fall within the parameters of Lemma 1. To show hardness, we pick any $c = \Theta(\text{poly} \log(N))$. We normalize the Hamiltonian in Eq. (50) as $A' = A/c$, and we see that this matrix satisfies the norm condition we impose for our problem class. Moreover, simulation of A' for time $c2\pi(2T+1)$ simulates a circuit with T gates, where $c2\pi(2T+1) = \mathcal{O}(\text{poly} \log(N))$ for $T = \mathcal{O}(\text{poly} \log(N))$. ■

We now move on to an efficient classical algorithm for a general class of matrices. We remark that the normalized and unnormalized versions of the local measurement problem are equivalent for time evolution since the function is unitary and, therefore, does not change the state's norm.

Proposition 16. Problems I and II for e^{iAt} are classically easy with Hermitian $A \in \mathbb{C}^{N \times N}$, $\|A\| \leq 1$, for $\|A\|_1 t = \mathcal{O}(\log \log N)$ in the sparse access model and $\lambda_A t = \mathcal{O}(\log \log N)$ in the Pauli access model.

Proof. We follow the quantum algorithm of Ref. [66], but simulate it classically using the algorithms of Lemmas 7 and 8. Let us denote $\hat{t} = \gamma t$, where $\gamma = \|A\|_1$ for the sparse access model and $\gamma = \lambda_A$ for the Pauli access model. Consider a fragmentation of the time evolution operator $e^{iAt} = (e^{iAt/r})^r$. We approximate each fragment of the time evolution via a truncated Taylor series $e^{iAt/r} \approx \sum_{k=0}^K ((iAt/r)^k/k!)$. The truncation error satisfies

$$\left\| e^{iAt} - \left(\sum_{k=0}^K \frac{(iAt/r)^k}{k!} \right)^r \right\| \leq r \left\| e^{iAt/r} - \sum_{k=0}^K \frac{(iAt/r)^k}{k!} \right\|$$

$$\leq \varepsilon, \quad (55)$$

where the first inequality is due to a series of triangle inequalities, and the second inequality is true for

choice of $r = t/\ln 2$, $K = \mathcal{O}((\log t/\varepsilon)/(\log \log t/\varepsilon))$. We note for later that the size of the sum $\sum_{k=0}^K ((\hat{t}/r)^k/k!)$ is upper bounded by $e^{\hat{t}/r}$, and thus the size of the sum $\left(\sum_{k=0}^K ((\hat{t}/r)^k/k!) \right)^r$ is upper bounded by $e^{\hat{t}}$.

We can approximate a matrix entry with a probabilistic distribution over matrix entries of matrix powers

$$\begin{aligned} \langle j | e^{iAt} | m \rangle &\stackrel{\varepsilon}{\approx} \langle j | \left(\sum_{k=0}^K \frac{1}{k!} \left(\frac{i\hat{t}}{r} \right)^k \left(\frac{A}{\gamma} \right)^k \right)^r | m \rangle \\ &= \sum_{k_1, \dots, k_r=0}^K \frac{1}{k_1!} \cdots \frac{1}{k_r!} \left(\frac{i\hat{t}}{r} \right)^{k_1 + \dots + k_r} \\ &\quad \times \langle j | \left(\frac{A}{\gamma} \right)^{k_1 + \dots + k_r} | m \rangle, \end{aligned} \quad (56)$$

where we denote $\stackrel{\varepsilon}{\approx}$ as an additive approximation to error ε . We recall the quantity on the right-hand side may be approximated to additive error ε with success probability at least $1 - \delta$ with cost $K(\alpha^2/\varepsilon^2) \log(1/\delta)$ using Lemmas 7 and 8, where

$$\alpha = \sum_{k_1, \dots, k_r=0}^K \frac{1}{k_1!} \cdots \frac{1}{k_r!} \left(\frac{\hat{t}}{r} \right)^{k_1 + \dots + k_r} = \sum_{k=0}^K \frac{1}{k!} \left(\frac{\hat{t}}{r} \right)^k \leq e^{\hat{t}}. \quad (57)$$

Thus, we have an efficient algorithm when $\hat{t} = \mathcal{O}(\log \log(N))$.

For the local measurement problem we have

$$\begin{aligned} \langle j | e^{iAt} \pi e^{iAt} | j \rangle &\stackrel{2\varepsilon}{\approx} \quad (58) \\ &= \langle j | \sum_{k_1, \dots, k_r=0}^K \frac{1}{k_1!} \cdots \frac{1}{k_r!} \left(\frac{i\hat{t}}{r} \right)^{k_1 + \dots + k_r} \sum_{k'_1, \dots, k'_r=0}^K \frac{1}{k'_1!} \cdots \\ &\quad \times \frac{1}{k'_r!} \left(\frac{i\hat{t}}{r} \right)^{k'_1 + \dots + k'_r} \langle j | \left(\frac{A}{\gamma} \right)^{k_1 + \dots + k_r} \pi \left(\frac{A}{\gamma} \right)^{k'_1 + \dots + k'_r} | j \rangle, \end{aligned} \quad (59)$$

where similar to before the quantity on the right-hand side may be approximated to additive error ε with success probability at least $1 - \delta$ with cost $K \frac{\alpha^4}{\varepsilon^2} \log(\frac{1}{\delta})$ using Lemma 9. ■

Finally, we provide an algorithm for Hamiltonian Simulation for $\mathcal{O}(1)$ -sparse matrices A satisfying $\|A\| t \leq 1$.

Proposition 17 (Constant time evolution). Problems I and II for e^{iAt} are classically easy with $\mathcal{O}(1)$ -sparse Hermitian $A \in \mathbb{C}^{N \times N}$ satisfying $\|A\| t = \mathcal{O}(\log \log N)$ in the sparse access model.

Proof. We may assume $\|A\| = 1$ and $t = \mathcal{O}(\log \log N)$ by working on $A/\|A\|$ and evolving it for time $\|A\| t$, which is $\mathcal{O}(\log \log N)$ by hypothesis.

Given any precision ε , using the Anger-Jacobi expansion (Lemma 3) we may approximate e^{ixt} up to precision ε with a polynomial over Chebyshev polynomials of degree $m = \mathcal{O}(t + \log(1/\varepsilon))$. Thus, we can solve both problems by computing these Chebyshev polynomials using the first algorithm from Proposition 9, which has complexity $\mathcal{O}(ms^m) = ((t + \log(1/\varepsilon))s^{\mathcal{O}(t + \log(1/\varepsilon))})$. Since $1/\varepsilon = \text{poly log}(N)$ and $t = \mathcal{O}(\log \log N)$ it holds that $\mathcal{O}(t + \log(1/\varepsilon)) = \mathcal{O}(\log \log N)$ and thus the resulting algorithm is polylogarithmic on the dimension. ■

E. Classical eigenvalue transform

In this section, we present our classical algorithms for general classes of polynomials. We recall that hardness for general polynomials holds for both Problems I and II even for constant precision (e.g., consider Chebyshev polynomials in Theorem 4 and Proposition 7). Thus, we should not expect efficient classical algorithms for too generic a class of matrices—even $\mathcal{O}(1)$ -sparse matrices in sparse access. The algorithms which we elucidate here allow efficient processing of large matrices if they are very sparse (by direct application of Theorem 3 and Proposition 6), or for much milder conditions on the sparsity in Pauli access only if they have an inverse-polynomial-sized norm (by combining the above ideas with importance sampling).

Observation 2. Theorem 3 allows exact solution to Problems I and II in Pauli access to the matrix A for any degree- d polynomial in time $\mathcal{O}(d^2 L^2 \log N)$ and $\mathcal{O}((d^2 L^2 + 2^{2L}) \log N)$, respectively, where we recall L is the number of coefficients of A in the Pauli basis. Proposition 6 allows an exact solution to both Problem I and 2 in $\mathcal{O}(d^2 k^3)$ time when the k nonzero entries of A (in the computational basis) are given as a list. The additional factor of d is due to the fact that a general degree- d polynomial consists of $\mathcal{O}(d)$ monomials.

Theorem 6 (Super-sparse classical matrix processing). Consider a matrix A satisfying $\|A\| \leq 1$. Consider a function $f(x)$ which is approximated as $|f(x) - g(x)|_{[-1,1]} \leq \varepsilon$, where $g(x)$ is a polynomial of degree $d_{f,\varepsilon}$ computed in time $t_{f,\varepsilon}$. The entry estimation problem can be solved classically for $f(A)$ in time $\mathcal{O}(t_{f,\varepsilon} + d_{f,\varepsilon}^2 \cdot L^2 \log N)$ in the Pauli access model where L denotes the number of Pauli terms; or in time $\mathcal{O}(t_{f,\varepsilon} + d_{f,\varepsilon}^2 \cdot k^3)$ where k denotes the number of nonzero computational basis entries. The local measurement problem can be solved within polynomially equivalent runtimes.

Proof. Directly use the algorithm of Theorem 3 in the Pauli basis and Proposition 6 in the computational basis. In both cases, we obtain complexity quadratic in the degree

by noting that any degree- d polynomial can be constructed from its d constituent monomials. As both primitives are exact, the coefficients of the polynomials do not factor into the complexity. The dependence on $t_{f,\varepsilon}$ comes simply as a one-off preprocessing step. ■

Using Theorem 6 we directly have efficient algorithms for the inverse and time-evolution (complex exponential) functions whenever $L = \mathcal{O}(\log \log N)$ in the Pauli model or $k = \mathcal{O}(\text{poly log}(N))$ in the computational basis, whenever the condition number or evolution time is $\mathcal{O}(\text{poly log}(N))$. This can be seen from the fact that both functions have efficient polynomial approximations (Lemmas 2 and 3). Our result extends to matrices which are approximately sparse—that is, matrices which have many Pauli coefficients, but only a small subset which are large.

