Electronic Colloquium on Computational Complexity, Report No. 88 (2020)

Eliminating Intermediate Measurements in Space-Bounded Quantum Computation

Bill Fefferman

Department of Computer Science, The University of Chicago

Zachary Remscrim*

Department of Mathematics, MIT

Abstract

A foundational result in the theory of quantum computation known as the "principle of safe storage" shows that it is always possible to take a quantum circuit and produce an equivalent circuit that makes all measurements at the end of the computation. While this procedure is time efficient, meaning that it does not introduce a large overhead in the number of gates, it uses extra ancillary qubits and so is not generally space efficient. It is quite natural to ask whether it is possible to defer measurements to the end of a quantum computation without increasing the number of ancillary qubits.

We give an affirmative answer to this question by exhibiting a procedure to eliminate all intermediate measurements that is simultaneously space-efficient and time-efficient. A key component of our approach, which may be of independent interest, involves showing that the well-conditioned versions of many standard linear-algebraic problems may be solved by a quantum computer in less space than seems possible by a classical computer.

1 Introduction

Quantum computation has the potential to obtain dramatic speedups for important problems such as quantum simulation (see, e.g., [19, 28]) and integer factorization [43]. While fully scalable, fault-tolerant quantum computers may still be far from fruition, we have now entered an exciting period in which impressive but resource constrained quantum experiments are being implemented in many academic and industrial labs. As the field transitions from "proof of principle" demonstrations of provable quantum advantage to solving useful problems on near-term experiments, it is particularly critical to characterize the algorithmic power of feasible models of quantum computations that have restrictive resources such as "time" (i.e., the number of gates in the circuit) and "space" (i.e., the number of qubits on which the circuit operates) and to understand how these resources can be traded-off.

A foundational question in this area asks if it is possible to *space-efficiently* defer intermediate measurements in a quantum computation (see e.g., [18, 23, 31, 35, 45, 49–51]). While a classic result known as the "principle of safe storage" states that it is always possible to *time-efficiently* defer

*Corresponding author, remscrim@mit.edu

ISSN 1433-8092

intermediate measurements to the end of a computation (see e.g., [2,33]), this procedure uses extra ancilla qubits and so is not generally space-efficient. More specifically, if a quantum circuit Q acts on s qubits and performs m intermediate measurements, the circuit Q' constructed using this principle operates on $s + poly(m)$ qubits; if, for example, $s = O(\log t)$ and $m = \Theta(t)$, this entails an *exponential blowup* in the amount of needed *space*.

Our main result solves this problem. We show that every problem solvable with a quantum computation that acts on a designated number of qubits can also be solved by a "unitary" quantum algorithm that uses the same space and makes all measurements at the end of the computation. Stated more formally in the language of complexity theory, we let $\mathsf{BQ_USPACE}(s(n))$ (resp. $\mathsf{BQSPACE}(s(n))$) denote the class of promise problems recognizable with two-sided bounded-error by a uniform family of unitary (resp. general) quantum circuits, where, for each input of length n, there is a corresponding circuit that operates on $O(s(n))$ qubits and has $2^{O(s(n))}$ gates. Note that it is standard to require that the running time of a computation is at most exponential in its space-bound; see, for instance, [31,49,51] for the importance of this restriction in quantum space-bounded computation, as well as [39] for the importance of the analogous restriction in probabilistic space-bounded computation. Furthermore, let $\mathsf{QMASPACE}(s(n))$ denote those promise problems recognized by a quantum Merlin-Arthur protocol that operates in space $O(s(n))$ and time $2^{O(s(n))}$.

Our main result is:

Theorem 1. *For any space-constructible function $s : \mathbb{N} \to \mathbb{N}$, where $s(n) = \Omega(\log n)$, we have*

$$\mathsf{BQ_USPACE}(s(n)) = \mathsf{BQSPACE}(s(n)) = \mathsf{QMASPACE}(s(n)).$$

Remark. To the best of our knowledge, the containment $\mathsf{BPSPACE}(s(n)) \subseteq \mathsf{BQ_USPACE}(s(n))$ was not known to hold, where $\mathsf{BPSPACE}(s(n))$ denotes the analogously defined class of language recognizable by a *probabilistic* algorithm in space $O(s(n))$ (and time $2^{O(s(n))}$); see, for instance, [31,50] for previous discussion of this question. As one would expect quantum computation to generalize probabilistic computation, the lack of a proof of this containment was unfortunate. Since it is clear that $\mathsf{BPSPACE}(s(n)) \subseteq \mathsf{BQSPACE}(s(n))$, we have as a corollary of Theorem 1, that $\mathsf{BPSPACE}(s(n)) \subseteq \mathsf{BQ_USPACE}(s(n))$, resolving this question.

Before proceeding further, it is worthwhile to briefly discuss why it is desirable to be able to eliminate intermediate measurements. Firstly, quantum measurements are a natural resource, much as time and space are; in addition to the general desirability of using as few resources as possible in any sort of computational task, it is especially desirable to avoid intermediate measurements, due to the technical challenges involved in implementing such measurements and resetting such qubits to their initial states (for a discussion of these issues from an experimental perspective see, e.g., [14]). Secondly, unitary computations are *reversible*. The ability to "undo" a unitary subroutine, by running that subroutine in reverse, is used routinely in the design and analysis of quantum algorithms (see, for instance, [6,17,18,30,32,44,53]). Moreover, reversible computations may be performed without generating heat [27]. Thirdly, by demonstrating that unitary quantum space and general quantum space are (polynomially) equivalent in power, we show that the definition of quantum space is quite *robust*, as allowing intermediate measurements, or even general quantum operations, does not affect the definition of $\mathsf{BQ_USPACE}(s(n))$.

We also study the one-sided (bounded-error and unbounded-error) analogues of the aforementioned two-sided bounded-error space-bounded quantum complexity classes. We show the following results (see Section 2.2 for definitions of the complexity classes appearing in the following theorems).

Theorem 2. *For any space-constructible function $s : \mathbb{N} \to \mathbb{N}$, where $s(n) = \Omega(\log n)$, we have*

$$\mathsf{RQMASPACE}(s(n)) = \mathsf{RQ_USPACE}(s(n)) \subseteq \mathsf{RQSPACE}(s(n)) \subseteq \mathsf{coQMASPACE_1}(s(n)).$$

Theorem 3. *For any space-constructible function $s : \mathbb{N} \to \mathbb{N}$, where $s(n) = \Omega(\log n)$, we have*

$$\mathsf{NQMASPACE}(s(n)) = \mathsf{NQ_USPACE}(s(n)) = \mathsf{NQSPACE}(s(n))$$

$$= \mathsf{coPreciseQMA_1SPACE}(s(n)) = \mathsf{coC_=SPACE}(s(n)).$$

1.1 Exact and Approximate Linear Algebra

In order to prove the theorems stated above, we study the complexity of solving various standard linear-algebraic problems, both exactly and approximately. Let intDET denote the problem of computing the determinant of an $n \times n$ integer-valued matrix, and, following its original definition by Cook [12], let DET^* denote the class of problems NC^1 (Turing) reducible to intDET (the definition of this class is somewhat delicate, see [4,5,29] for further discussion). Let $\mathsf{BQ_UL} = \mathsf{BQ_USPACE}(\log(n))$, $\mathsf{BQL} = \mathsf{BQSPACE}(\log(n))$, and $\mathsf{BPL} = \mathsf{BPSPACE}(\log(n))$ denote the bounded-error quantum and probabilistic logspace classes. Finally, let $\mathsf{PQ_UL}$, PQL, and PL denote the corresponding unbounded-error classes. Before our work, the following relationships were known [12, 49, 51]:

$$\mathsf{NC}^1 \subseteq \mathsf{L} \subseteq \mathsf{RL} \subseteq \mathsf{BPL} \subseteq \mathsf{BQL} \subseteq \mathsf{PL} = \mathsf{PQ_UL} = \mathsf{PQL} \subseteq \mathsf{DET}^* \subseteq \mathsf{NC}^2 \subseteq \mathsf{L}^2,$$

$$\mathsf{RL} \subseteq \mathsf{NL} \subseteq \mathsf{PL}, \quad \text{and} \quad \mathsf{L} \subseteq \mathsf{BQ_UL} \subseteq \mathsf{BQL}.$$

Many natural linear-algebraic problems are complete for DET^*, including intDET, $\mathsf{intMATINV}$ (the problem of computing a single entry of the inverse of an invertible integer-valued matrix), and $\mathsf{intMATPOW}$ (the problem of computing a single entry of the m^{th} power of an $n \times n$ integer-valued matrix, for $m = \mathit{poly}(n)$). It seems rather unlikely that any such DET^*-complete problem is in the class BQL, as this would imply $\mathsf{BQL} = \mathsf{DET}^*$, and, therefore, $\mathsf{NL} \subseteq \mathsf{BQL}$. We next consider the problem of *approximating* the answer to these, and other related, linear-algebraic problems. In particular, let *poly*-conditioned-MATINV denote the promise problem of approximating (to additive $1/\mathit{poly}(n)$ accuracy) a single entry of the inverse of an $n \times n$ matrix A with *condition number* $\kappa(A) = \mathit{poly}(n)$ (see Section 3 for a precise definition of this problem). Ta-Shma [45], building on the landmark result of Harrow, Hassidim, and Lloyd [20], showed *poly*-conditioned-$\mathsf{MATINV} \in \mathsf{BQL}$; Fefferman and Lin [18] improved upon this result by showing that *poly*-conditioned-MATINV is, in fact, $\mathsf{BQ_UL}$-complete, thereby exhibiting the first known natural $\mathsf{BQ_UL}$-complete (promise) problem. In Section 3.1, we extend this result by showing that the *poly*-conditioned versions of many standard DET^*-complete problems are $\mathsf{BQ_UL}$-complete.

Theorem 4. *Each poly-conditioned promise problem given in Definition 15 is* $\mathsf{BQ_UL}$*-complete.*

This is an interesting result, in and of itself, as it demonstrates an intriguing relationship between $\mathsf{BQ_UL}$ and DET^*. Moreover, this result will then allow us to prove Theorem 1 and the other main results of this paper. In particular, in Section 3.2, we will show that the problem *poly*-conditioned-$\mathsf{ITMATPROD}$ is BQL-hard (again, see Section 3 for a precise definition of this problem). By the preceding theorem, *poly*-conditioned-$\mathsf{ITMATPROD} \in \mathsf{BQ_UL}$, which implies $\mathsf{BQL} = \mathsf{BQ_UL}$; Theorem 1, which states the more general equivalence for any larger (space-constructible) space bound, then follows from a standard padding argument. In Section 3.3, we continue the study of *fully logarithmic approximation schemes*, initiated by Doron and Ta-Shma [16], and show that the BQL vs. BPL question is equivalent to several distinct questions involving the relative power of quantum and probabilistic fully logarithmic approximation schemes.

Consider $\kappa(n)$-conditioned-DET, the problem of approximating, to within a *multiplicative* factor $(1 + 1/\mathit{poly}(n))$, the determinant of an $n \times n$ with condition number $\kappa(n)$. Boix-Adserà, Eldar, and

Mehraban [8] recently showed $\kappa(n)$-conditioned-$\mathsf{DET} \in \mathsf{DSPACE}(\log(n)\log(\kappa(n))poly(\log\log n))$. Furthermore, they raised the following question: is *poly*-conditioned-DET BQL-complete? As an immediate consequence of Theorem 4, we answer their question in the affirmative.