Theorem 7 (Approximately Pauli-sparse matrices). Return to the setting of Theorem 6. Consider $A = \sum_{\ell=1}^L a_\ell P_\ell$ with a long tail of Pauli coefficients, such that $\sum_{\ell \in [L]/s} |a_\ell| \leq \delta$ for some index set s . Then, the entry estimation problem can be solved in time $\mathcal{O}(t_{f,\varepsilon} + d_{f,\varepsilon}^2 \cdot |s| 2^{|s|} \log N)$, and the local measurement problem can be solved in time $\mathcal{O}(t_{f,\varepsilon} + (d_{f,\varepsilon}^2 \cdot |s| 2^{|s|} + 2^{2|s|}) \log N)$, both attaining additive error $\mathcal{O}(\varepsilon + d_{f,\varepsilon}^2 \delta \log N)$.

Proof. We run the same algorithms as in Theorem 6 on the matrix $A' = \sum_{\ell \in s} a_\ell P_\ell$, and show that they attain small error. For monomials of and we have the operator Lipschitz inequality of Ref. [89, Theorem 11.2] which states that

$$\|f(A) - f(A')\| = \mathcal{O}(L_f \|A - A'\| \log N), \quad (60)$$

where L_f is the Lipschitz constant of f on the spectrums of A and A' , which is simply the degree $d_{f,\varepsilon}$ in the case of our polynomial $g(A)$. The stated error bound follows by noting that the operator norm of any operator is upper bounded by the ℓ_1 norm of its Pauli coefficients. Finally, the stated runtimes simply follow once again by direct application of Theorem 3. ■

Let us now state a result that gives an alternate physical intuition as to why this set of problems is classically simulable.

Proposition 18. Consider a state mapped by a matrix polynomial $|\psi\rangle = P_d(A)|0\rangle/\|P_d(A)|0\rangle\|$. Across any cut $X \mid Y$ the reduced state ρ_X has quantum α -Rényi entropy satisfying

$$S_\alpha(\rho_X) \leq L + 1, \quad (61)$$

for any d and any $0 < \alpha < 1$. Thus, whenever $L = \mathcal{O}(\log \log N)$ then all the α -Rényi entropies (including the entanglement entropy) are $\mathcal{O}(\log \log(N))$.

Proof. From the proof of Theorem 3 we know that any polynomial of A consists of at most 2^{L+1} Pauli terms. Further, the application of a single Pauli on the $|0\rangle$ state is a product state. Thus, the Schmidt rank of $P_d(A)|0\rangle$ is upper bounded by 2^{L+1} and we have

$$S_0(\rho_X) := \log(\text{rank}(\rho_X)) \leq L + 1. \quad (62)$$

The proposition statement follows from the fact that the α -Rényi entropies are nonincreasing in α . ■

Different from the rest of the manuscript, here we are dealing with normalized states, but in most cases the normalization is well-behaved and itself can be evaluated in polynomial time. Importantly, Proposition 18 implies that the normalized states admit an efficient MPS representation, meaning one with bond dimension scaling polynomially in $\log N$ and the approximation error [90]. Note, however, that Proposition 18 itself does not automatically imply there is a classical polynomial time algorithm, as finding that MPS representation may be intractable.

We now move onto our second classical algorithm for general polynomials. Here, we combine the above algorithm with an importance-sampling sketch. This allows for matrix processing for generic sparsity, so long as one can efficiently sample from the Pauli coefficients.

Theorem 8 (Classical matrix processing with suppressed norm). Instantiate Problems I and II with a degree- d polynomial p_d which is bounded as $|p_d(x)|_{[-1,1]} \leq 1$. Suppose that we can sample from the Pauli coefficients of $A = \sum_\ell a_\ell P_\ell$ such that, with probability $|a_\ell|/\lambda_A$, the triple $(|a_\ell|, a_\ell, \ell)$ is returned and $\lambda_A = \sum_\ell |a_\ell|$ is known. Then, both problems can be solved efficiently to inverse-polynomial failure probability and constant error if the condition

$$d^2 \lambda_A \log \text{rank}(A) \sqrt{\log(N)} = \mathcal{O}(1), \quad (63)$$

is satisfied, such that $\lambda_A \leq 1 - \eta$ for some $\eta = \Omega(1)$. In particular, when the polynomial degree satisfies $d = \mathcal{O}(\text{poly log}(N))$, this implies that there is an efficient algorithm starting from Pauli access for A with some value of $\lambda_A = \mathcal{O}(1/\text{poly log}(N))$.

Proof. The algorithm follows by two steps: (1) perform an importance-sampling sketch on the Pauli coefficients; (2) use our algorithm for super-sparse Pauli matrix processing on the sketched matrix (Theorem 3). When we sample according to the distribution $\{|a_\ell|/\lambda_A\}_\ell$, each time upon obtaining index ℓ we output the (matrix-valued) random variable $X_\ell = a_\ell \lambda_A P_\ell / |a_\ell|$. This is an unbiased estimator for A . Sampling $L' = (8\lambda_A^2/\varepsilon^2) \log(2N/\delta)$ times, we obtain a Pauli representation of a matrix $A^{(L')}$ which satisfies an operator-norm approximation $\|A^{(L')} - A\| \leq \varepsilon'$.

This statement can be gleaned from operator-Bernstein inequalities [91, Theorem 6] (see Lemmas A12, A13 in Appendix A). We emphasize that here we do not need to perform any matrix arithmetic in the computational basis; we will only need to keep track of the Pauli coefficients of $A^{(L')}$, which is efficient in L' . Moreover, we note that the operator-norm condition allows us to write

$$\|A^{(L')}\| \leq \|A\| + \|A^{(L')} - A\| \leq 1 - \eta + \varepsilon', \quad (64)$$

where we have used the triangle inequality and fact that $\|A\| \leq \lambda_A \leq 1 - \eta$. Hereon we ensure that $\varepsilon' \leq \eta$ so that $\|A^{(L')}\| \leq 1$.

Now we use Theorem 3 on our representation of the matrix $A^{(L')}$ to obtain an exact Pauli representation of a degree- d polynomial $p_d(A^{(L')})$ to approximate both problems. We note that an operator-norm approximation of a general function f can be specified from known operator Lipschitz bounds [89, Theorem 11.2] as

$$\begin{aligned} & \|f(A^{(L')}) - f(A)\| \\ & \leq CL_f \|A^{(L')} - A\| \cdot \log(\min\{\text{rank}(A), \text{rank}(A^{(L')})\}) \\ & \leq CL_f \varepsilon' \cdot \log \text{rank}(A), \end{aligned} \quad (65)$$

for some numerical constant C , where L_f is the Lipschitz constant of f on the eigenvalues of $A^{(L')}$ and A , which both lie within $[-1, 1]$. Thus, for a choice of $\tilde{L}$, given computation of $p_d(A^{(\tilde{L})})$ from the Pauli coefficients of $A^{(\tilde{L})}$ we have operator-norm approximation

$$\|p_d(A^{(\tilde{L})}) - p_d(A)\| \leq \varepsilon, \quad (66)$$

with probability at least $(1 - \delta)$, where $A^{(\tilde{L})}$ is constructed from $\tilde{L} = \mathcal{O}((\lambda_A^2 d^4 / \varepsilon^2) \log^2 \text{rank}(A) \log(2N/\delta))$ Pauli terms. Here we have used the fact that bounded degree- d polynomials on $[-1, 1]$ have Lipschitz constant d^2 (see Lemma A14). As the dominant factor in the runtime of Theorem 3 is $\mathcal{O}(2^{\tilde{L}})$ for Problem I and $\mathcal{O}(2^{2\tilde{L}})$ for Problem II, we have an efficient algorithm if we ask for constant precision ε , inverse-polynomial failure probability, and when our stated condition is satisfied. Specifically, choosing any constant $\varepsilon \leq C\eta$ ensures that $\varepsilon' \leq \eta$ as previously required. ■

Let us end with two contextualizations of Theorem 8. The specified sampling access can be efficiently instantiated starting from Pauli access as a preprocessing step whenever the number of Pauli terms is $L = \mathcal{O}(\text{poly log}(N))$, or for larger L whenever the coefficients have sufficient structure. Finally, we can understand condition (63) as a suppressed-norm condition on the 1-norm

of Pauli coefficients. Equally, we can see this as a condition on other matrix norms via standard norm conversions—for instance, a sufficient condition which implies (63) is to suppress the Frobenius norm of a matrix as $\|A\|_F = \mathcal{O}(\sqrt{N}/(d^2\sqrt{L}\log^{1.5}(N)))$.

ACKNOWLEDGMENTS

The authors would like to thank Simon Apers, Ariel Bendersky, Fernando Brandão, Tom O’Leary, and James Watson for helpful discussions. M.B. acknowledges support from the EPSRC Grant No. EP/W032643/1 and the Excellence Cluster—Matter and Light for Quantum Computing (ML4Q). S.W. acknowledges funding provided by the Institute for Quantum Information and Matter, an NSF Physics Frontiers Center (NSF Grant PHY-2317110). S.W. and M.B. thank the Technology Innovation Institute for scientific visits, when part of this work was carried out. S.C. acknowledges financial support from the Technology Innovation Institute for a long-term internship.

DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

APPENDIX A

1. Useful lemmas

We recall the Chebyshev polynomials and the Bessel functions alongside some of their properties.

Definition A1 (Chebyshev polynomials and the Bessel functions). The Chebyshev polynomials of the first kind are obtained from the recurrence relation

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x). \end{aligned} \quad (\text{A1})$$

The Chebyshev polynomials of the second kind U_n are obtained following the same recurrence, but considering $U_1(x) = 2x$. They satisfy

$$\begin{aligned} |T_n(x)| &\leq 1 \quad \text{for } x \in [-1, 1], \\ |U_n(x)| &\leq n+1 \quad \text{for } x \in [-1, 1], \\ T_n\left(\cos\left(\frac{\pi j}{n}\right)\right) &= (-1)^j, \\ T'_n(x) &= nU_{n-1}(x). \end{aligned} \quad (\text{A2})$$

The Bessel functions of the first kind, denoted as $J_\alpha(x)$, where $\alpha \in \mathbb{R}$ are defined as

$$J_\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m! \Gamma(m + \alpha + 1)} \left(\frac{x}{2}\right)^{2m+\alpha}. \quad (\text{A3})$$

For $\alpha = 0, 1, 2, \dots$, it holds that $|J_\alpha(x)| \leq 1$ for all $x \in \mathbb{R}^{\geq 0}$.