Proposition 5. *poly-conditioned-*DET *is* $\mathsf{BQ_UL}$*-complete (and, therefore,* BQL*-complete)*

To briefly explain the significance of the question posed by Boix-Adserà, Eldar, and Mehraban [8], note that their result shows *poly*-conditioned-$\mathsf{DET} \in \mathsf{DSPACE}(\log^2(n)poly(\log\log n))$. Therefore, as *poly*-conditioned-DET is BQL-complete, the statement $\mathsf{BQL} \subseteq \mathsf{DSPACE}(\log^{2-\epsilon} n)$ would follow from either a small improvement in their result (i.e., proving a stronger upper bound on the needed amount of deterministic space in terms of $\kappa(n)$) or from a small improvement in our result (i.e., proving $\kappa(n)$-conditioned-DET remains BQL-hard for *subpolynomial* $\kappa(n)$). Recall that the strongest currently known "dequantumization" result of this type is the classic result of Watrous [51], which states $\mathsf{BQL} \subseteq \mathsf{DSPACE}(\log^2 n)$ (cf. the strongest currently known "derandomization" result of this type, given by Saks and Zhou [40], which states $\mathsf{BPL} \subseteq \mathsf{DSPACE}(\log^{\frac{3}{2}} n)$). In particular, if $\mathsf{BQL} \not\subseteq \mathsf{DSPACE}(\log^{2-\epsilon} n)$, $\forall \epsilon > 0$, then both our result and the result of Boix-Adserà, Eldar, and Mehraban are essentially optimal.

In Section 4, we study well-conditioned versions of the "standard" $\mathsf{C_=L}$-complete problems. We will observe that the relationship of these problems to the *one*-sided error versions of space-bounded quantum complexity classes is very much analogous to the relationship, discussed earlier, of the well-conditioned versions of the "standard" DET^*-complete problems to the *two*-sided error versions of space-bounded quantum complexity classes. This enables us to prove Theorem 2 and Theorem 3, which concern the relative power of unitary and general quantum space in the one-sided error cases.

In Section 5, we establish the $\mathsf{BQ_UL}$-completeness of "scaled-down" versions of certain QMA-complete problems and certain $\mathsf{DQC1}$-complete [25] problems. We conclude by stating several open problems related to our work in Section 6.

2 Preliminaries

2.1 General Notation and Definitions

Let $\mathrm{Mat}(n) = \mathbb{C}^{n\times n}$ denote the set of all $n \times n$ complex matrices and let $\mathrm{Herm}(n) = \{A \in \mathrm{Mat}(n) : A = A^\dagger\}$ denote the set of all $n \times n$ Hermitian matrices. For $A \in \mathrm{Mat}(n)$, let $\sigma_1(A) \geq \cdots \geq \sigma_n(A) \geq 0$ denote its singular values and let $\lambda_1(A), \ldots, \lambda_n(A) \in \mathbb{C}$ denote its eigenvalues (with multiplicity); if $A \in \mathrm{Herm}(n)$, then $\lambda_j(A) \in \mathbb{R}, \forall j$, and we order the eigenvalues such that $\lambda_1(A) \geq \cdots \geq \lambda_n(A)$. Let I_n denote the $n \times n$ identity matrix, $\mathrm{Pos}(n) = \{A \in \mathrm{Herm}(n) : \lambda_n(A) \geq 0\}$ denote the $n \times n$ positive semidefinite matrices, $\mathrm{Proj}(n) = \{A \in \mathrm{Pos}(n) : A^2 = A\}$ denote the $n \times n$ projection matrices, $\mathrm{U}(n) = \{A \in \mathrm{Mat}(n) : AA^\dagger = I_n\}$ denote the $n \times n$ unitary matrices, and $\mathrm{Den}(n) = \{A \in \mathrm{Pos}(V) : \mathrm{tr}(A) = 1\}$ denote the $n \times n$ density matrices. Let $\mathbb{Q}[i]_n = \{\frac{r+ci}{d} : r, c, d \in \mathbb{Z}, |r|, |c|, |d| \leq 2^{O(n)}\}$ denote the set of all $O(n)$-bit Gaussian rationals and let $\widehat{\mathrm{Mat}}(n)$ (resp. $\widehat{\mathrm{Herm}}(n), \widehat{\mathrm{Pos}}(n)$, etc.) denote the subset of $\mathrm{Mat}(n)$ (resp. $\mathrm{Herm}(n), \mathrm{Pos}(n)$, etc.) which consist of those matrices whose entries are all $O(n)$-bit Gaussian rationals. For a sequence of matrices $A_1, \ldots, A_m$, and for indices j_1, j_2, where $1 \leq j_1 \leq j_2 \leq m$, let $A_{j_1,j_2} = \prod_{j=j_1}^{j_2} A_j$.

We consider parameterized promise problems of the form $\mathsf{P} = (\mathsf{P}_{n,f_1,\ldots,f_h})_{n\in\mathbb{N}}$, for functions $f_1, \ldots, f_h : \mathbb{N} \to \mathbb{R}$; $\mathsf{P}_{n,f_1,\ldots,f_h}$ consists of instances of size n which satisfy various conditions expressed in terms of $f_1(n), \ldots, f_h(n)$. For a promise problem P defined over some alphabet Σ, we, by slight abuse of notation, also write P to denote the subset of Σ^* that satisfies the promise; analogously, we write $\mathsf{P}_{n,f_1,\ldots,f_h}$ to denote those instances of size n that satisfy the promise. For $\langle X\rangle \in \mathsf{P}$, let $\mathsf{P}(\langle X\rangle) \in \{0,1\}$ denote the desired output on input X. We also use the notation $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0)$, where $\mathsf{P}_j = \{\langle X\rangle \in \mathsf{P} : \mathsf{P}(\langle X\rangle) = j\}$.

We say that $\mathsf{P} = (\mathsf{P}_{n,f_1,\ldots,f_h})_{n\in\mathbb{N}}$ is many-one reducible to $\mathsf{P}' = (\mathsf{P}'_{m,f'_1,\ldots,f'_{h'}})_{m\in\mathbb{N}}$ if $\exists p_0,\ldots,p_{h'}$, where each p_j is a real $(h+1)$-variate polynomial, such that $\forall n \in \mathbb{N}$, $\exists g_n : \mathsf{P}_{n,f_1,\ldots,f_h} \to \mathsf{P}'_{m,f'_1,\ldots,f'_{h'}}$ such that (1) $m = p_0(n, f_1(n),\ldots,f_h(n))$, (2) $f'_j(m) = p_j(n, f_1(n),\ldots,f_h(n))$, $\forall j$, and (3) such that $\mathsf{P}(\langle X\rangle) = \mathsf{P}'(g_n(\langle X\rangle))$, $\forall \langle X\rangle \in \mathsf{P}_{n,f_1,\ldots,f_h}$. If $(g_n)_{n\in\mathbb{N}}$ is computable in deterministic logspace (resp. uniform NC^1, uniform AC^0), we write $\mathsf{P} \leq^m_{\mathsf{L}} \mathsf{P}'$ (resp. $\mathsf{P} \leq^m_{\mathsf{NC}^1} \mathsf{P}'$, $\mathsf{P} \leq^m_{\mathsf{AC}^0} \mathsf{P}'$). For a complexity class $\mathcal{C}$, we say that P' is $\mathcal{C}$-complete if (1) $\mathsf{P}' \in \mathcal{C}$ and (2) $\mathsf{P} \leq^m_{\mathsf{L}} \mathsf{P}'$, $\forall \mathsf{P} \in \mathcal{C}$.

We assume that the reader has familiarity with quantum computation and the theory of quantum information; see, for instance, [24,33,54] for an introduction. A quantum system, on s qubits, that is in a *pure state* is described by a unit vector $|\psi\rangle$ in the 2^s-dimensional Hilbert space $\mathbb{C}^{2^s}$. A *mixed state* of the same system is described by some ensemble $\{(p_i, |\psi_i\rangle) : i \in I\}$, for some index set I, where $p_i \in [0,1]$ denotes the probability of the system being in the pure state $|\psi_i\rangle \in \mathbb{C}^{2^s}$, and $\sum_i p_i = 1$. This ensemble corresponds to the density matrix $A = \sum_i p_i |\psi_i\rangle\langle\psi_i| \in \mathrm{Den}(2^s)$. Of course, many distinct ensembles correspond to the density matrix A; however, all ensembles that correspond to a particular density matrix will behave the same, for our purposes (see, for instance, [33, Section 2.4] for a detailed discussion of this phenomenon).

Let $\mathrm{T}(n,m)$ denote the set of all *superoperators* of the form $\Phi : \mathrm{Mat}(n) \to \mathrm{Mat}(m)$ (i.e., Φ is a $\mathbb{C}$-linear map from the $\mathbb{C}$-vector space $\mathrm{Mat}(n)$ to the $\mathbb{C}$-vector space $\mathrm{Mat}(m)$). Let $\mathrm{T}(n) = \mathrm{T}(n,n)$ and let $\mathbb{1}_n \in \mathrm{T}(n)$ denote the identity operator. Consider some $\Phi \in \mathrm{T}(n,m)$. We say that Φ is *positive* if, $\forall A \in \mathrm{Pos}(n)$, we have $\Phi(A) \in \mathrm{Pos}(m)$. We say that Φ is *completely-positive* if $\Phi \otimes \mathbb{1}_r$ is positive, $\forall r \in \mathbb{N}$, where $\otimes$ denotes the tensor product. We say that Φ is *trace-preserving* if $\mathrm{tr}(\Phi(A)) = \mathrm{tr}(A)$, $\forall A \in \mathrm{Mat}(n)$. If Φ is both completely-positive and trace-preserving, then we say Φ is a *quantum channel*; let $\mathrm{Chan}(n,m) = \{\Phi \in \mathrm{T}(n,m) : \Phi \text{ is a quantum channel}\}$ denote the set of all such channels, and let $\mathrm{Chan}(n) = \mathrm{Chan}(n,n)$.

Let vec denote the usual vectorization map that takes a matrix $A \in \mathrm{Mat}(n)$ to the vector $\mathrm{vec}(A) \in \mathbb{C}^{n^2}$ consisting of the entries of A (in some fixed order). For $\Phi \in \mathrm{T}(n)$, let $K(\Phi) \in \mathrm{Mat}(n^2)$ denote the *natural representation* of Φ, which is defined to be the (unique) matrix for which $\mathrm{vec}(\Phi(A)) = K(\Phi)\mathrm{vec}(A)$, $\forall A \in \mathrm{Mat}(n)$.

2.2 Space-Bounded Quantum Computation

We now briefly recall the definitions of several needed space-bounded quantum complexity classes. We begin by noting that, in many of the previous papers that considered space-bounded quantum computation [23, 31, 35, 45, 49–52], the *quantum Turing machine* model was used to define the various complexity classes of interest. Arguably, this is the "natural" model to be used to define these classes. However, as the (equivalent) model of uniformly generated *quantum circuits* are, arguably, more familiar to quantum complexity theorists and physicists, we state these definitions using quantum circuits. We emphasize that all of the results in this paper apply to both the uniform quantum circuit model and the quantum Turing machine model.