Employing them it is possible to approximate the function e^{itx} with a polynomial of low degree. Moreover, the coefficients of these polynomials are also small.

Lemma A1 (Bound on the coefficients of Chebyshev polynomials). The coefficients of the polynomial $T_n(x)$ are upper bounded by 4^n .

Proof. We prove this simple fact by induction. Observe that it holds for $n = 0, 1$. For the inductive case, let c_n denote the biggest coefficient of the n th Chebyshev polynomial. Then, by Eq. (A1) it holds that

$$c_n \leq 2c_{n-1} + c_{n-2} \leq 2 \times 4^{n-1} + 4^{n-2} \leq 4^n. \quad (\text{A4})$$

Whenever A is hermitian the value $\langle i|A|j \rangle$ can be expressed as a linear combination of terms of the form $\langle \psi|A|\psi \rangle$ for different vectors $|\psi \rangle$. Thus, any algorithm that computes $\langle \psi|A|\psi \rangle$ for arbitrary vectors $|\psi \rangle$ can be employed to solve the proposed problem. We prove this simple algebraic property for completeness.

Lemma A2 (Decomposition of off-diagonal entries). Let $A \in \mathbb{C}^{N \times N}$ be a Hermitian matrix. Then, for any $k, j \in [N]$ the value $\langle k|A|j \rangle$ can be written as a linear combination of a constant number of terms of the form $\langle \psi|A|\psi \rangle$.

Proof. We show that both the real part $\text{Re}(\langle k|A|j \rangle)$ and the imaginary part $\text{Im}(\langle k|A|j \rangle)$ can be written as a linear combination of terms of the form $\langle \psi|A|\psi \rangle$,

$$\begin{aligned} 2\text{Re}\langle k|A|j \rangle &= \langle k|A|j \rangle + \langle j|A|k \rangle \\ &= (\langle k| + \langle j| - \langle j|)A|j \rangle + \langle j|A|k \rangle \\ &= (\langle k| + \langle j|)A|j \rangle - \langle j|A|j \rangle + \langle j|A|k \rangle \\ &= (\langle k| + \langle j|)A(|k \rangle + |j \rangle - |k \rangle) - \langle j|A|j \rangle \\ &\quad + \langle j|A|k \rangle \\ &= (\langle k| + \langle j|)A(|k \rangle + |j \rangle) - \langle k|A|k \rangle - \langle j|A|j \rangle, \end{aligned} \quad (\text{A5})$$

$$\begin{aligned} 2i\text{Im}\langle k|A|j \rangle &= \langle k|A|j \rangle - \langle j|A|k \rangle \\ &= (\langle k| + i\langle j| - i\langle j|)A|j \rangle - \langle j|A|k \rangle \\ &= i(\langle k| + i\langle j|)A(-i|j \rangle) - i\langle j|A|j \rangle - \langle j|A|k \rangle \\ &= i(\langle k| + i\langle j|)A(|k \rangle - i|j \rangle - |k \rangle) - i\langle j|A|j \rangle \\ &\quad - \langle j|A|k \rangle \\ &= i(\langle k| + i\langle j|)A(|k \rangle - i|j \rangle) - i\langle k|A|k \rangle \\ &\quad - i\langle j|A|j \rangle. \end{aligned} \quad (\text{A6})$$

We now demonstrate a lemma on the well-known spectral decomposition of the cyclic operators.

Lemma A3 (Spectral decomposition of cyclic shifts). Consider the cyclic shift operator $S = \sum_{\ell=0}^{M-1} |\ell+1\rangle\langle\ell|$, where the $+$ operation is understood modulo M . Then, the eigenvalues of S are $e^{\frac{i2\pi k}{M}}$ with corresponding eigenvectors $|\psi_k\rangle = (1/\sqrt{M}) \sum_{\ell=0}^{M-1} e^{-\frac{i2\pi k\ell}{M}} |\ell\rangle$, for $k = 0, \dots, M-1$. Moreover, $|0\rangle = (1/\sqrt{M}) \sum_{k=0}^{M-1} |\psi_k\rangle$.

Consider the cyclic shift with an additional -1 phase factor, defined as $S' = \sum_{\ell=0}^{M-2} |\ell+1\rangle\langle\ell| - |0\rangle\langle M-1|$. Then, the eigenvalues of S' are $e^{\frac{i\pi(2k+1)}{M}}$ with corresponding eigenvectors $|\psi'_k\rangle = (1/\sqrt{M}) \sum_{\ell=0}^{M-1} e^{-\frac{i\pi(2k+1)\ell}{M}} |\ell\rangle$, for $k = 0, \dots, M-1$. Moreover, $|0\rangle = (1/\sqrt{M}) \sum_{k=0}^{M-1} |\psi'_k\rangle$.

Proof. Observe that $S^M - \mathbb{1} = 0$. Thus, all eigenvalues of S are of the form $e^{\frac{i2\pi k}{M}}$ for $k = 0, \dots, M-1$. By direct computation

$$\begin{aligned} S|\psi_k\rangle &= \frac{1}{\sqrt{M}} \sum_{\ell=0}^{M-1} e^{-\frac{i2\pi k\ell}{M}} S|\ell\rangle = \frac{1}{\sqrt{M}} \sum_{\ell=0}^{M-1} e^{-\frac{i2\pi k\ell}{M}} |\ell+1\rangle \\ &= e^{\frac{i2\pi k}{M}} \frac{1}{\sqrt{M}} \sum_{\ell=0}^{M-1} e^{-\frac{i2\pi k(\ell+1)}{M}} |\ell+1\rangle = e^{\frac{i2\pi k}{M}} |\psi_k\rangle. \end{aligned} \quad (\text{A7})$$

Finally $(1/\sqrt{M}) \sum_{k=0}^{M-1} |\psi_k\rangle = (1/M) \sum_{k=0}^{M-1} \sum_{\ell=0}^{M-1} e^{-\frac{i2\pi k\ell}{M}} |\ell\rangle = (1/M) \sum_{\ell=0}^{M-1} \left(\sum_{k=0}^{M-1} e^{-\frac{i2\pi k\ell}{M}} \right) |\ell\rangle = |0\rangle$.

Regarding S' , note that $(S')^M + \mathbb{1} = 0$, and thus all its eigenvalues are of the form $e^{\frac{i\pi(2k+1)}{M}}$ for $k = 0, \dots, M-1$. The rest follows from the same computations as before. ■

Lemma A4. Let W be defined in Eq. (12) and let $|s_x\rangle = |\text{step}_0\rangle|\mathbf{x}\rangle|0\rangle^{\otimes r-n}$ be as in Eq. (5). Then,

$$\frac{Q^+|s_x\rangle}{|\alpha_{x,0}|} = |\text{step}_0\rangle|\phi_0^+\rangle \quad \text{s.t. } C|\phi_0^+\rangle = \frac{\alpha_{x,0}}{|\alpha_{x,0}|} |0\rangle|\psi_{x,0}\rangle \quad (\text{A8})$$

$$\frac{Q^-|s_x\rangle}{|\alpha_{x,1}|} = |\text{step}_0\rangle|\phi_0^-\rangle \quad \text{s.t. } C|\phi_0^-\rangle = \frac{\alpha_{x,1}}{|\alpha_{x,1}|} |1\rangle|\psi_{x,1}\rangle, \quad (\text{A9})$$

where $Q^\pm = (\mathbb{1} \pm W^M)/2$ are the projectors onto the eigenspaces $\mathcal{S}^\pm$ of W^M with eigenvalues $+1$ and -1 .

Proof. The normalization factor of the vector $Q^+|s_x\rangle$ can be computed as

$$\begin{aligned} \langle s_x|Q^+|s_x\rangle &= \frac{1}{2} \langle 0|\langle \mathbf{x}|\langle 0|\mathbb{1} + W^M|0\rangle|\mathbf{x}\rangle|0\rangle \\ &= \frac{1}{2} (1 + \langle 0|\langle \mathbf{x}|C^\dagger(Z \otimes \mathbb{1}^{r-1})C|\mathbf{x}\rangle|0\rangle) \\ &= |\alpha_{x,0}|^2, \end{aligned} \quad (\text{A10})$$

where the last two equalities follow by using Eqs. (13) and (5), respectively. Analogously, $\langle s_x|Q^-|s_x\rangle = |\alpha_{x,1}|^2$. By using Eq (13), we obtain

$$\begin{aligned} \frac{Q^\pm|s_x\rangle}{|\alpha_{x,0}|} &= \frac{1 \pm W^M}{2} |s_x\rangle = \frac{1}{2} |\text{step}_0\rangle (|\mathbf{x}\rangle|0\rangle^{\otimes r-n} \\ &\quad \pm C^\dagger(Z \otimes \mathbb{1}^{r-1})C|\mathbf{x}\rangle|0\rangle^{\otimes r-n}) := |\text{step}_0\rangle|\phi_0^\pm\rangle \end{aligned} \quad (\text{A11})$$

from which we can directly calculate $C|\phi_0^\pm\rangle$ by using Eq. (5). ■

Lemma A5. Let W be defined in Eq. (12). Then the eigenvalues of W are $e^{-\frac{i2\pi\ell}{M}}$ and $e^{\frac{i\pi(2\ell+1)}{M}}$, with $\ell = 0, \dots, M-1$.