Definition 6. A (unitary) *quantum circuit* is a sequence of quantum gates, each of which is a member of some fixed set of gates that is universal for quantum computation (e.g., {H,CNOT,T}). We say that a family of quantum circuits $\{Q_w : w \in \mathsf{P}\}$ is $\mathsf{DSPACE}(s(n))$-uniform if there is a deterministic Turing machine that, on any input $w \in \mathsf{P}$, runs in space $O(s(|w|))$ (and hence time $2^{O(s(|w|))}$), and outputs a description of Q_w.

Definition 7. Consider functions $c, k : \mathbb{N} \to [0,1]$ and $s : \mathbb{N} \to \mathbb{N}$, with $s(n) = \Omega(\log n)$, all of which are computable in $\mathsf{DSPACE}(s(n))$. Let $\mathsf{Q_U SPACE}(s(n))_{c(n),k(n)}$ denote the collection of all promise problems $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0)$ such that there is a $\mathsf{DSPACE}(s(n))$-uniform family of (unitary) quantum circuits $\{Q_w : w \in \mathsf{P}\}$, where Q_w acts on $h_w = O(s(|w|))$ qubits and has $2^{O(s(|w|))}$ gates, that has the following properties. The circuit Q_w is applied to h_w qubits that were initialized in the all-zeros state $|0^{h_w}\rangle$, after which the first qubit is measured in the standard basis. If the result is 1, then we say Q_w *accepts* w; if, instead, the result is 0, then we say Q_w *rejects* w. Let $\Pi_1 = |1\rangle\langle 1| \otimes I_{2^{h_w - 1}}$. We require that the following conditions hold:

Completeness: $w \in \mathsf{P}_1 \Rightarrow \Pr[Q_w \text{ accepts } w] = \|\Pi_1 Q_w |0^{h_w}\rangle\|^2 \geq c(|w|)$.

Soundness: $w \in \mathsf{P}_0 \Rightarrow \Pr[Q_w \text{ accepts } w] = \|\Pi_1 Q_w |0^{h_w}\rangle\|^2 \leq k(|w|)$.

Note that, in the preceding definitions, Q_w has $2^{O(s(|w|))}$ gates (this requirement is immediately forced by the uniformity condition). That is to say, in our definition of quantum space $s(n)$, we require that the computation also runs in time $2^{O(s(|w|))}$. We refer the reader to the excellent survey paper by Saks [39] for a thorough discussion of the desirability of requiring that space-bounded *probabilistic* computations run in time at most exponential in their space bound, as well as to, for instance, [31, 49, 51] for discussions of the analogous issue for *quantum* computations.

Definition 8. We then define unitary quantum space $s(n)$ for a variety of types of error, as follows.

Two-sided bounded-error: $\mathsf{BQ_U SPACE}(s(n)) = \mathsf{Q_U SPACE}(s(n))_{\frac{2}{3},\frac{1}{3}}$.

One-sided bounded-error: $\mathsf{RQ_U SPACE}(s(n)) = \mathsf{Q_U SPACE}(s(n))_{\frac{1}{2},0}$.

One-sided unbounded-error: $\mathsf{NQ_U SPACE}(s(n)) = \bigcup_{c:\mathbb{N}\to(0,1]} \mathsf{Q_U SPACE}(s(n))_{c(n),0}$.

Then, for $\mathsf{X} \in \{\mathsf{B}, \mathsf{R}, \mathsf{N}\}$, let $\mathsf{XQ_U L} = \mathsf{XQ_U SPACE}(\log n)$ denote *unitary quantum logspace* with error-bounds specified by the modifier X.

Note that the particular choice of universal gateset does not affect the definition of the two-sided bounded-error classes $\mathsf{BQ_U SPACE}(s(n))$ (in particular, of $\mathsf{BQ_U L}$) due to the space-efficient version [31] of the Solovay-Kitaev theorem; however, the definitions of the one-sided (bounded and unbounded) error classes are, potentially, gateset dependent. Furthermore, note that the class $\mathsf{BQ_U L}$ would remain the same if defined as $\mathsf{BQ_U L} = \mathsf{Q_U SPACE}(\log n)_{c(n),k(n)}$ for any c, k for which $c(n) = 1 - \frac{1}{poly(n)}$, $k(n) = \frac{1}{poly(n)}$, and $\exists q : \mathbb{N} \to \mathbb{N}_{>0}$, where $q(n) = poly(n)$, such that $c(n) - k(n) \geq \frac{1}{q(n)}$, $\forall n$ [17]; an analogous claim holds for $\mathsf{RQ_U L}$ [49].

We next consider general space-bounded quantum computation. Most basically, we wish to define a model of quantum computation that allows *intermediate quantum measurements*. That is to say, rather than considering a purely unitary quantum computation in which a single measurement is performed at the end of the computation, we now allow measurements to be performed throughout the computation, and for the results of those measurements to be used to control the computation. As we wish for our main result (the fact that unitary and general space-bounded quantum computation are equivalent in power) to be as strong as possible, we wish to define a model of general space-bounded quantum computation that is as powerful as possible. To that end, we will define a space-bounded variant of the general quantum circuit model considered in the classic paper of Aharonov, Kitaev, and Nisan [2], in which gates are now arbitrary *quantum channels*.

Definition 9. A *general quantum circuit* on h qubits is a sequence $\Phi = (\Phi_1, \ldots, \Phi_t)$ of quantum channels, where each $\Phi_j \in \mathrm{Chan}(2^h)$. By slight abuse of notation, we use Φ to denote the element $\Phi_t \circ \cdots \circ \Phi_1 \in \mathrm{Chan}(2^h)$ obtained by composing the individual gates of the circuit in order. We say that a family of general quantum circuits $\{\Phi_w = (\Phi_{w,1}, \ldots, \Phi_{w,t_w}) : w \in \mathsf{P}\}$ is $\mathsf{DSPACE}(s(n))$-uniform if there is a deterministic Turing machine that, on any input $w \in \mathsf{P}$, runs in space $O(s(|w|))$ (and hence time $2^{O(s(|w|))}$), and outputs a description of Φ_w; to be precise, a description of Φ_w consists of the entries of each $K(\Phi_{w,j})$, where we require that $K(\Phi_{w,j}) \in \widehat{\mathrm{Mat}}(2^{2h})$.

Note that the operation of applying a unitary transformation is a special case of a quantum channel, and so the general quantum circuit model extends the ordinary (unitary) quantum circuit model. Moreover, note that the process of performing any (partial or full) quantum measurement in the computational basis is described by a quantum channel and that the form of the preceding definition allows the results of intermediate measurements to be used to control which operations are applied at later stages of the computation (this can be accomplished by using a subset of the qubits as classical bits to store the results of earlier measurements, thereby making these results available to gates that appear later in the computation). It is necessary to establish some reasonable restriction on the complexity of computing a description of each gate of the circuit, as we do not wish to unreasonably increase the power of the model by allowing, for example, non-computable numbers to be used in defining each gate (see, for instance, [31, 36, 37, 51] for further discussion of this issue).

Definition 10. Consider functions $c, k : \mathbb{N} \to [0, 1]$ and $s : \mathbb{N} \to \mathbb{N}$, with $s(n) = \Omega(\log n)$, all of which are computable in $\mathsf{DSPACE}(s(n))$. Let $\mathsf{QSPACE}(s(n))_{c(n),k(n)}$ denote the collection of all promise problems $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0)$ such that there is a $\mathsf{DSPACE}(s(n))$-uniform family of general quantum circuits $\{\Phi_w : w \in \mathsf{P}\}$, where Φ_w acts on $h_w = O(s(|w|))$ qubits and has $2^{O(s(|w|))}$ gates, that has the following properties. The circuit Φ_w is applied to h_w qubits that were initialized in the all-zeros state $|0^{h_w}\rangle$, after which the first qubit is measured in the standard basis. If the result is 1, then Φ_w *accepts* w; otherwise, Φ_w *rejects* w. Let $\mathbb{1}_1 \otimes \mathrm{tr} \in \mathrm{Chan}(2^{h_w}, 2)$ denote the operation in which all qubits except the first qubit are *traced out*.

Completeness: $w \in \mathsf{P}_1 \Rightarrow \Pr[\Phi_w \text{ accepts } w] = |\langle 1|(\mathbb{1}_1 \otimes \mathrm{tr})(\Phi_w(|0^{h_w}\rangle\langle 0^{h_w}|))|1\rangle| \geq c(|w|)$.

Soundness: $w \in \mathsf{P}_0 \Rightarrow \Pr[\Phi_w \text{ accepts } w] = |\langle 1|(\mathbb{1}_1 \otimes \mathrm{tr})(\Phi_w(|0^{h_w}\rangle\langle 0^{h_w}|))|1\rangle| \leq k(|w|)$.

We note that this general quantum circuit model is equivalent to the space-bounded (general) quantum Turing machine model of Watrous [51] (modulo a small issue concerning the precise class of allowed transition amplitudes, in which our definition is slightly more restrictive). We also note that the results of this paper would apply to any "reasonable" variant of space-bounded quantum computation that is classically controlled, which includes all of the "standard" variants that have been considered (see, for instance, [31, 35, 45, 51]). We refer the reader to [31, Section 2] for a thorough discussion of the various models of space-bounded quantum computation, and, in particular, of the reasonableness of requiring classical control; we also discuss the issue of classical vs. quantum control later in Section 6.

Definition 11. We define *general* quantum space $s(n)$ for each error-type, analogously to the *unitary* case of Definition 8. $\mathsf{BQSPACE}(s(n)) = \mathsf{QSPACE}(s(n))_{\frac{2}{3},\frac{1}{3}}$, $\mathsf{BQL} = \mathsf{BQSPACE}(\log n)$, $\mathsf{RQSPACE}(s(n)) = \mathsf{QSPACE}(s(n))_{\frac{1}{2},0}$, etc.

Note that the particular choice of constants $\frac{2}{3}$ and $\frac{1}{3}$ that appear in the definition of $\mathsf{BQSPACE}$ are arbitrary, as the completeness and soundness parameters can straightforwardly be amplified;

a similar statement holds for $\mathsf{RQSPACE}$. Finally, we define a space-bounded variant of quantum Merlin-Arthur proof systems, essentially following [18].

Definition 12. Consider functions $c, k : \mathbb{N} \to [0,1]$ and $s : \mathbb{N} \to \mathbb{N}$, with $s(n) = \Omega(\log n)$, all of which are computable in $\mathsf{DSPACE}(s(n))$. Let $s(n)$*−bounded−*$\mathsf{QMA}_{c(n),k(n)}$ denote the collection of all promise problems $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0)$ such that there is a $\mathsf{DSPACE}(s(n))$-uniform family of (unitary) quantum circuits $\{V_w : w \in \mathsf{P}\}$, where V_w acts on $m_w + h_w = O(s(|w|))$ qubits and has $2^{O(s(|w|))}$ gates, that has the following properties. Let $\Pi_1 = |1\rangle\langle 1| \otimes I_{2^{m_w+h_w-1}}$ and let Ψ_{m_w} denote the set of m_w-qubit states. For each $w \in \mathsf{P}$, the verification circuit V_w is applied to the state $|\psi\rangle \otimes |0^{h_w}\rangle$, where $|\psi\rangle \in \Psi_{m_w}$ is a (purported) proof of the fact that $w \in \mathsf{P}_1$. Then, the first qubit is measured in the standard basis. If the result is 1, then w is *accepted*; otherwise, w is *rejected*.

Completeness: $w \in \mathsf{P}_1 \Rightarrow \exists |\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] = \|\Pi_1 V_w(|\psi\rangle \otimes |0^{h_w}\rangle)\|^2 \geq c(|w|)$.

Soundness: $w \in \mathsf{P}_0 \Rightarrow \forall |\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] = \|\Pi_1 V_w(|\psi\rangle \otimes |0^{h_w}\rangle)\|^2 \leq k(|w|)$.

Definition 13. We then define space-bounded QMA, for a variety of types of error, as follows.

Bounded-error:

Two-sided : $\mathsf{QMASPACE}(s(n)) = s(n)\text{−}\textit{bounded}\text{−}\mathsf{QMA}_{\frac{2}{3},\frac{1}{3}}$.

Perfect completeness: $\mathsf{QMASPACE}_1(s(n)) = s(n)\text{−}\textit{bounded}\text{−}\mathsf{QMA}_{1,\frac{1}{2}}$.

Perfect soundness: $\mathsf{RQMASPACE}(s(n)) = s(n)\text{−}\textit{bounded}\text{−}\mathsf{QMA}_{\frac{1}{2},0}$.

Unbounded-error:

Perfect completeness: $\mathsf{PreciseQMASPACE}_1(s(n)) = \bigcup_{k:\mathbb{N}\to[0,1)} s(n)\text{−}\textit{bounded}\text{−}\mathsf{QMA}_{1,k(n)}$.

Perfect soundness: $\mathsf{NQMASPACE}(s(n)) = \bigcup_{c:\mathbb{N}\to(0,1]} s(n)\text{−}\textit{bounded}\text{−}\mathsf{QMA}_{c(n),0}$.

Let $\mathsf{QMAL} = \mathsf{QMASPACE}(\log n)$, $\mathsf{QMAL}_1 = \mathsf{QMASPACE}_1(\log n)$, etc.

3 Well-Conditioned Determinant

We define the following well-conditioned versions of the standard DET^*-complete problems [12].

Definition 14. Consider functions $m : \mathbb{N} \to \mathbb{N}$, $\kappa : \mathbb{N} \to \mathbb{R}_{\geq 1}$, and $\epsilon : \mathbb{N} \to \mathbb{R}_{>0}$. For a sequence of matrices $A_1, \ldots, A_m$, and for indices j_1, j_2, where $1 \leq j_1 \leq j_2 \leq m$, let $A_{j_1,j_2} = \prod_{j=j_1}^{j_2} A_j$.

(i) $\mathsf{DET}_{n,\kappa,\epsilon^{-1}}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $b \in \mathbb{R}_{\leq 0}$.
Promise: $\sigma_1(A) \leq 1$, $\sigma_n(A) \geq \frac{1}{\kappa(n)}$, $|\det(A)| \in [\kappa(n)^{-n}, e^{b-\epsilon(n)}] \cup [e^b, 1]$.
Output: 1 if $|\det(A)| \geq e^b$, 0 otherwise.

(ii) $\mathsf{DET}^+_{n,\kappa,\epsilon^{-1}}$
Input: $H \in \widehat{\mathrm{Pos}}(n)$, $b \in \mathbb{R}_{\leq 0}$.
Promise: $\sigma_1(H) \leq 1$, $\sigma_n(H) \geq \frac{1}{\kappa(n)}$, $\det(H) \in [\kappa(n)^{-n}, e^{b-\epsilon(n)}] \cup [e^b, 1]$.
Output: 1 if $\det(H) \geq e^b$, 0 otherwise.

(iii) $\mathsf{MATPOW}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{R}_{\geq 0}$.
Promise: $\sigma_1(A^j) \leq \kappa(n), \forall j \in \{1, \ldots, m\}$, $|A^m[s,t]| \in [0, b - \epsilon(n)] \cup [b, \kappa(n)]$.
Output: 1 if $|A^m[s,t]| \geq b$, 0 otherwise.

(iv) $\mathsf{ITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{R}_{\geq 0}$.
Promise: $\sigma_1(A_{j_1,j_2}) \leq \kappa(n)$ for $1 \leq j_1 \leq j_2 \leq m$, $|A_{1,m}[s,t]| \in [0, b - \epsilon(n)] \cup [b, \kappa(n)]$.
Output: 1 if $|A_{1,m}[s,t]| \geq b$, 0 otherwise.

(v) $\mathsf{ITMATPROD}^{\geq 0}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{R}_{\geq 0}$.
Promise: $\sigma_1(A_{j_1,j_2}) \leq \kappa(n)$ for $1 \leq j_1 \leq j_2 \leq m$, $A_{1,m}[s,t] \in [0, b - \epsilon(n)] \cup [b, \kappa(n)]$.
Output: 1 if $A_{1,m}[s,t] \geq b$, 0 otherwise.

(vi) $\mathsf{SUMITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$, $E \subseteq \{1, \ldots, n\}^2$, $b \in \mathbb{R}_{\geq 0}$.
Promise: $\sigma_1(A_{j_1,j_2}) \leq \kappa(n)$ for $1 \leq j_1 \leq j_2 \leq m$, $\left| \sum_{(s,t) \in E} A_{1,m}[s,t] \right| \in [0, b - \epsilon(n)] \cup [b, |E|\kappa(n)]$.
Output: 1 if $\left| \sum_{(s,t) \in E} A_{1,m}[s,t] \right| \geq b$, 0 otherwise.

(vii) $\mathsf{MATINV}_{n,\kappa,\epsilon^{-1}}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{R}_{\geq 0}$.
Promise: $\sigma_1(A) \leq 1$, $\sigma_n(A) \geq \frac{1}{\kappa(n)}$, $|A^{-1}[s,t]| \in [0, b - \epsilon(n)] \cup [b, \kappa(n)]$.
Output: 1 if $|A^{-1}[s,t]| \geq b$, 0 otherwise.

(viii) $\mathsf{MATINV}^{+}_{n,\kappa,\epsilon^{-1}}$
Input: $H \in \widehat{\mathrm{Pos}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{R}_{\geq 0}$.
Promise: $\sigma_1(H) \leq 1$, $\sigma_n(H) \geq \frac{1}{\kappa(n)}$, $|H^{-1}[s,t]| \in [0, b - \epsilon(n)] \cup [b, \kappa(n)]$.
Output: 1 if $|H^{-1}[s,t]| \geq b$, 0 otherwise.

Note that, with the exception of the problem DET (and DET^+), each of the above problems are defined such that they correspond to approximating some quantity with *additive* error $\epsilon/2$; for example, MATINV involves determining if $|A^{-1}[s,t]| \leq b - \epsilon$ or $|A^{-1}[s,t]| \geq b$. To clarify our definition of DET, observe that this problem, which involves determining if $|\det(A)| \leq e^{b-\epsilon}$ or $|\det(A)| \geq e^b$, is equivalent to the problem of determining if $\ln(|\det(A)|) \leq b - \epsilon$ or $\ln(|\det(A)|) \geq b$. In other words, we have defined DET such that it corresponds to obtaining an approximation of $\ln(|\det(A)|)$ with *additive* error $\epsilon/2$; this is equivalent to obtaining a $e^{\pm\frac{\epsilon}{2}}$ *multiplicative* approximation of $|\det(A)|$. As we will see in Section 3.1 and Section 3.3, this is the "correct" definition of DET, in the sense that it is the version of the determinant problem that most closely corresponds to the other linear-algebraic problems (matrix powering, matrix inversion, etc.) defined above.

Moreover, note that the problems as stated above are somewhat "over parameterized." For example, if $\langle A, s, t, b \rangle \in \mathsf{MATINV}_{n,\kappa(n),\epsilon^{-1}(n)}$, then $\langle \epsilon(n)A, s, t, \epsilon^{-1}(n)b \rangle \in \mathsf{MATINV}_{n,\kappa(n)\epsilon^{-1}(n),1}$ and $\mathsf{MATINV}(\langle A, s, t, b \rangle) = \mathsf{MATINV}(\langle \epsilon(n)A, s, t, \epsilon^{-1}(n)b \rangle)$. These additional parameters are convenient as they allow us to express certain results more cleanly.

Definition 15. For each promise problem $\mathsf{P}_{n,m,\kappa,\epsilon^{-1}}$ in Definition 14, we define *poly*-conditioned-P to be the promise problem $\mathsf{P}_{n,n^{O(1)},n^{O(1)},n^{O(1)}}$. For example,
poly-conditioned-DET^+
Input: $H \in \widehat{\mathrm{Pos}}(n)$, $b \in \mathbb{R}_{\leq 0}$.
Promise: $\sigma_1(H) \leq 1$, $\sigma_n(H) \geq n^{-O(1)}$, $\det(H) \in [n^{-O(n)}, e^{b-n^{-O(1)}}] \cup [e^b, 1]$.
Output: 1 if $\det(H) \geq e^b$, 0 otherwise.

3.1 $\mathbf{BQ_UL}$ Completeness

We now show that all of the above *poly*-conditioned problems are $\mathsf{BQ_UL}$-complete. By [18, Theorem 13], *poly*-conditioned-MATINV^+ is $\mathsf{BQ_UL}$-complete (note that the notation used in this paper to express the promised condition number differs from the notation used in [18]). Therefore, it suffices to show that, for each such *poly*-conditioned-P, we have *poly*-conditioned-P $\leq_{\mathsf{L}}^m$ *poly*-conditioned-MATINV^+ and *poly*-conditioned-$\mathsf{MATINV}^+ \leq_{\mathsf{L}}^m$ *poly*-conditioned-P.

Remark. We note that reductions between the standard versions of these problems (i.e., where there is no assumption of being well-conditioned) are well-known [5, 12, 13, 29, 46–48]. However, these reductions, generally, *do not* preserve the property of being *poly*-conditioned. In order to prove our $\mathsf{BQ_UL}$-completeness results, our definition of $\leq_{\mathsf{L}}^m$-reducibility (see Section 2.1) requires that this property is preserved. Therefore, we must, generally, exhibit reductions that are rather different from the "standard" reductions. Moreover, to our knowledge, even for those problems for which an already known reduction does preserve the property of being *poly*-conditioned, no proof that this property is preserved have previously appeared, and so we must exhibit such a proof.

We will prove several lemmas that exhibit reductions between particular problems in Definition 14. The proofs of these lemmas share a common structure: for a pair of promise problems P, P', we show how to transform an instance $w \in \mathsf{P}$ to an instance $f(w) \in \mathsf{P}'$ such that the reduction function f preserves the answer (i.e., $\mathsf{P}(w) = \mathsf{P}'(f(w))$) and also preserves the property of being well-conditioned. Note that $\mathsf{P} \leq_{\mathsf{L}}^m \mathsf{P}' \Rightarrow$ *poly*-conditioned-P $\leq_{\mathsf{L}}^m$ *poly*-conditioned-P′. In the following, we assume that $m(n)$, $\kappa(n)$, and $\epsilon(n)^{-1}$ can be computed to $O(\log n)$ bits of precision in uniform AC^0. For $m \in \mathbb{N}_{\geq 1}$ and $r, c \in \{1, \ldots, m\}$, define $F_{m,r,c} \in \widehat{\mathrm{Mat}}(m)$ such that $F_{m,r,c}[r, c] = 1$ and $F_{m,r,c}[r', c'] = 0$, $\forall (r', c') \neq (r, c)$.