Let $|s_x\rangle = |\text{step}_0\rangle|\mathbf{x}\rangle|0\rangle^{\otimes r-n}$ be the input bitstring to BQPCIRCUITSIMULATION. Denote P_ℓ^+ and P_ℓ^- the projectors onto the subspace corresponding to eigenvalues $e^{-\frac{i2\pi\ell}{M}}$ and $e^{\frac{i\pi(2\ell+1)}{M}}$, respectively. Then, $\omega_\ell^+ := \langle s_x|P_\ell^+|s_x\rangle = |\alpha_{x,0}|^2/M$ and $\omega_\ell^- := \langle s_x|P_\ell^-|s_x\rangle = |\alpha_{x,1}|^2/M$, for $\ell = 0, \dots, M-1$.

Proof. To prove the first part of the lemma, consider the sequence of states $|\phi_0\rangle, |\phi_1\rangle = V_0|\phi_0\rangle, |\phi_2\rangle = V_1|\phi_1\rangle, \dots, |\phi_{M-1}\rangle = V_{M-2}|\phi_{M-2}\rangle$ built from a state $|\phi_0\rangle$ of r qubits. Notice that if $C|\phi_0\rangle = \alpha_0|0\rangle|\psi_0\rangle^{r-1}$ with $|\alpha_0| = 1$ then $V_{M-1}|\phi_{M-1}\rangle = C^\dagger(Z \otimes \mathbb{1}^{r-1})C|\phi_0\rangle = |\phi_0\rangle$. Therefore, similar to Lemma A3, one can verify by direct calculation that the state $|\psi_k^+\rangle = (1/\sqrt{M}) \sum_{\ell=0}^{M-1} e^{-\frac{i2\pi k\ell}{M}} |\text{step}_\ell\rangle \otimes |\phi_\ell\rangle$ is an eigenstate of W with eigenvalue $e^{i\frac{2\pi k\ell}{M}}$. Similarly, if $C|\phi_0\rangle = \alpha_1|1\rangle|\psi_1\rangle^{r-1}$ with $|\alpha_1| = 1$ then $V_{M-1}|\phi_{M-1}\rangle = C^\dagger(Z \otimes \mathbb{1}^{r-1})C|\phi_0\rangle = -|\phi_0\rangle$ and one can verify that $|\psi_k^-\rangle = (1/\sqrt{M}) \sum_{\ell=0}^{M-1} e^{-\frac{i\pi(2k+1)\ell}{M}} |\text{step}_\ell\rangle \otimes |\phi_\ell\rangle$ is an eigenstate of W with eigenvalue $e^{i\frac{\pi(2k+1)\ell}{M}}$.

To prove the second part, we start by noticing that, analogous to Lemma A3, $(1/\sqrt{M}) \sum_{k=0}^{M-1} |\psi_k^+\rangle = |\text{step}_0\rangle \otimes |\phi_0\rangle$ and, therefore, the overlap $\langle \text{step}_0| \otimes \langle \phi_0|P_k^+|\text{step}_0\rangle \otimes |\phi_0\rangle = 1/M$. From this fact and Eq. (A8) in Lemma A4, we get the desired overlap ω_ℓ^+ , since $|s_x\rangle = Q^+|s_x\rangle + Q^-|s_x\rangle$ and $|\phi_0^+\rangle$ can be used to build a sequence $|\phi_0^+\rangle, \dots, |\phi_{M-1}^+\rangle = V_{M-2}|\phi_{M-2}^+\rangle$. The overlap ω_ℓ^- is obtained analogously. ■

Lemma A6. Let $\mathcal{G} = \{P_\ell\}_{1 \leq \ell \leq L}$ be a set of L $2^n \times 2^n$ generalized Pauli matrices and denote by $\langle \mathcal{G} \rangle$ the Pauli subgroup generated by $\mathcal{G}$. Then, $|\langle \mathcal{G} \rangle| \leq 2^{L+1}$.

Proof. Since any pair of generalized Pauli matrices either commutes or anticommutes (and $P_\ell^2 = 1$ for any P_ℓ) it holds that

$$\langle \mathcal{G} \rangle = \{(-1)^{k_0} P_1^{k_1} \dots P_L^{k_L} : k_i \in \{0, 1\}, \quad 0 \leq i \leq L\}. \quad (\text{A12})$$

Therefore, $|\langle \mathcal{G} \rangle| \leq 2^{L+1}$. ■

Lemma A7 (Pauli decomposition of basis elements). Any computational basis matrix element $|i\rangle\langle j| \in \mathbb{C}^{N \times N}$ has the form

$$|i\rangle\langle j| = \sum_{\ell=1}^{2^{\lceil \log N \rceil}} a_\ell^{(ij)} P_\ell, \quad (\text{A13})$$

where $|a_\ell^{ij}| = 2^{-\lceil \log N \rceil}$. Thus, $|i\rangle\langle j|$ has Pauli norm 1.

Proof. Over $\mathbb{C}^{N \times N}$ the basis matrix elements can be expressed in terms of single-qubit Pauli matrices,

$$\begin{aligned} |0\rangle\langle 0| &= \frac{1}{2}(\mathbb{1} + Z), \\ |1\rangle\langle 1| &= \frac{1}{2}(\mathbb{1} - Z), \\ |0\rangle\langle 1| &= \frac{1}{2}(X + iY), \\ |1\rangle\langle 0| &= \frac{1}{2}(X - iY). \end{aligned} \quad (\text{A14})$$

Then, $|i\rangle\langle j|$ can be written as a tensor product of $\lceil \log N \rceil$ of these elements, which gives a Pauli decomposition of $2^{\lceil \log N \rceil}$ terms with coefficients of magnitude $2^{-\lceil \log N \rceil}$. Furthermore, each index can be computed classically in $\mathcal{O}(\log N)$ given i, j . ■

Lemma A8 (Pauli decomposition of universal gates). It holds that

$$\begin{aligned} H &= \frac{\mathbb{1} + X}{\sqrt{2}}, \\ T &= \frac{3}{4}\mathbb{1}\mathbb{1}\mathbb{1} + \frac{1}{4}Z\mathbb{1}\mathbb{1} + \frac{1}{4}\mathbb{1}Z\mathbb{1} - \frac{1}{4}ZZ\mathbb{1} + \frac{1}{4}\mathbb{1}\mathbb{1}X \\ &\quad - \frac{1}{4}Z\mathbb{1}X - \frac{1}{4}\mathbb{1}ZX + \frac{1}{4}ZZX, \end{aligned} \quad (\text{A15})$$

where we denote H as the Hadamard gate, T as the Toffoli gate, and we use $P_1 P_2 P_3$ as shorthand for $P_1 \otimes P_2 \otimes P_3$.

Lemma A9 (Pauli norm of unitary). The Pauli norm of any n -qubit unitary satisfies $\lambda \leq 4^n$.

Proof. We prove this by showing that the magnitude of each coefficient in the Pauli decomposition cannot be larger than 1. Denote an arbitrary unitary as $U = \sum_{\ell=1}^{4^n} a_\ell P_\ell$. First observe that

$$\text{Tr}[UP_\ell] = \text{Tr}[a_\ell P_\ell^2] = 2^n a_\ell, \quad (\text{A16})$$

and moreover that

$$|\text{Tr}[UP_\ell]| \leq \|U\|_2 \|P_\ell\|_2 = 2^n, \quad (\text{A17})$$

where we have used the Cauchy-Schwarz inequality, followed by the fact that U and P_ℓ are unitary. Together these two equations imply that $|a_\ell| \leq 1$ for all $\ell \in [4^n]$. ■

Lemma A10 (Pauli norm is multiplicative). If A and B are Hermitian matrices with Pauli norm λ_A and λ_B , then $A \otimes B$ satisfies

$$\lambda_{A \otimes B} = \lambda_A \lambda_B. \quad (\text{A18})$$

Proof. Let $A = \sum_\ell a_\ell P_\ell, B = \sum_k b_k P_k$, then

$$A \otimes B = \sum_{\ell, k} a_\ell b_k P_\ell \otimes P_k \quad (\text{A19})$$

and

$$\lambda_{A \otimes B} = \sum_{k, \ell} |a_\ell b_k| = \sum_{k, \ell} |a_\ell| |b_k| = \lambda_A \lambda_B. \quad (\text{A20})$$

■

Lemma A11 (Pauli decomposition of clock construction). For any M and $0 \leq k \leq M-1$ it holds that the operator $\mathbb{1}^{\otimes k} \otimes |10\rangle\langle 01| \otimes \mathbb{1}^{\otimes M-k-2}$ has a Pauli decomposition of weight 1 and $\mathcal{O}(1)$ Pauli terms.

Proof. By Lemma A7 the operator $|10\rangle\langle 01|$ has Pauli weight 1 and is written down as a sum of 4 Pauli terms. We can pad these terms with $M-k-2$ identities on the back and k identities upfront to obtain the result. ■

Lemma A12 (Operator-Bernstein inequality; adapted from Ref. [91, Theorem 6]).]

Let $X_i \in \mathbb{C}^{N \times N}$ be i.i.d. Hermitian matrix-valued random variables. Take $p, q \in \mathbb{R}$ such that $\|\mathbb{E}[(X_i - \mathbb{E}[X_i])^2]\| \leq p^2$ and $\|X_i - \mathbb{E}[X_i]\| \leq q$. Then, for any $\varepsilon \leq 2mp/q$ we have

$$\text{Prob} \left[\left\| \frac{1}{m} \sum_i X_i - \mathbb{E}[X_i] \right\| > \varepsilon \right] \leq 2N \exp \left(-\frac{\varepsilon^2 m}{4p^2} \right). \quad (\text{A21})$$

Let us now inspect what this implies for importance-sampling matrices in the Pauli basis.

Lemma A13 (Importance sampling in the Pauli basis). For Hermitian $A = \sum_l a_l P_l$ decomposed in the Pauli basis, denote $\lambda_A = \sum_l |a_l|$. Suppose we sample according to the distribution $\{|a_l|/\lambda_A\}_l$ and each time upon obtaining index l output random variable $X_l = (a_l/|a_l|)\lambda_A P_l$. This is an unbiased estimator for A , and we obtain $\|\sum_{i=1}^m X_i - A\| \leq \varepsilon \leq 1$ with probability at least $(1 - \delta)$ for any number of samples

$$m \geq \frac{8\lambda_A^2}{\varepsilon^2} \log\left(\frac{2N}{\delta}\right). \quad (\text{A22})$$

Proof. We directly use Lemma A12. First, we see that $\mathbb{E}[X_i] = A$. We also have the following bound:

$$\|X_i - A\| \leq \|X_i\| + \|A\| \leq 2\lambda_A, \quad (\text{A23})$$

where we have used the triangle inequality and the fact that $\|A\| \leq \lambda_A$. Additionally, we have

$$\begin{aligned} \|\mathbb{E}[(X_i - A)^2]\| &= \|\mathbb{E}[(X_i)^2] - A^2\| \leq \|\mathbb{E}[(X_i)^2]\| \\ &\quad + \|A^2\| \leq 2\lambda_A^2, \end{aligned} \quad (\text{A24})$$

where we have used the submultiplicativity of the operator norm and the fact that $X_i^2 = \lambda_A^2 \mathbb{1}$. Thus, based on these two bounds we can take $p = \sqrt{2}\lambda_A$ and $q = 2\lambda_A$. Using these values for Eq. (A21) we obtain the stated result. ■

Lemma A14 (Lipschitz constant of bounded polynomial, Ref. [92]). Let p_d be a polynomial of degree d and let $c = \max_x |p_d(x)|_{[a,b]}$ over some interval $[a, b]$. Then, the derivative of $p_d(x)$ [denote as $p'_d(x)$] satisfies

$$|p'_d(x)|_{[a,b]} \leq \frac{2c \cdot d^2}{b - a}. \quad (\text{A25})$$

2. Additional results and proofs

We start by providing a proof of Lemma 9, which gives a classical randomized algorithm for polynomials for the local measurement problem (Problem II).

Proof of Lemma 9. Let us first deal with the algorithm for Pauli access. We can explicitly write

$$\begin{aligned} &\langle i | f(A) \pi f(A) | i \rangle \\ &= W \sum_{r,r'} \sum_{\ell_1 \dots \ell_r} \sum_{\ell'_1 \dots \ell'_r} \frac{\alpha_r \alpha_{r'} (a_{\ell_1} \dots a_{\ell_r}) (a_{\ell'_1} \dots a_{\ell'_r})}{2W} \\ &\quad \times \langle i | P_{\ell_1} \dots P_{\ell_r} (\mathbb{1}^{\otimes n} + Z \otimes \mathbb{1}^{\otimes(n-1)}) P'_{\ell'_1} \dots P'_{\ell'_r} | i \rangle, \end{aligned} \quad (\text{A26})$$

where we denote $W = \sum_{r,r'} \sum_{\ell_1 \dots \ell_r} \sum_{\ell'_1 \dots \ell'_r} |\alpha_r \alpha_{r'} (a_{\ell_1} \dots a_{\ell_r}) (a_{\ell'_1} \dots a_{\ell'_r})|$, which can be bounded as

$$\begin{aligned} W &\leq \left(\sum_r \sum_{\ell_1 \dots \ell_r} |\alpha_r (a_{\ell_1} \dots a_{\ell_r})| \right)^2 \\ &\leq \left(\sum_r |\alpha_r \lambda_A^r| \right)^2 = \|f(\lambda_A x)\|_{\ell_1}^2. \end{aligned} \quad (\text{A27})$$

We can interpret Eq. (A26) as W multiplied by a probabilistic sum over (the diagonal entry of) Pauli strings with a phase factor, each appearing with probability $\{|\alpha_r \alpha_{r'} (a_{\ell_1} \dots a_{\ell_r}) (a_{\ell'_1} \dots a_{\ell'_r})|/2W\}$. Thus, we can sample from this probability distribution. By Hoeffding's inequality, the sample complexity required to attain precision ε with probability at least $(1 - \delta)$ is

$$C_{\text{samp}} = \mathcal{O}\left(\frac{W^2}{\varepsilon^2} \log\left(\frac{1}{\delta}\right)\right) = \mathcal{O}\left(\frac{\|f(\lambda_A x)\|_{\ell_1}^4}{\varepsilon^2} \log\left(\frac{1}{\delta}\right)\right). \quad (\text{A28})$$

For each sample, we must evaluate the diagonal entry of a product of up to $m + 1$ Pauli strings. This costs $\mathcal{O}(m \log N)$ time complexity per sample.

For the sparse problem, a similar sampling procedure is used following the technique of Refs. [31] and [24], with the simple observation that π is a 1-sparse matrix of 1s on the diagonal, and so is trivially integrated into a path integral Monte Carlo algorithm. ■

Next, we show a statement of hardness for the local measurement problem, when it is normalized (i.e., built from normalized quantum states). Let us start by formally defining the normalized problem.

Problem: NORMALIZED – LM – MONOMIAL^{AMODEL} _{$\|A\|$}

Input: An $N \times N$ Hermitian matrix A with $\|A\| \leq 1$ and accessible through AMODEL, a positive real number m , a precision ε and a threshold g , such that $m, 1/\varepsilon, g = \mathcal{O}(\text{poly log}(N))$. b is an upper bound on $\|A\|$, t and $\frac{1}{\varepsilon}$ are poly log (N) .

Output: Denote $\pi = |0\rangle\langle 0| \otimes \mathbb{1}_{N/2}$ and $r = \langle 0 | A^m \pi A^m | 0 \rangle / \|A^m | 0 \rangle\|^2$. Then, answer YES if $r \geq g + \varepsilon$ and NO if $r \leq g - \varepsilon$.

Proposition A1. NORMALIZED-LM-MONO-MIAL^{SPARSEACCESS} _{$\|A\|$} and NORMALIZED-LM-MONO-MIAL^{PAULIACCESS} _{$\|A\|$} are BQP-HARD, even for constant precision $1/\varepsilon = \Omega(1)$.

Proof. We consider the same matrix A as in Eq. (24) for the proof of the unnormalized problem. Now, for

$|\phi_m\rangle = A^M|0\rangle/\|A^M|0\rangle\|$ explicit evaluation gives

$$\langle\phi_m|\pi|\phi_m\rangle = \frac{\sum_{k+1 \leq \ell \leq 2k+2} p_t^2(\ell)}{\sum_{\ell} p_t^2(\ell)} |\alpha_{0,1}|^2. \quad (\text{A29})$$

For $p_{\infty} = u$ the stationary ratio in the above expression is now $(k+2)/M \geq \frac{1}{3}$ (denote this value as a). We now show that for large enough t the ratio is still a constant, and thus any problem in BQP can be simulated by solving the normalized local measurement problem. For an arbitrary distribution $\mathbf{p}_m$ satisfying $\|\mathbf{p}_m - \mathbf{u}\|_1 = \varepsilon$ the numerator of the ratio is minimized when $\mathbf{p}_m$ takes uniform value $1/M - a\varepsilon/2$ across all $k+1 \leq \ell \leq 2k+2$. The denominator is maximized for the peaked distribution where $\mathbf{p}_m(k+1) = 1/M - \varepsilon/2$, $\mathbf{p}_m(k) = 1/M + \varepsilon/2$, and $\mathbf{p}_m(\ell) = 1/M$ otherwise. Thus we have

$$\begin{aligned} & \frac{\sum_{k+1 \leq \ell \leq 2k+2} p_t^2(\ell)}{\sum_{\ell} p_t^2(\ell)} \\ & \geq \frac{\sum_{k+1 \leq \ell \leq 2k+2} \left(\frac{1}{M} - \frac{a\varepsilon}{2}\right)^2}{\left(\frac{1}{M} - \frac{\varepsilon}{2}\right)^2 + \left(\frac{1}{M} + \frac{\varepsilon}{2}\right)^2 + (M-2)\frac{1}{M^2}} \end{aligned} \quad (\text{A30})$$

$$\geq \frac{1 - \varepsilon M/2}{3 + \varepsilon^2 M/4} \quad (\text{A31})$$

$$\geq \frac{2}{13}, \quad (\text{A32})$$

where the last inequality is true for any $\varepsilon \leq 1/M$ (nothing that $M \geq 1$). Inspecting Eq. (26), it is thus sufficient to take $t = \mathcal{O}(M^2 \log M) = \mathcal{O}(\text{poly log}(N))$ for $M = \mathcal{O}(\text{poly log}(N))$. This ensures that $\langle\phi_m|\pi|\phi_m\rangle \geq \frac{2}{13}|\alpha_{0,1}|^2$ which can be determined by solving the monomial problem to error $\mathcal{O}(1)$. As shown in the proofs of Theorem 1 and Proposition 1, A can be instantiated in both sparse and Pauli access efficiently. ■

Proposition A2. The problems $\text{INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$ and $\text{INVERSE}_{\lambda_A}^{\text{PAULIACCESS}}$ are BQP-HARD, even under the conditions $\|A\|_1 = \mathcal{O}(1/\text{poly log}(N))$ and $\lambda_A = \mathcal{O}(1/\text{poly log}(N))$ (hardness statement from Theorem 5).

Proof. We begin with $\text{INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$. We use once again Eq. (17), but considering the matrix $A/2$, where A is defined in Eq. (15). It holds that $\|A/2\|_1 \leq 1$, and the eigenvalues of $A/2$ are the same eigenvalues as those from A , but divided by two.

The largest eigenvalue of A (in magnitude) is $\cos(0) = 1$. If $T = 0 \bmod 2$ the eigenvalue with smallest magnitude is $\cos(\pi T/(2T+1))$, while if $T = 1 \bmod 2$ it is $\cos(\pi(T+1)/(2T+1))$. Using the fact that $\cos(x) = -\sin(x - \pi/2)$ and the standard bounds $x - x^3/6 \leq \sin(x) \leq x$ it can be seen that $\kappa_A = \Omega(\text{poly log}(M))$: in the

former case $\kappa_A = \cos(\pi T/(2T+1))^{-1} = \sin(\pi/2(2T+1))^{-1} \geq 2(2T+1)/\pi = 2M/\pi$, while in the latter $\kappa_A = |\cos(\pi(T+1)/(2T+1))|^{-1} = \sin(\pi/2(2T+1))^{-1} \geq 2M/\pi$.