Note that, the proof (which appears in Section 3.2) of Theorem 1 (which states the equivalence between unitary and general quantum space) only requires the first three of the following lemmas. The remaining lemmas are used to prove the other main results of this paper.

Lemma 16. *$\mathsf{ITMATPROD} \leq_{\mathsf{AC}^0}^m \mathsf{MATPOW}$.*

Proof. Following [12], consider $\langle A_1, \ldots, A_m, s, t, b\rangle \in \mathsf{ITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$. Let $\widehat{A} \in \widehat{\mathrm{Mat}}(nm + n)$ consist of $n \times n$ blocks, where the blocks immediately above the main diagonal blocks are given by $A_1, \ldots, A_m$, and all other entries are 0; more explicitly, $\widehat{A} = \sum_{r=1}^{m}(F_{m+1,r,r+1} \otimes A_r)$. For $j \in \{1, \ldots, m\}$, we have

$$\widehat{A}^j = \sum_{r=1}^{m+1-j} (F_{m+1,r,r+j} \otimes A_{r,r+j-1}).$$

Let $\widehat{s} = s$ and $\widehat{t} = nm + t$. Then $\widehat{A}^m[\widehat{s}, \widehat{t}] = A_{1,m}[s, t]$. Moreover, for any $j \in \{1, \ldots, m\}$, we have

$$\sigma_1(\widehat{A}^j) = \sigma_1\left(\sum_{r=1}^{m+1-j} (F_{m+1,r,r+j} \otimes A_{r,r+j-1})\right) = \sigma_1\left(\bigoplus_{r=1}^{m+1-j} A_{r,r+j-1}\right) = \max_r \sigma_1(A_{r,r+j-1}) \leq \kappa(n).$$

Let $\widehat{b} = b$. We then conclude that $\langle \widehat{A}, \widehat{s}, \widehat{t}, \widehat{b} \rangle \in \mathsf{MATPOW}_{nm+n,m,\kappa(n),\epsilon^{-1}(n)}$ and, furthermore, that $\mathsf{ITMATPROD}(\langle A_1, \ldots, A_m, s, t, b \rangle) = \mathsf{MATPOW}(\langle \widehat{A}, \widehat{s}, \widehat{t}, \widehat{b} \rangle)$. □

Lemma 17. $\mathsf{MATPOW} \leq^m_{\mathsf{AC}^0} \mathsf{MATINV}$.

Proof. Following [12], consider $\langle A, s, t, b \rangle \in \mathsf{MATPOW}_{n,m,\kappa,\epsilon^{-1}}$. For $j \in \{0, \ldots, m\}$, let $G_j = \sum_{r=1}^{m+1-j} F_{m+1,r,r+j} \in \widehat{\mathrm{Mat}}(m+1)$. Let $Y = G_1 \otimes A \in \widehat{\mathrm{Mat}}(nm+n)$ consist of $n \times n$ blocks, where the blocks immediately above the main diagonal blocks are all given by A. Let $Z = I_{nm+n} - Y \in \widehat{\mathrm{Mat}}(nm+n)$ and observe that

$$Z^{-1} = \sum_{j=0}^{m} \left(G_j \otimes A^j\right).$$

Let $\widehat{s} = s$ and $\widehat{t} = nm + t$. Then $Z^{-1}[\widehat{s}, \widehat{t}] = A^m[s,t]$. We also have $\sigma_1(Z) \leq \sigma_1(I_{nm+n}) + \sigma_1(Y) \leq 1 + \kappa(n)$ and

$$\sigma_1(Z^{-1}) = \sigma_1\left(\sum_{j=0}^{m}\left(G_j \otimes A^j\right)\right) \leq \sum_{j=0}^{m} \sigma_1(G_j \otimes A^j) \leq 1 + \sum_{j=1}^{m} \sigma_1(A^j) \leq 1 + \sum_{j=1}^{m} \kappa(n) \leq 1 + m\kappa(n).$$

This implies $\sigma_{nm+n}(Z) = \sigma_1(Z^{-1})^{-1} \geq (1 + m\kappa(n))^{-1}$. Let $\widehat{Z} = \frac{1}{\lceil 1+\kappa(n) \rceil} Z \in \widehat{\mathrm{Mat}}(nm+n)$ and $\widehat{b} = \lceil 1 + \kappa(n) \rceil b$. We then conclude that $\langle \widehat{Z}, \widehat{s}, \widehat{t}, \widehat{b} \rangle \in \mathsf{MATINV}_{nm+n,(1+m\kappa(n))\lceil 1+\kappa(n)\rceil, \lceil 1+\kappa(n)\rceil^{-1}\epsilon^{-1}(n)}$ and $\mathsf{MATPOW}(\langle A, s, t, b \rangle) = \mathsf{MATINV}(\langle \widehat{Z}, \widehat{s}, \widehat{t}, \widehat{b} \rangle)$. □

Lemma 18. $\mathsf{MATINV} \leq^m_{\mathsf{NC}^1} \mathsf{MATINV}^+$.

Proof. Consider $\langle A, s, t, b \rangle \in \mathsf{MATINV}_{\kappa,\epsilon^{-1}}$. Let $\widehat{H} = \frac{1}{3}\begin{pmatrix} A^\dagger A & -A^\dagger \\ -A & 2I \end{pmatrix} \in \widehat{\mathrm{Pos}}(2n)$. Then $\widehat{H}^{-1} = 3\begin{pmatrix} 2(A^\dagger A)^{-1} & A^{-1} \\ (A^\dagger)^{-1} & I \end{pmatrix}$. Moreover, $\sigma_1(\widehat{H}) \leq 1$ and $\sigma_{2n}(\widehat{H}) \geq \frac{1}{9}(\sigma_n(A))^2 \geq (3\kappa(n))^{-2}$. Let $\widehat{s} = s$, $\widehat{t} = t + n$, and $\widehat{b} = 3b$. Then $\widehat{H}^{-1}[\widehat{s}, \widehat{t}] = 3A^{-1}[s,t]$. Therefore, $\langle \widehat{H}, \widehat{s}, \widehat{t}, \widehat{b} \rangle \in \mathsf{MATINV}^+_{2n,(3\kappa(n))^2,(3\epsilon(n))^{-1}}$ and $\mathsf{MATINV}(\langle A, s, t, b \rangle) = \mathsf{MATINV}(\langle H, \widehat{s}, \widehat{t}, \widehat{b} \rangle)$. □

Lemma 19. $\mathsf{MATINV}^+ \leq^m_{\mathsf{AC}^0} \mathsf{SUMITMATPROD}$.

Proof. Consider $\langle H, s, t, b \rangle \in \mathsf{MATINV}^+_{\kappa,\epsilon^{-1}}$. For $m \in \mathbb{N}$, we have

$$\sum_{j=0}^{m} (I - H)^j = H^{-1}(I - (I - H)^{m+1}).$$

Let $\widehat{m} = \lceil \kappa(n) \rceil \lfloor 1 + \log(\lfloor 4\kappa(n)\epsilon(n)^{-1} \rfloor) \rfloor$. For $j \in \{1, \ldots, \widehat{m}\}$, let $\widehat{A}_j = I_{jn} \oplus (I_{\widehat{m}-j+1} \otimes (I - H)) \in \widehat{\mathrm{Mat}}(n\widehat{m} + n)$. For $1 \leq j_1 \leq j_2 \leq \widehat{m}$, we have

$$\sigma_1(\widehat{A}_{j_1,j_2}) = \sigma_1\left(I_{j_1 n} \oplus \left(\bigoplus_{k=1}^{\widehat{m}-j_1+1} (I-H)^{\min(k, j_2-j_1+1)}\right)\right) = \max_{k \in \{0,\ldots,j_2-j_1+1\}} \sigma_1((I-H)^k) = 1.$$

Let $\widehat{E} = \{(s + jn, t + jn) : j \in \{0, \dots, \widehat{m}\}\}$. We then have

$$\sum_{(\widehat{s},\widehat{t})\in\widehat{E}} \widehat{A}_{1,\widehat{m}}[\widehat{s},\widehat{t}] = \sum_{j=0}^{\widehat{m}} (I-H)^j[s,t] = (H^{-1}(I-(I-H)^{\widehat{m}+1}))[s,t].$$

This implies

$$\left|\left|\sum_{(\widehat{s},\widehat{t})\in\widehat{E}} \widehat{A}_{1,\widehat{m}}[\widehat{s},\widehat{t}]\right| - |H^{-1}[s,t]|\right| \le |(H^{-1}(I-H)^{\widehat{m}+1})[s,t]| \le \sigma_1(H^{-1}(I-H)^{\widehat{m}+1})$$

$$\le \sigma_1(H^{-1})(\sigma_1(I-H))^{\widehat{m}+1} \le \kappa(n)\left(1-\frac{1}{\kappa(n)}\right)^{\widehat{m}+1} \le \frac{1}{4}\epsilon(n).$$

Let $\widehat{b} = b - \frac{1}{4}\epsilon(n)$. We then conclude that $\langle \widehat{A}_1, \dots, \widehat{A}_{\widehat{m}}, \widehat{E}, \widehat{b}\rangle \in \mathsf{SUMITMATPROD}_{n\widehat{m}+n,\widehat{m},1,2\epsilon^{-1}(n)}$ and $\mathsf{MATINV}(\langle H, s, t, b\rangle) = \mathsf{SUMITMATPROD}(\langle \widehat{A}_1, \dots, \widehat{A}_{\widehat{m}}, \widehat{E}, \widehat{b}\rangle)$. □

Lemma 20. $\mathsf{SUMITMATPROD} \le^m_{\mathsf{AC}^0} \mathsf{ITMATPROD}$.

Proof. Consider some $\langle A_1, \dots, A_m, E, b\rangle \in \mathsf{SUMITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$. Let $T_{c,d} \in \widehat{\mathrm{Mat}}(n)$ denote the permutation matrix corresponding to interchanging elements $c, d \in \{1, \dots, n\}$ and leaving all other elements fixed. For $j \in \{1, \dots, m\}$, let $\widehat{A}_j = \bigoplus_{(s,t)\in E} T_{1,t}A_jT_{1,s} \in \widehat{\mathrm{Mat}}(n|E|)$. Let $R \in \widehat{\mathrm{Mat}}(|E|)$ be defined such that $R_{r,c} = 1$ if $r = c$ or $r = 1$, and $R_{r,c} = 0$ otherwise; let $\widehat{A}_0 = R \otimes I_n$ and $\widehat{A}_{m+1} = \widehat{A}_0^\dagger$. We then have $\widehat{A}_{0,m+1}[1,1] = \sum_{(s,t)\in E} A_{1,m}[s,t]$. Notice that $\sigma_1(\widehat{A}_0) = \sigma_1(\widehat{A}_{m+1}) = \sigma_1(R)\sigma_1(I_n) \le \sqrt{2|E|}$, which implies

$$\sigma_1\left(\widehat{A}_{j_1,j_2}\right) \le 2|E|\sigma_1\left(A_{\max(j_1,1),\min(j_2,m)}\right) \le 2|E|\kappa(n) \le 2n^2\kappa(n), \text{ for } 0 \le j_1 \le j_2 \le m+1.$$

Let $\widehat{s} = \widehat{t} = 1$ and $\widehat{b} = b$. Then $\langle \widehat{A}_0, \dots, \widehat{A}_{m+1}, \widehat{s}, \widehat{t}, \widehat{b}\rangle \in \mathsf{ITMATPROD}_{n^3,m+2,2n^2\kappa(n),\epsilon^{-1}(n)}$ and $\mathsf{SUMITMATPROD}(\langle A_1, \dots, A_m, E, b\rangle) = \mathsf{ITMATPROD}(\langle \widehat{A}_0, \dots, \widehat{A}_{m+1}, \widehat{s}, \widehat{t}, \widehat{b}\rangle)$. □

Lemma 21. $\mathsf{DET}^+ \le^m_{\mathsf{AC}^0} \mathsf{SUMITMATPROD}$.