Following the notation from Theorem 1 and Eq. (22) (the inverse function is odd), we need to show that

$$\frac{2}{M} \left(1 + \sum_{\ell=1}^{\frac{M-1}{2}} \frac{2}{\theta_{\ell}^+} \right) \geq k, \quad (\text{A33})$$

for some constant k .

Assume that $T = 0 \bmod 2$ and observe that $(\theta_{\ell+1}^+)^{-1} + (\theta_{\frac{M-1}{2}-\ell}^+)^{-1} \geq 0$ for all $\ell = 0, \dots, \frac{M-1}{4} - 1$. Therefore,

$$\begin{aligned} \sum_{\ell=1}^{\frac{M-1}{2}} (\theta_{\ell}^+)^{-1} &= \sum_{\ell=0}^{\frac{M-1}{4}-1} (\theta_{\ell+1}^+)^{-1} + (\theta_{\frac{M-1}{2}-\ell}^+)^{-1} \\ &\geq (\theta_{\frac{M-1}{4}}^+)^{-1} + (\theta_{\frac{M-1}{4}+1}^+)^{-1} \\ &\geq \cos\left(\frac{\pi(\frac{M-1}{2})}{M}\right)^{-1} + \cos\left(\frac{\pi(\frac{M-1}{2}+2)}{M}\right)^{-1} \\ &= \sin\left(\frac{\pi}{2M}\right)^{-1} - \sin\left(\frac{3\pi}{2M}\right)^{-1} \\ &\geq \frac{2M}{\pi} - \frac{2M}{3\pi - \frac{27\pi^3}{24M^2}}. \end{aligned} \quad (\text{A34})$$

If M is big enough ($M \geq 4$ suffices) then $27\pi^3/24M^2 \leq \pi$, and consequently

$$\frac{2M}{\pi} - \frac{2M}{3\pi - \frac{162\pi}{4M^2}} \geq \frac{2M}{\pi} - \frac{2M}{2\pi} = \frac{M}{\pi}$$

and thus the left-hand side of Eq. (A33) is lower bounded by $1/\pi$. We observe that constraining that $M \geq 7$ does not affect the proof, since the problem BQPCIRCUITSIMULATION is still BQP-HARD if its input is conditioned this way.

Meanwhile, if $T = 1 \bmod 2$ it holds that $(\theta_{\ell}^+)^{-1} + (\theta_{\frac{M-1}{2}-\ell}^+)^{-1} \leq 0$ for $\ell = 0, \dots, \frac{T-1}{2}$. Thus,

$$\begin{aligned}
1 + \sum_{\ell=1}^{\frac{M-1}{2}} \frac{2}{\theta_{\ell}^+} &= 1 + 2 \left(\theta_{\frac{M-1}{2}}^+ \right)^{-1} + 2 \sum_{\ell=1}^{\frac{T-1}{2}} \left((\theta_{\ell}^+)^{-1} + \left(\theta_{\frac{M-1}{2}-\ell}^+ \right)^{-1} \right) \\
&\leq 2 \left(\left(\theta_{\frac{T-1}{2}}^+ \right)^{-1} + \left(\theta_{\frac{T+1}{2}}^+ \right) \right) \\
&= 2 \left(\cos \left(\frac{\pi(T-1)}{M} \right)^{-1} + \cos \left(\frac{T+1}{2} \right)^{-1} \right) \\
&= 2 \left(\sin \left(\frac{3\pi}{2M} \right)^{-1} - \sin \left(\frac{\pi}{2M} \right)^{-1} \right). \tag{A35}
\end{aligned}$$

Equation (A35) is the same as Eq. (A34) but with opposite signs. Therefore, we conclude that if $T = 1 \bmod 2$ then the expression in Eq. (A33) is upper bounded by a constant. Therefore, we can distinguish between acceptance and rejection of the original circuit with a fixed constant precision for both cases.

Now let us consider the hypothesis where $\|A\|_1 = \mathcal{O}(1/\text{poly log}(N))$. We pick some $c = \Theta(\text{poly log}(N))$, and consider the rescaled matrix $A' = A/c$. Repeating the proof steps, we see that the gap between the possible values of $[A'^{-1}]_{j,j} = c[A^{-1}]_{j,j}$ is $\Omega(c)$, while $\kappa_{A'} = \kappa_A$. Thus, the value $|\alpha_{x,1}|^2$ is still determinable via a constant-error solution to the inverse problem (or even via a solution with error $\mathcal{O}(c)$). The proof for the stronger hypothesis $\lambda_A = \mathcal{O}(1/\text{poly log}(N))$ follows similarly. ■

Theorem A1. The problems $\text{LM-INVERSE}_{\|A\|_1}^{\text{SPARSEACCESS}}$ and $\text{LM-INVERSE}_{\lambda_A}^{\text{PAULIACCESS}}$ are BQP-HARD.

Proof. We employ the construction from Ref. [2] with some extra tweaks and consider as elemental gates the set $\{T, H\}$ containing the Toffoli and Hadamard gates, which correspond to matrices with real entries only.

Given a BQP circuit of T gates $U_{T-1} \dots U_0$ consider the following unitary clock construction

$$U = \sum_{t=0}^{T-1} \mathcal{T}_t \otimes U_t + \mathcal{T}_{t+T} \otimes \mathbb{1} + \mathcal{T}_{t+2T} \otimes U_{T-1-t}^\dagger, \tag{A36}$$

which essentially amounts to a clock construction over the circuit $U_0^\dagger \dots U_{T-1}^\dagger \mathbb{1} \dots \mathbb{1} U_{T-1} \dots U_0$ that computes and uncomputes the answer of the BQP circuit, but keeps the computed solution for T steps [93]. Now consider the matrix

$$A = \mathbb{1} - Ue^{-1/T}, \tag{A37}$$

which is invertible, since

$$\begin{aligned}
A|\mathbf{x}\rangle = 0 &\iff |\mathbf{x}\rangle = e^{-1/T}U|\mathbf{x}\rangle \\
&\implies \|\mathbf{x}\|_2 = e^{-1/T} \|\mathbf{x}\|_2 \\
&\implies |\mathbf{x}\rangle = \mathbf{0}. \tag{A38}
\end{aligned}$$

It holds that $\kappa_A = \mathcal{O}(T)$, and

$$A^{-1} = \sum_{k \geq 0} U^k e^{-k/T}. \tag{A39}$$

We can compute $A^{-1}|\text{step}_0\rangle|0\rangle$ straightforwardly observing that

$$U^k |\text{step}_0\rangle|0\rangle = \begin{cases} |\text{step}_{k \bmod 3T}\rangle \otimes U_{k \bmod T} \dots U_0|0\rangle & \text{for } 0 \leq k \bmod 3T < T, \\ |\text{step}_{k \bmod 3T}\rangle \otimes U_{T-1} \dots U_0|0\rangle & \text{for } T \leq k \bmod 3T < 2T, \\ |\text{step}_{k \bmod 3T}\rangle \otimes U_{(-1-k) \bmod T} \dots U_0|0\rangle & \text{for } 2T \leq k \bmod 3T < 3T. \end{cases} \tag{A40}$$

Thus, we have

$$A^{-1}|\text{step}_0\rangle|0\rangle = \sum_{k \geq 0} e^{-k/T} U^k |\text{step}_0\rangle|0\rangle \tag{A41}$$

$$= \sum_{\substack{k \geq 0 \\ 0 \leq k \bmod 3T < T}} e^{-k/T} |\text{step}_{k \bmod 3T}\rangle \otimes U_{k \bmod T} \dots U_0|0\rangle \tag{A42}$$

$$+ \sum_{\substack{k \geq 0 \\ T \leq k \bmod 3T < 2T}} e^{-k/T} |\text{step}_{k \bmod 3T}\rangle \otimes U_{T-1} \dots U_0 |0\rangle \quad (\text{A43})$$

$$+ \sum_{\substack{k \geq 0 \\ 2T \leq k \bmod 3T < 3T}} e^{-k/T} |\text{step}_{k \bmod 3T}\rangle \otimes U_{(-1-k) \bmod T} \dots U_0 |0\rangle. \quad (\text{A44})$$

We can simplify each summation further as

$$\sum_{\substack{k \geq 0 \\ 0 \leq k \bmod 3T < T}} e^{-k/T} |\text{step}_{k \bmod 3T}\rangle \otimes U_{k \bmod T} \dots U_0 |0\rangle \quad (\text{A45})$$

$$= \sum_{k=0}^{T-1} \sum_{m \geq 0} e^{-(k+3mT)/T} |\text{step}_k\rangle \otimes U_k \dots U_0 |0\rangle \quad (\text{A46})$$

$$= \sum_{k=0}^{T-1} e^{-k/T} |\text{step}_k\rangle \otimes U_k \dots U_1 |0\rangle \sum_{m \geq 0} (e^{-3})^m \quad (\text{A47})$$

$$= \frac{e^3}{e^3 - 1} \sum_{k=0}^{T-1} e^{-k/T} |\text{step}_k\rangle \otimes U_k \dots U_0 |0\rangle \quad (\text{A48})$$

and conclude that

$$A^{-1} |0\rangle = \frac{e^3}{e^3 - 1} \left(\sum_{k=0}^{T-1} e^{-k/T} |\text{step}_k\rangle \otimes U_k \dots U_0 |0\rangle \right. \quad (\text{A49})$$

$$+ \sum_{k=T}^{2T-1} e^{-k/T} |\text{step}_k\rangle \otimes U_{T-1} \dots U_0 |0\rangle \quad (\text{A50})$$

$$\left. + \sum_{k=2T}^{3T-1} e^{-k/T} |\text{step}_k\rangle \otimes U_{3T-1-k} \dots U_0 |0\rangle \right). \quad (\text{A51})$$

We can assume without loss of generality that the circuit $U_{T-1} \dots U_0$ uses the first qubit to store the acceptance probability $\alpha_{0,1}$, and that it remains in the $|0\rangle$ state until the last gate. Let $\pi = |1\rangle\langle 1| \otimes \mathbb{1}_{N/2}$. Therefore, $\langle 0|U^k \pi U^k |0\rangle = 0$ for $0 \leq k < T$ and $2T \leq k < 3T$, and

$$\begin{aligned} & \langle \text{step}_0 | \langle 0 | A^{-1} \pi A^{-1} | \text{step}_0 \rangle | 0 \rangle \\ &= \left(\frac{e^3}{e^3 - 1} \right)^2 |\alpha_{0,1}|^2 \sum_{k=T}^{2T-1} e^{-2k/T} \\ &= \left(\frac{e^3}{e^3 - 1} \right)^2 |\alpha_{0,1}|^2 \left(\frac{e^{-2}(1 - e^{-2})}{1 - e^{-2/T}} \right). \end{aligned} \quad (\text{A52})$$

Thus, we can distinguish between $|\alpha_{0,1}|^2 \geq \frac{2}{3}$ and $|\alpha_{0,1}|^2 \leq \frac{1}{3}$ by estimating $\langle \text{step}_0 | \langle 0 | A^{-1} \pi A^{-1} | \text{step}_0 \rangle | 0 \rangle$. Moreover, the gap between both cases is

$$\frac{1}{3} \left(\frac{e^3}{e^3 - 1} \right)^2 \frac{e^{-2}(1 - e^{-2})}{1 - e^{-2/T}} \geq \frac{1}{3} \left(\frac{e^3}{e^3 - 1} \right)^2 e^{-2}, \quad (\text{A53})$$

and thus, we can distinguish them with constant precision. If we consider the normalized measurement case, we can compute the norm of $\|A^{-1}|0\rangle\|$ as

$$\|A^{-1}|0\rangle\| = \frac{e^3}{e^3 - 1} \sqrt{\sum_{k=0}^{3T-1} e^{-2k/T}} = \frac{e^3}{e^3 - 1} \sqrt{\frac{1 - e^{-6}}{1 - e^{-2/T}}} \quad (\text{A54})$$

and then the measurement gives the result

$$\begin{aligned} \frac{\langle 0 | A^{-1} \pi A^{-1} | 0 \rangle}{\|A^{-1}|0\rangle\|^2} &= \frac{e^{-2}(1 - e^{-2})}{1 - e^{-6}} |\alpha_{0,1}|^2 \\ &= \frac{e^{-2}}{1 - e^{-2} - e^{-4}} |\alpha_{0,1}|^2, \end{aligned} \quad (\text{A55})$$

over which we can distinguish between acceptance and rejection of the original BQP circuit with constant precision.

Finally, note that A is not symmetric in general, but if we extend the system to have an extra qubit and we take

$$A' = \begin{bmatrix} 0 & A \\ A^\dagger & 0 \end{bmatrix}, \quad (\text{A56})$$

then we have

$$(A')^{-1} = \begin{bmatrix} 0 & (A^\dagger)^{-1} \\ A^{-1} & 0 \end{bmatrix} \quad (\text{A57})$$

and A' is symmetric, since $A^\dagger = A^T$ due to our choice of elemental gates (Toffoli and Hadamard). Moreover, $(\langle 0 | \langle 0 | A'^{-1} \pi A'^{-1} | 0 \rangle | 0 \rangle) / (\|A'^{-1}|0\rangle\|^2) = (\langle 0 | A^{-1} \pi A^{-1} | 0 \rangle) / (\|A^{-1}|0\rangle\|^2)$ and the local measurement has to be performed on the second qubit, slightly different of our previous convention for the local measurement problem.

It is possible to build efficient sparse access for A' , and moreover, it is Pauli sparse. Finally, if we take $A'' = A'/2$ or rather $A'' = A'/\lambda_{A'}$ we obtain each of the desired reductions. ■

-
- [1] D. Janzing and P. Wocjan, A simple PromiseBQP-complete matrix problem, *Theory Comput.* **3**, 61 (2007).
 - [2] A. W. Harrow, A. Hassidim, and S. Lloyd, Quantum algorithm for linear systems of equations, *Phys. Rev. Lett.* **103**, 150502 (2009).
 - [3] D. Nagaj, Fast universal quantum computation with railroad-switch local Hamiltonians, *J. Math. Phys.* **51**, 062201 (2010).
 - [4] A. Gilyén, Y. Su, G. Hao Low, and N. Wiebe, in *Proceedings of the 51st ACM Symposium on the Theory of Computing (STOC)* (2019), pp. 193–204.
 - [5] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, Grand unification of quantum algorithms, *Phys. Rev. X* **2**, 040203 (2021).
 - [6] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, *Quantum* **3**, 163 (2019).
 - [7] Y. Dong, X. Meng, K. B. Whaley, and L. Lin, Efficient phase-factor evaluation in quantum signal processing, *Phys. Rev. A* **103**, 042419 (2021).
 - [8] T. d. L. Silva, L. Borges, and L. Aolita, Fourier-based quantum signal processing, *arXiv:2206.02826*.
 - [9] D. Motlagh and N. Wiebe, Generalized quantum signal processing, *PRX Quantum* **5**, 020368 (2024).
 - [10] N.-H. Chia, A. Gilyén, T. Li, H.-H. Lin, E. Tang, and C. Wang, in *Proceedings of the 52nd ACM Symposium on the Theory of Computing (STOC)* (2020), pp. 387–400.
 - [11] S. Gharibian and F. Le Gall, in *Proceedings of the 54th ACM Symposium on the Theory of Computing (STOC)* (2022), pp. 19–32.
 - [12] J. Haah, Product decomposition of periodic functions in quantum signal processing, *Quantum* **3**, 190 (2019).
 - [13] L. Lin and Y. Tong, Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers, *PRX Quantum* **3**, 010318 (2022).
 - [14] G. Wang, D. S. França, R. Zhang, S. Zhu, and P. D. Johnson, Quantum algorithm for ground state energy estimation using circuit depth with exponentially improved dependence on precision, *Quantum* **7**, 1167 (2023).
 - [15] G. Wang, D. S. França, G. Rendon, and P. D. Johnson, Efficient ground-state-energy estimation and certification on early fault-tolerant quantum computers, *Phys. Rev. A* **111**, 012426 (2025).
 - [16] D. An, J.-P. Liu, and L. Lin, Linear combination of Hamiltonian simulation for nonunitary dynamics with optimal state preparation cost, *Phys. Rev. Lett.* **131**, 150603 (2023).
 - [17] D. An, A. M. Childs, and L. Lin, Quantum algorithm for linear non-unitary dynamics with near-optimal dependence on all parameters, *arXiv:2312.03916*.
 - [18] G. H. Low and Y. Su, Quantum eigenvalue processing, *arXiv:2401.06240* [quant-ph].
 - [19] X.-M. Zhang and X. Yuan, On circuit complexity of quantum access models for encoding classical data, *arXiv:2311.11365*.
 - [20] E. Campbell, Random compiler for fast Hamiltonian simulation, *Phys. Rev. Lett.* **123**, 070503 (2019).
 - [21] K. Wan, M. Berta, and E. T. Campbell, Randomized quantum algorithm for statistical phase estimation, *Phys. Rev. Lett.* **129**, 030503 (2022).
 - [22] S. Wang, S. McArdle, and M. Berta, Qubit-efficient randomized quantum algorithms for linear algebra, *PRX Quantum* **5**, 020324 (2024).
 - [23] K. Nakaji, M. Bagherimehrab, and A. Aspuru-Guzik, qSWIFT: High-order randomized compiler for Hamiltonian simulation, *arXiv:2302.14811*.
 - [24] A. Montanaro and C. Shao, in *Proceedings of the 56th ACM Symposium on the Theory of Computing (STOC)* (2024), pp. 573–584.
 - [25] E. Tang, Dequantizing algorithms to understand quantum advantage in machine learning, *Nat. Rev. Phys.* **4**, 692 (2022).
 - [26] D. Aharonov, I. Arad, and T. Vidick, Guest column: The quantum PCP conjecture, *ACM Sigact News* **44**, 47 (2013).
 - [27] S. Sachdeva and N. K. Vishnoi, Faster algorithms via approximation theory, *Found. Trends Theor. Comput. Sci.* **9**, 125 (2014).
 - [28] A. M. Childs, R. Kothari, and R. D. Somma, Quantum algorithm for systems of linear equations with exponentially improved dependence on precision, *SIAM J. Comput.* **46**, 1920 (2017).
 - [29] A. Tosta, T. d. L. Silva, G. Camilo, and L. Aolita, Randomized semi-quantum matrix processing, *arXiv:2307.11824*.
 - [30] S. Apers and L. Miclo, Quantum walks, the discrete wave equation and Chebyshev polynomials, *arXiv:2402.07809*.
 - [31] S. Apers, S. Sen, and D. Szabó, A (simple) classical algorithm for estimating Betti numbers, *arXiv:2211.09618*.
 - [32] R. P. Feynman, Quantum mechanical computers, *Opt. News* **11**, 11 (1985).
 - [33] This also extends to functions approximated by a polynomial of said degree.
 - [34] E. Tang, in *Proceedings of the 51st ACM Symposium on the Theory of Computing (STOC)* (2019), pp. 217–228.
 - [35] E. Tang, Quantum principal component analysis only achieves an exponential speedup because of its state preparation assumptions, *Phys. Rev. Lett.* **127**, 060503 (2021).
 - [36] A. Gilyén, Z. Song, and E. Tang, An improved quantum-inspired algorithm for linear regression, *Quantum* **6**, 754 (2022).
 - [37] C. Shao and A. Montanaro, Faster quantum-inspired algorithms for solving linear systems, *ACM Trans. Quantum Comput.* **3**, 20 (2022).
 - [38] This problem consists of finding the ground state energy of a local Hamiltonian given a “guiding” vector with $\Omega(1/\text{poly}(N))$ overlap with the corresponding eigenspace.
 - [39] N. Bansal, S. Bravyi, and B. M. Terhal, Classical approximation schemes for the ground-state energy of quantum and classical Ising spin Hamiltonians on planar graphs, *Quantum Inf. Comput.* **9**, 701 (2009).
 - [40] The precise problems to which our computational hardness results will directly apply are actually the promise problem version of these estimation tasks, i.e., deciding whether the target quantity is above or below a value range, instead of directly estimating it. Both problems are intimately connected: for example, using an algorithm that decides given g whether $A_{j,j}^m \geq g$ it is possible to approximate the value