Proof. Consider some $\langle H, b\rangle \in \mathsf{DET}^+_{n,\kappa,\epsilon^{-1}}$. By the promise, $H \in \widehat{\mathrm{Pos}}(n)$, $\lambda_1(H) = \sigma_1(H) \le 1$, and $\lambda_n(H) = \sigma_n(H) \ge \kappa(n)^{-1}$, which implies $\sigma_1(I-H) = \lambda_1(I-H) = 1 - \lambda_n(H) \le 1 - \kappa(n)^{-1} < 1$. This implies $\ln(H) = -\sum_{k=1}^{\infty} \frac{(I-H)^k}{k}$, where here $\ln(H)$ denotes the matrix logarithm. Recall that, as a consequence of Jacobi's formula, $\ln(\det(H)) = \mathrm{tr}(\ln(H))$.

For $m \in \mathbb{N}_{\ge 1}$, let $S_m = \sum_{k=1}^{m} \frac{(I-H)^k}{k}$, let $R_m = \sum_{k=m+1}^{\infty} \frac{(I-H)^k}{k} = -\log(H) - S_m$, and let $D_m \in \widehat{\mathrm{Mat}}(m)$ denote the diagonal matrix where $D_m[k,k] = \frac{1}{k}$. Let $\widehat{l} = \lfloor 1 + \log(\lfloor \kappa(n) \rfloor)\rfloor$, let $\widehat{A}_1 = I_{n\widehat{l}} \oplus (-D_m \otimes (I-H)) \in \widehat{\mathrm{Mat}}(n\widehat{l} + nm)$, and, for $k \in \{2, \dots, m\}$, let $\widehat{A}_k = I_{n(\widehat{l}+k-1)} \oplus (I_{m+1-k} \otimes (I-H)) \in \widehat{\mathrm{Mat}}(n\widehat{l} + nm)$. Then

$$\widehat{A}_{1,m} = \prod_{j=1}^{m} \widehat{A}_j = I_{n\widehat{l}} \oplus \left(\bigoplus_{k=1}^{m} \frac{-(I-H)^k}{k}\right).$$

Let $E_m = \{(d,d) : d \in \{1, \ldots, n\widehat{l} + nm\}\}$. We then have

$$\sum_{(s,t)\in E_m} \widehat{A}_{1,m}[s,t] = \operatorname{tr}(\widehat{A}_{1,m}) = \operatorname{tr}(I_{n\widehat{l}}) - \sum_{k=1}^{m} \operatorname{tr}\left(\frac{(I-H)^k}{k}\right) = n\widehat{l} - \operatorname{tr}(S_m) = n\widehat{l} + \ln(\det(H)) + \operatorname{tr}(R_m).$$

Moreover, for $1 \leq j_1 \leq j_2 \leq m$, we have

$$\sigma_1(\widehat{A}_{j_1,j_2}) \leq \max\left(\sigma_1(I_{n\widehat{l}}), \max_{k\in\{0,\ldots,j_2-j_1\}} \sigma_1((I-H)^k)\right) = 1.$$

As shown above, $\sigma_1(I-H) \leq 1 - \kappa(n)^{-1}$, which implies

$$\sigma_1(R_m) = \sigma_1\left(\sum_{k=m+1}^{\infty} \frac{(I-H)^k}{k}\right) \leq \sum_{k=m+1}^{\infty} \frac{(\sigma_1(I-H))^k}{k} \leq \sum_{k=m+1}^{\infty} \frac{(1-\kappa(n)^{-1})^k}{k} \leq \kappa(n)\left(1 - \frac{1}{\kappa(n)}\right)^{m+1}.$$

If $m \geq \kappa(n)\ln(2n\kappa(n)\epsilon(n)^{-1})$, then

$$\operatorname{tr}(R_m) \leq n\sigma_1(R_m) \leq n\kappa(n)\left(1 - \frac{1}{\kappa(n)}\right)^{\kappa(n)\ln(2n\kappa(n)\epsilon(n)^{-1})} \leq n\kappa(n)\left(\frac{1}{e}\right)^{\ln(2n\kappa(n)\epsilon(n)^{-1})} = \frac{1}{2}\epsilon(n).$$

Let $\widehat{m} = \lceil\kappa(n)\rceil\lfloor 1 + \log(\lfloor 2n\kappa(n)\epsilon(n)^{-1}\rfloor)\rfloor \geq \kappa(n)\ln(2n\kappa(n)\epsilon(n)^{-1})$ and $\widehat{E} = E_{\widehat{m}}$. Note that $\operatorname{tr}(R_m) \geq 0$. We then have,

$$n\widehat{l} + \ln(\det(H)) \leq \sum_{(s,t)\in\widehat{E}} \widehat{A}_{1,m}[s,t] = n\widehat{l} + \ln(\det(H)) + \operatorname{tr}(R_{\widehat{m}}) \leq n\widehat{l} + \ln(\det(H)) + \frac{1}{2}\epsilon(n).$$

If $\det(H) \geq e^b$, then $\sum_{(s,t)\in\widehat{E}} \widehat{A}_{1,m}[s,t] \geq n\widehat{l} + b$; if $\det(H) \leq e^{b-\epsilon(n)}$, then $\sum_{(s,t)\in\widehat{E}} \widehat{A}_{1,m}[s,t] \leq n\widehat{l} + b - \frac{1}{2}\epsilon(n)$. Let $\widehat{b} = n\widehat{l} + b$. Therefore, $\langle \widehat{A}_1, \ldots, \widehat{A}_{\widehat{m}}, \widehat{E}, \widehat{b}\rangle \in \mathsf{SUMITMATPROD}_{n(\widehat{l}+\widehat{m}),\widehat{m},1,2\epsilon^{-1}(n)}$ and $\mathsf{DET}(\langle H, b\rangle) = \mathsf{SUMITMATPROD}(\langle \widehat{A}_1, \ldots, \widehat{A}_{\widehat{m}}, \widehat{E}, \widehat{b}\rangle)$. $\square$

Lemma 22. $\mathsf{ITMATPROD} \leq^m_{\mathsf{AC}^0} \mathsf{ITMATPROD}^{\geq 0}$.

Proof. Consider some $\langle A_1, \ldots, A_m, s, t, b\rangle \in \mathsf{ITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$. Let $\chi_t \in \mathbb{R}^n$ denote the vector where $\chi_t[t] = 1$ and $\chi_t[k] = 0\ \forall k \neq t$. For $j \in \{1, \ldots, 2m+1\}$, we define

$$\widehat{A}_j = \begin{cases} A_j, & j \leq m \\ \chi_t\chi_t^\dagger, & j = m+1 \\ A^\dagger_{2m+2-j}, & j \geq m+2 \end{cases}$$

We then have $\widehat{A}_{1,2m+1}[s,s] = A_{1,m}[s,t]\overline{A_{1,m}[s,t]} = |A_{1,m}[s,t]|^2$. Consider j_1, j_2 such that $1 \leq j_1 \leq j_2 \leq m+1$. If $j_1 \leq m+1 \leq j_2$, then

$$\sigma_1(\widehat{A}_{j_1,j_2}) = \sigma_1(A_{j_1,m}\chi_t\chi_t^\dagger A^\dagger_{j_2,m}) \leq \sigma_1(A_{j_1,m})\sigma_1(\chi_t\chi_t^\dagger)\sigma_1(A^\dagger_{j_2,m}) \leq \kappa(n)^2.$$

If $j_2 < m+1$, then $\sigma_1(\widehat{A}_{j_1,j_2}) = \sigma_1(A_{j_1,j_2}) \leq \kappa(n)$; finally, if $j_1 > m+1$, then $\sigma_1(\widehat{A}_{j_1,j_2}) = \sigma_1(A_{j_1-m-1,j_2-m-1}) \leq \kappa(n)$.

Let $\widehat{s} = \widehat{t} = s$ and $\widehat{b} = b^2$. Then $\langle \widehat{A}_1, \ldots, \widehat{A}_{2m+1}, \widehat{s}, \widehat{t}, \widehat{b}\rangle \in \mathsf{ITMATPROD}^{\geq 0}_{n,2m+1,\kappa^2,\epsilon^{-2}}$ and $\mathsf{ITMATPROD}(\langle A_1, \ldots, A_m, s, t, b\rangle) = \mathsf{ITMATPROD}(\langle \widehat{A}_1, \ldots, \widehat{A}_{2m+1}, \widehat{s}, \widehat{t}, \widehat{b}\rangle)$. $\square$

Lemma 23. $\mathsf{ITMATPROD}^{\geq 0} \leq^m_{\mathsf{AC}^0} \mathsf{DET}$.

Proof. Consider some $\langle A_1, \ldots, A_m, s, t, b\rangle \in \mathsf{ITMATPROD}^{\geq 0}_{n,m,\kappa,\epsilon^{-1}}$. Let $Y = \sum_{r=1}^{m} (F_{m+1,r,r+1} \otimes A_r) \in \widehat{\mathrm{Mat}}(nm+n)$ consist of $n \times n$ blocks, where the blocks immediately above the main diagonal blocks are given by $A_1, \ldots, A_m$, and all other entries are 0. Let $B = I_{nm+n} - Y \in \widehat{\mathrm{Mat}}(nm+n)$ and observe that

$$B^{-1} = I_{nm+n} + \sum_{r=1}^{m} \sum_{c=r+1}^{m+1} (F_{m+1,r,c} \otimes A_{r,c-1}).$$

For $k \in \{1, \ldots, nm+n\}$, let $\chi_k \in \mathbb{R}^{nm+n}$ denote the vector that is 1 in entry k and 0 elsewhere; let $v = \chi_s$ and let $u = \chi_{nm+t}$. Notice that $\det(B) = 1$. By the matrix determinant lemma,

$$\det(B + uv^\dagger) = (1 + v^\dagger B^{-1} u) \det(B) = 1 + B^{-1}[s, nm+t] = 1 + A_{1,m}[s,t].$$

Next, observe that

$$\sigma_1(B + uv^\dagger) \leq \sigma_1(uv^\dagger) + \sigma_1(I) + \sigma_1(Y) \leq 2 + \max_j \sigma_1(A_j) \leq 2 + \kappa(n).$$

Notice that $B^{-1} = \sum_{j=0}^{m} Y^j$, which implies

$$\sigma_1(B^{-1}) \leq \sum_{j=0}^{m} \sigma_1(Y^j) \leq 1 + \sum_{j=1}^{m} \left(\max_{k \in \{1,\ldots,m-j+1\}} \sigma_1(A_{k,k+j-1}) \right) \leq 1 + \sum_{j=1}^{m} \kappa(n) = 1 + m\kappa(n).$$