- A_{jj}^m by doing binary search on the value g in the range $[-\|A\|^m, \|A\|^m]$. The formal promise problems considered for hardness analysis are defined in Sec. IV for each function.
- [41] Y. Dong, L. Lin, and Y. Tong, Ground-state preparation and energy estimation on early fault-tolerant quantum computers via quantum eigenvalue transformation of unitary matrices, *PRX Quantum* **3**, 040305 (2022).
 - [42] J. H. Reif, Logarithmic depth circuits for algebraic functions, *SIAM J. Comput.* **15**, 231 (1986).
 - [43] C. H. Bennett, Logical reversibility of computation, *IBM J. Res. Dev.* **17**, 525 (1973).
 - [44] A. Shpilka and A. Yehudayoff, Arithmetic circuits: A survey of recent results and open questions, *Found. Trends Theor. Comput. Sci.* **5**, 207 (2010).
 - [45] S. Wang, X. Li, W. J. B. Lee, S. Deb, E. Lim, and A. Chattopadhyay, A comprehensive study of quantum arithmetic circuits, [arXiv:2406.03867](#).
 - [46] Z. Zhang, Q. Wang, and M. Ying, Parallel quantum algorithm for Hamiltonian simulation, *Quantum* **8**, 1228 (2024).
 - [47] S. McArdle, S. Endo, Alán Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, *Rev. Mod. Phys.* **92**, 015003 (2020).
 - [48] A. Lucas, Ising formulations of many NP problems, *Front. Phys.* **2**, 74887 (2014).
 - [49] J. Kempe, A. Kitaev, and O. Regev, The complexity of the local Hamiltonian problem, *SIAM J. Comput.* **35**, 1070 (2006). Earlier version in *FSTTCS'04*.
 - [50] S. Aaronson and D. Gottesman, Improved simulation of stabilizer circuits, *Phys. Rev. A* **70**, 052328 (2004).
 - [51] C. Gidney, Stim: A fast stabilizer circuit simulator, *Quantum* **5**, 497 (2021).
 - [52] P. Rall, D. Liang, J. Cook, and W. Kretschmer, Simulation of qubit quantum circuits via Pauli propagation, *Phys. Rev. A* **99**, 062337 (2019).
 - [53] A. Angrisani, A. Schmidhuber, M. S. Rudolph, M. Cerezo, Z. Holmes, and H.-Y. Huang, Classically estimating observables of noiseless quantum circuits, *Phys. Rev. Lett.* **135**, 170602 (2025).
 - [54] C. Cirstoiu, A Fourier analysis framework for approximate classical simulations of quantum circuits. [arXiv:2410.13856](#).
 - [55] S. Lerch, R. Puig, M.S. Rudolph, A. Angrisani, T. Jones, M. Cerezo, S. Thanasilp, and Z. Holmes, Efficient quantum-enhanced classical simulation for patches of quantum landscapes, [arXiv:2411.19896](#).
 - [56] S. Bravyi, D. Browne, P. Calpin, E. Campbell, D. Gosset, and M. Howard, Simulation of quantum circuits by low-rank stabilizer decompositions, *Quantum* **3**, 181 (2019).
 - [57] B. Coecke and R. Duncan, in *International Colloquium on Automata, Languages, and Programming* (Springer, 2008), pp. 298–310.
 - [58] O. Koska, M. Baboulin, and A. Gazda, A tree-approach Pauli decomposition algorithm with application to quantum computing, [arXiv:2403.11644](#).
 - [59] L. Hantzko, L. Binkowski, and S. Gupta, Tensorized Pauli decomposition algorithm, *Phys. Scr.* **99**, 085128 (2024).
 - [60] T. N. Georges, B. K. Berntson, C. Sünderhauf, and A. V. Ivanov, Pauli decomposition via the fast Walsh-Hadamard transform, *New J. Phys.* **27**, 033004 (2025).
 - [61] The standard way [4, Lemma 48] of preparing a block encoding of an s -sparse matrix A with largest matrix entry magnitude ≤ 1 given through the sparse access model returns a subnormalized block encoding of A/s . To the authors' knowledge, there is not currently a known method to generically obtain a non-normalized block encoding when starting from sparse access.
 - [62] D. Janzing and P. Wocjan, BQP-complete problems concerning mixing properties of classical random walks on sparse graphs, [arXiv:quant-ph/0610235](#).
 - [63] The notion of BQP-completeness as we described it is based on Karp-reductions, see Definition 5 for definitions. Meanwhile, a polynomial-time Turing reduction from problem A to B consists of an algorithm that is able to decide A in polynomial-time using an oracle able to solve problem B . In other words, results (C.2) show that the problem of computing Chebyshev polynomials is BQP-COMplete under Turing reductions.
 - [64] R. P. Feynman, Simulating physics with computers, *Int. J. Theor. Phys.* **21**, 467 (1982).
 - [65] A. Peres, Reversible logic and quantum computers, *Phys. Rev. A* **32**, 3266 (1985).
 - [66] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian dynamics with a truncated Taylor series, *Phys. Rev. Lett.* **114**, 090502 (2015).
 - [67] More precisely, given an algorithm that computes a $(k\varepsilon)$ -approximation of $\langle i|A^{-1}|j\rangle$, we can obtain an ε -approximation of the same value by using the same algorithm to compute a $(k\varepsilon)$ -approximation of $\langle i|(A/k)^{-1}|j\rangle = k\langle i|A^{-1}|j\rangle$ and then dividing that approximation by k .
 - [68] S. E. Venegas-Andraca, Quantum walks: A comprehensive review, *Quantum Inf. Process.* **11**, 1015 (2012).
 - [69] K. Seki and S. Yunoki, Quantum power method by a superposition of time-evolved states, *PRX Quantum* **2**, 010333 (2021).
 - [70] T. A. Bespalova and O. Kyriienko, Hamiltonian operator approximation for energy measurement and ground-state preparation, *PRX Quantum* **2**, 030318 (2021).
 - [71] W. Kirby, M. Motta, and A. Mezzacapo, Exact and efficient Lanczos method on a quantum computer, *Quantum* **7**, 1018 (2023).
 - [72] T. O'Leary, L. W. Anderson, D. Jaksch, and M. Kiffner, Partitioned quantum subspace expansion, *Quantum* **9**, 1726 (2025).
 - [73] The “non-promise” problems, known commonly as decision problems, consist of the subset of promise problems such that $\Pi_{\text{yes}} \cup \Pi_{\text{no}} = \{0, 1\}^*$.
 - [74] Observe that we are defining BQP as a class of promise problems. Formally, one might denote this class as PROMISEBQP, while reserving BQP for the class of non-promise problems. Nonetheless, using the term BQP to refer directly to the promise class is common in the literature, and we follow this convention.
 - [75] D. Aharonov and A. Ta-Shma, Adiabatic quantum state generation, *SIAM J. Comput.* **37**, 47 (2007). Earlier version in *STOC'03*.
 - [76] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, *Commun. Math. Phys.* **270**, 359 (2007).
 - [77] We say that a function g ε -approximates a function f in domain $\mathcal{D}$ when for all $x \in \mathcal{D}$ it holds that $|f(x) - g(x)| \leq \varepsilon$.

- [78] A. Y. Kitaev, Quantum measurements and the Abelian Stabilizer Problem, [arXiv:quant-ph/9511026](#).
- [79] Michael A. Nielsen and Isaac L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, 2000).
- [80] L. Lin, Lecture notes on quantum algorithms for scientific computation, [arXiv:2201.08309](#).
- [81] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, in *Quantum Computation and Quantum Information: A Millennium Volume*, Contemporary Mathematics Series (AMS, 2002), Vol. 305, pp. 53–74.
- [82] G. H. Low and I. L. Chuang, Optimal Hamiltonian simulation by quantum signal processing, [Phys. Rev. Lett. **118**, 010501 \(2017\)](#).
- [83] M. Berestycki, Mixing times of markov chains: techniques and examples. Lecture Notes (2016).
- [84] From now on, we omit the dependence on N of η and simply write η to denote $\eta(N)$.
- [85] Technically, the decision version of this problem is BQP-COMPLETE.
- [86] This follows straightforwardly from error propagation in multiplication.
- [87] We do not explicitly prove this fact, but it is a consequence of the proof from Theorem 8.
- [88] D. Aharonov, V. Jones, and Z. Landau, “in *Proceedings of the 38th ACM Symposium on the Theory of Computing (STOC)* (2006), pp. 427–436.
- [89] A. B. Aleksandrov and V. V. Peller, Estimates of operator moduli of continuity, [J. Funct. Anal. **261**, 2741 \(2011\)](#).
- [90] F. Verstraete and J. I. Cirac, Matrix product states represent ground states faithfully, [Phys. Rev. B **73**, 094423 \(2006\)](#).
- [91] D. Gross, Recovering low-rank matrices from few coefficients in any basis, [IEEE Trans. Inf. Theory **57**, 1548 \(2011\)](#).
- [92] A. Markov, Sur une question posée par Mendeleieff, *Bull. Acad. Sci. St. Petersburg* **62**, 1 (1889).
- [93] This trick is known as *idling*.