By the Sherman-Morrison formula, $(B + uv^\dagger)^{-1} = B^{-1}(I - (1 + v^\dagger B^{-1} u)^{-1} uv^\dagger B^{-1})$. Recall that, by the promise, $v^\dagger B^{-1} u = A_{1,m}[s,t] \in \mathbb{R}_{\geq 0}$. Therefore,

$$\sigma_1((B + uv^\dagger)^{-1}) \leq \sigma_1(B^{-1})(\sigma_1(I) + \sigma_1((1 + v^\dagger B^{-1}u)^{-1} uv^\dagger)\sigma_1(B^{-1})) \leq (1 + m\kappa(n))(2 + m\kappa(n)).$$

This implies $\sigma_{nm+n}(B) = \sigma_1(B^{-1})^{-1} \geq ((1 + m\kappa(n))(2 + m\kappa(n)))^{-1}$. Let $\widehat{l} = \lfloor 1 + \ln(\lfloor 2 + \kappa(n) \rfloor) \rfloor$ and let $\widehat{B} = e^{-\widehat{l}}(B + uv^\dagger) \in \widehat{\mathrm{Mat}}(nm+n)$. Then, for $j \in \{1, \ldots, nm+n\}$, $\sigma_j(\widehat{B}) = e^{-\widehat{l}} \sigma_j(B)$; in particular, $\sigma_1(\widehat{B}) \leq 1$ and $\sigma_{nm+n}(\widehat{B}) \geq (2 + m\kappa(n))^{-3}$. Moreover,

$$|\det(\widehat{B})| = |e^{-\widehat{l}(nm+n)} \det(B + uv^\dagger)| = |e^{-\widehat{l}(nm+n)}(1 + A_{1,m}[s,t])| = e^{-\widehat{l}(nm+n)}(1 + A_{1,m}[s,t]).$$

Let $\widehat{a} = \ln(1 + b - \epsilon(n)) - \widehat{l}(nm+n)$ and $\widehat{b} = \ln(1+b) - \widehat{l}(nm+n)$. If $A_{1,m}[s,t] \geq b$, then $|\det(\widehat{B})| \geq e^{\widehat{b}}$; if $A_{1,m}[s,t] \leq b - \epsilon(n)$, then $|\det(\widehat{B})| \leq e^{\widehat{a}}$. We have

$$\widehat{b} - \widehat{a} = \ln\left(\frac{1+b}{1+b-\epsilon(n)}\right) = \ln\left(1 + \frac{\epsilon(n)}{1+b-\epsilon(n)}\right) \geq \ln\left(1 + \frac{\epsilon(n)}{1+\kappa(n)}\right) \geq \frac{\epsilon(n)}{2(1+\kappa(n))}.$$

Therefore, $\langle \widehat{B}, \widehat{b}\rangle \in \mathsf{DET}_{nm+m,(2+m\kappa(n))^3,\epsilon^{-1}(n)(2+2\kappa(n))}$ and $\mathsf{ITMATPROD}(\langle A_1, \ldots, A_m, s, t, b\rangle) = \mathsf{DET}(\langle \widehat{B}, \widehat{b}\rangle)$. ☐

Lemma 24. $\mathsf{DET} \leq^m_{\mathsf{NC}^1} \mathsf{DET}^+$.

Proof. Consider $\langle A, b\rangle \in \mathsf{DET}_{n,\kappa,\epsilon^{-1}}$. Let $\widehat{H} = AA^\dagger \in \widehat{\mathrm{Pos}}(n)$ and $\widehat{b} = 2b$. Then, $\det(\widehat{H}) = |\det(A)|^2$ and $\sigma_j(\widehat{H}) = \sigma_j^2(A)$, $\forall j$. Therefore, $\langle \widehat{H}, \widehat{b}\rangle \in \mathsf{DET}_{n,\kappa^2,2\epsilon^{-1}}$ and $\mathsf{DET}(\langle A, b\rangle) = \mathsf{DET}(\langle \widehat{H}, \widehat{b}\rangle)$. ☐

We now prove Theorem 4 from Section 1.1, which we restate here for convenience.

Theorem 4. *Each poly-conditioned promise problem given in Definition 15 is* $\mathsf{BQ_UL}$*-complete.*

Proof. By [18, Theorem 13], *poly*-conditioned-MATINV^+ is $\mathsf{BQ_UL}$-complete. By the preceding lemmas, we have

$$\mathsf{MATINV}^+ \leq^m_{\mathsf{AC}^0} \mathsf{SUMITMATPROD} \leq^m_{\mathsf{AC}^0} \mathsf{ITMATPROD} \leq^m_{\mathsf{AC}^0} \mathsf{MATPOW} \leq^m_{\mathsf{AC}^0} \mathsf{MATINV} \leq^m_{\mathsf{NC}^1} \mathsf{MATINV}^+$$

and

$$\mathsf{DET}^+ \leq^m_{\mathsf{AC}^0} \mathsf{SUMITMATPROD} \leq^m_{\mathsf{AC}^0} \mathsf{ITMATPROD} \leq^m_{\mathsf{AC}^0} \mathsf{ITMATPROD}^{\geq 0} \leq^m_{\mathsf{AC}^0} \mathsf{DET} \leq^m_{\mathsf{NC}^1} \mathsf{DET}^+.$$

Note that $\leq^m_{\mathsf{AC}^0}$ or $\leq^m_{\mathsf{NC}^1}$ reducibility implies $\leq^m_{\mathsf{L}}$ reducibility; further, note that $\mathsf{P} \leq^m_{\mathsf{L}} \mathsf{P}' \Rightarrow$ *poly*-conditioned-$\mathsf{P} \leq^m_{\mathsf{L}}$ *poly*-conditioned-P'. Therefore, we conclude that each such *poly*-conditioned problem is $\mathsf{BQ_UL}$-complete. ☐

3.2 BQSPACE vs. BQ$_\mathsf{U}$SPACE vs. BPSPACE vs. QMASPACE

In this section, we explore the relationships between the classes $\mathsf{BQSPACE}(s(n))$, $\mathsf{BQ_USPACE}(s(n))$, $\mathsf{BPSPACE}(s(n))$, and $\mathsf{QMASPACE}(s(n))$. First, we consider the case in which $s(n) = \log n$. While, trivially, $\mathsf{BPL} \subseteq \mathsf{BQL}$, it is not obvious, *a priori*, that $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$. To the best of our knowledge, the strongest partial result in this direction is the classic result of Watrous [49, Theorem 4.12], which showed that BPL is contained in a variant of $\mathsf{BQ_UL}$ in which there is no bound on the running time of the QTM.

By Theorem 4, *poly*-conditioned-$\mathsf{MATPOW} \in \mathsf{BQ_UL}$. As we next observe, this implies $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$ and, more strongly, $\mathsf{BQL} = \mathsf{BQ_UL}$. Of course, the statement $\mathsf{BQL} = \mathsf{BQ_UL}$ immediately implies $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$; nevertheless, we will first show, directly, that $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$. We then extend these results to any (space-constructible) space bound $s(n) = \Omega(\log n)$, by use of a standard padding argument.

Proposition 25. $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$.

Proof. Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{BPL}$. Then there is some probabilistic TM M that recognizes P with two-sided error $\leq \frac{1}{3}$ within time $t(n) = n^{O(1)}$ and space $s(n) = O(\log n)$, for any input $w \in \mathsf{P}$ of length $n = |w|$. Let $|M|$ denote the size of the finite control of M, let Γ denote the work-tape alphabet of M, and let $c(n) = |M|(n+2)(s(n))|\Gamma|^{s(n)} = n^{O(1)}$ denote the number of possible configurations of M on inputs of length n. It is well-known that, for input $w \in \mathsf{P}$, one may construct, in deterministic space $O(\log(|w|))$ a stochastic matrix $A_w \in \widehat{\mathrm{Mat}}(c(n))$ and values $x_w, y_w \in \{1, \ldots, c(n)\}$ such that $A_w^t[x_w, y_w]$ is precisely the probability that M accepts w within t steps [15,34]. Note that, as A_w is stochastic, so is A_w^t, $\forall t \in \mathbb{N}$; this implies $\sigma_1(A_w^t) \leq \sqrt{c(n)} = n^{O(1)}$. Therefore, $\langle A_w, x_w, y_w, \frac{2}{3}\rangle \in \mathsf{MATPOW}_{c(n),t(n),\sqrt{c(n)},3}$ and $\mathsf{MATPOW}(\langle A_w, x_w, y_w, \frac{2}{3}\rangle) = \mathsf{P}(w)$. By Theorem 4, $\mathsf{MATPOW}_{c(n),t(n),\sqrt{c(n)},3} \in \mathsf{BQ_UL}$, which implies $\mathsf{P} \in \mathsf{BQ_UL}$. ☐

By applying an analogous argument to general quantum Turing machines (where the stochastic matrix that describes a single step of the computation of a probabilistic TM is replaced by the quantum channel that describes a single step of the computation of a quantum TM), we may then show that $\mathsf{BQL} \subseteq \mathsf{BQ_UL}$ (and, therefore, that $\mathsf{BQL} = \mathsf{BQ_UL}$); for completeness, we include such a proof in Appendix A. Instead, in this section, we exhibit an analogous proof using the general quantum circuits of Definition 9.

Lemma 26. *poly-conditioned-*$\mathsf{ITMATPROD}$ *is* BQL*-hard.*

Proof. Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{BQL}$. By definition, there is some L-uniform family of general quantum circuits $\{\Phi_w = (\Phi_{w,1}, \ldots, \Phi_{w,t_w}) : w \in \mathsf{P}\}$, where Φ_w acts on $h_w = O(\log|w|)$ qubits and has $t_w = |w|^{O(1)}$ gates, such that if $w \in \mathsf{P}_1$, then $\Pr[\Phi_w \text{ accepts } w] \geq \frac{2}{3}$, and if $w \in \mathsf{P}_0$, then $\Pr[\Phi_w \text{ accepts } w] \leq \frac{1}{3}$. Without loss of generality we may, for convenience, assume that Φ_w "cleans-up" its workspace at the end of the computation, by measuring the first qubit in the computational basis, and then forcing every other qubit to the state $|0\rangle$ (by measuring each such qubit in the computational basis and, if the result 1 is obtained, flipping its value).

Let $d_w = 2^{2h_w} = |w|^{O(1)}$. For $j \in \{1, \ldots, t_w\}$, let $A(w)_j = K(\Phi_{w,t_w-j+1})$, and note that, by Definition 9, $A(w)_j \in \widehat{\mathrm{Mat}}(d_w)$ and $A(w)_j$ can be constructed in deterministic space $O(\log(|w|))$. Moreover, as $\Phi_{w,j} \in \mathrm{Chan}(2^{h_w})$, for $1 \leq j_1 \leq j_2 \leq t_w$, we have $\Phi_{w,t_w-j_2+1} \circ \cdots \circ \Phi_{w,t_w-j_1+1} \in \mathrm{Chan}(2^{h_w})$, which by [38, Theorem 1] implies the following bound on the largest singular value of any partial product of the $A(w)_j$

$$\sigma_1(A(w)_{j_1,j_2}) = \sigma_1\left(\prod_{j=j_1}^{j_2} A(w)_j\right) = \sigma_1(K(\Phi_{w,t_w-j_2+1} \circ \cdots \circ \Phi_{w,t_w-j_1+1})) \leq \sqrt{d_w} = n^{O(1)}.$$

Let $x_w = |10^{h_w-1}\rangle\langle 10^{h_w-1}|$ and $y_w = |0^{h_w}\rangle\langle 0^{h_w}|$. By Definition 10,

$$\Pr[\Phi_w \text{ accepts } w] = \left(\prod_{j=1}^{t_w} A(w)_j\right)[x_w, y_w] = A(w)_{1,t_w}[x_w, y_w].$$

We then conclude that $\langle A(w)_1, \ldots, A(w)_{t_w}, x_w, y_w, \frac{2}{3}\rangle \in \mathsf{ITMATPROD}_{n^{O(1)},n^{O(1)},n^{O(1)},n^{O(1)}}$ and that $\mathsf{ITMATPROD}(\langle A(w)_1, \ldots, A(w)_{t_w}, x_w, y_w, \frac{2}{3}\rangle) = \mathsf{P}(w)$. ☐

Theorem 27. $\mathsf{BQL} = \mathsf{BQ_UL} = \mathsf{QMAL}$.

Proof. Clearly, $\mathsf{BQ_UL} \subseteq \mathsf{BQL}$. By Lemma 26, *poly*-conditioned-$\mathsf{ITMATPROD}$ is BQL-hard; by Theorem 4, *poly*-conditioned-$\mathsf{ITMATPROD} \in \mathsf{BQ_UL}$, which implies $\mathsf{BQL} \subseteq \mathsf{BQ_UL}$. By [18, Theorem 18], $\mathsf{QMAL} = \mathsf{BQ_UL}$. ☐

We now prove Theorem 1, our main result stated in the introduction, which we restate here for convenience.

Theorem 1. *For any space-constructible function* $s : \mathbb{N} \to \mathbb{N}$*, where* $s(n) = \Omega(\log n)$*, we have*

$$\mathsf{BQ_USPACE}(s(n)) = \mathsf{BQSPACE}(s(n)) = \mathsf{QMASPACE}(s(n)).$$

Proof. Clearly, $\mathsf{BQ_USPACE}(s(n)) \subseteq \mathsf{BQSPACE}(s(n))$. By [18, Theorem 18 and Theorem 26], $\mathsf{QMASPACE}(s(n)) = \mathsf{BQ_USPACE}(s(n))$. All that remains is to show that $\mathsf{BQSPACE}(s(n)) \subseteq \mathsf{BQ_USPACE}(s(n))$, which follows from the preceding theorem and a standard padding argument; for the sake of completeness, we now briefly state this argument.

Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{BQSPACE}(s(n))$. By definition, there is some $\mathsf{DSPACE}(s(n))$-uniform family of general quantum circuits $\{\Phi_w = (\Phi_{w,1}, \ldots, \Phi_{w,t_w}) : w \in \mathsf{P}\}$, where Φ_w acts on $h_w = O(s(|w|))$ qubits and has $t_w = 2^{O(s(|w|))}$ gates, such that if $w \in \mathsf{P}_1$, then $\Pr[\Phi_w \text{ accepts } w] \geq \frac{2}{3}$, and if $w \in \mathsf{P}_0$, then $\Pr[\Phi_w \text{ accepts } w] \leq \frac{1}{3}$. Let M denote a deterministic Turing machine (DTM) that produces this family of circuits within the stated space bound. Let Σ denote the finite alphabet over which P is defined, and assume, without loss of generality, that $\{0, 1\} \subseteq \Sigma$.

We define $\mathsf{P}^{\log} = (\mathsf{P}^{\log}_1, \mathsf{P}^{\log}_0) \subseteq \Sigma^*$ such that $\mathsf{P}^{\log}_j = \{w01^{2^{s(|w|)}} : w \in \mathsf{P}_j\}$, for $j \in \{0,1\}$. We next show that $\mathsf{P}^{\log} \in \mathsf{BQL}$, by exhibiting a family of general quantum circuits $\{\Phi^{\log}_x = (\Phi^{\log}_{x,1}, \ldots, \Phi^{\log}_{x,t^{\log}_x}) : x \in \mathsf{P}^{\log}\}$, with the appropriate parameters, that recognizes $\mathsf{P}^{\log}$. Begin by noticing that, as s is space-constructible, there is a DTM D that uses space $O(\log n)$ on all inputs of length n, where, on any input $x \in \Sigma^*$, D checks if $x = w01^{2^{s(|w|)}}$, for some $w \in \Sigma^*$. If x is of this form, then D marks the rightmost symbol of w; otherwise, D rejects. We then construct a DTM M' which, on input $x \in \Sigma^*$ produces $\Phi^{\log}_x$, as follows. First, M' runs D. If D rejects, then M' outputs a trivial single-gate circuit, which acts on a single qubit and always rejects. Otherwise (i.e., when the input x is of the form $w01^{2^{s(|w|)}}$), M' simulates M on the prefix w, producing the circuit Φ_w; note that, in this case, $|x| = |w|+1+2^{s(|w|)}$, which implies M' runs in space $O(s(|w|)) = O(\log(2^{s(|w|)})) = O(\log(|x|))$, and that $\Phi^{\log}_x$ acts on $h^{\log}_x = h_w = O(s(|w|)) = O(\log(|x|))$ qubits and has $t^{\log}_x = t_w = 2^{O(s(|w|))} = |x|^{O(1)}$ gates. Therefore, $\{\Phi^{\log}_x : x \in \mathsf{P}^{\log}\}$ is a L-uniform family of general quantum circuits, with the appropriate parameters, that recognizes $\mathsf{P}^{\log}$ with two-sided bounded-error $\frac{1}{3}$.

Thus, by Theorem 27, $\mathsf{P}^{\log} \in \mathsf{BQ_UL}$. To complete the proof, we show that this implies $\mathsf{P} \in \mathsf{BQ_USPACE}(s(n))$. To see this, let $\{Q^{\log}_x : x \in \mathsf{P}^{\log}\}$ denote a L-uniform family of (unitary) quantum circuits that recognizes $\mathsf{P}^{\log}$ with two-sided bounded-error $\frac{1}{3}$, where $Q^{\log}_x$ acts on $O(\log(|x|))$ qubits and has $|x|^{O(1)}$ gates, and let M_U denote a logspace DTM that produces this circuit family. We then define a DTM M'_U, which, on input $w \in \Sigma^*$, simply simulates M_U on $x = w01^{2^{s(|w|)}}$; in particular, in order to keep track of the simulated head of M_U when it is in the suffix $01^{2^{s(|w|)}}$, M'_U marks $s(|w|)+1$ cells on its work tape (recall that s is space-constructible), which is then used as a classical counter that can count up to $2^{s(|w|)+1}-1$. By an analysis similar to that of the previous paragraph, we see that M'_U runs in space $O(s(n))$ and that the circuit family $\{Q_w : w \in \mathsf{P}\}$ that it produces recognizes P and has the correct parameters. $\square$

3.3 Fully Logarithmic Approximation Schemes

In this section, we study the class of functions that are well-approximable in quantum logspace, following (essentially) the notation and definitions of [16]. In particular, we work with the general (resp. unitary) quantum Turing machine model, rather than the *equivalent* model of a uniform family of general (resp. unitary) quantum circuits; of course, all results also apply to the quantum circuit model. For simplicity, throughout this section, we fix the alphabet $\Sigma = \{0,1\}$. We say that a function $f : \Sigma^* \to \mathbb{R}$ is *poly*-bounded if $|f(w)| \leq poly(|w|)$, $\forall w \in \Sigma^*$.

Definition 28. We say that a *poly*-bounded f has a *fully logarithmic quantum approximation scheme* FLQAS if there is a (general) quantum TM M_f that, on input $\langle x, \epsilon, \delta\rangle$, where $x \in \Sigma^*$ and $\epsilon, \delta \in \mathbb{R}_{>0}$, runs in time $poly(|x|, \epsilon^{-1}, \log(\delta^{-1}))$ and space $O(\log(|x|) + \log(\epsilon^{-1}) + \log(\log(\delta^{-1})))$, and outputs a value $y \in \mathbb{R}$ such that $\Pr[|f(x) - y| \geq \epsilon] \leq \delta$ (to be precise, M_f outputs a string that encodes a dyadic rational number y). In other words, with *confidence* at least $1-\delta$, the value y is an *additive* approximation of $f(x)$ with *error* at most ϵ. We analogously say that f has a $\mathsf{FLQ_UAS}$ if M_f is a *unitary* quantum TM, a FLRAS if M_f is a *randomized* TM, and a FLAS if M_f is a *deterministic* TM (where, in this last case, we set $\delta = 0$ and remove the dependence on δ from the time and space bounds).

Following the notation established in Section 2.1 and Definitions 14 and 15, let

$$\mathcal{D}(poly\text{-}matinv) = \bigcup_n \{\langle A, s, t\rangle \in \Sigma^* : A \in \widehat{\mathrm{Mat}}(n), 1 \geq \sigma_1(A) \geq \sigma_n(A) \geq n^{-O(1)}, s, t \in \{1, \ldots, n\}\}.$$

In other words, $\mathcal{D}(\textit{poly-matinv})$ consists of precisely those strings in Σ^* that are encodings of instances of a variant of *poly*-conditioned-MATINV where only the portion of the promise involving singular values is required (i.e., there is no restriction on $A^{-1}[s,t]$ involving b). We then consider the *poly*-conditioned matrix inversion function $\textit{poly-matinv} : \mathcal{D}(\textit{poly-matinv}) \to \mathbb{C}$, given by $\textit{poly-matinv}(\langle A, s, t\rangle) = A^{-1}[s,t]$. For consistency with [16], as well as to make the relationship between our results and certain logspace counting classes more clear, we then consider $\Re(\textit{poly-matinv}(\cdot)) : \mathcal{D}(\textit{poly-matinv}) \to \mathbb{R}$ (resp. $|\textit{poly-matinv}(\cdot)| : \mathcal{D}(\textit{poly-matinv}) \to \mathbb{R}_{\geq 0}$), which are given by the real part (resp. the magnitude) of the *poly*-conditioned matrix inversion function.

Similarly, we define $\mathcal{D}(\textit{poly-itmatprod}) \subseteq \Sigma^*$ to consist of all strings $\langle A_1, \ldots, A_m, s, t\rangle$, where $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$ and $s, t \in \{1, \ldots, n\}$, where $m = poly(n)$ and $\sigma_1(A_{j_1,j_2}) \leq poly(n)$ for $1 \leq j_1 \leq j_2 \leq m$. We then define the function $\textit{poly-itmatprod} : \mathcal{D}(\textit{poly-itmatprod}) \to \mathbb{C}$ such that $\textit{poly-itmatprod}(\langle A_1, \ldots, A_m, s, t\rangle) = A_{1,m}[s,t]$. We also define $\Re(\textit{poly-itmatprod}(\cdot)) : \mathcal{D}(\textit{poly-itmatprod}) \to \mathbb{R}$ and $|\textit{poly-itmatprod}(\cdot)| : \mathcal{D}(\textit{poly-itmatprod}) \to \mathbb{R}_{\geq 0}$ as above. Lastly, we define $\mathcal{D}(\textit{poly-det}) = \bigcup_n \{\langle A\rangle : A \in \widehat{\mathrm{Mat}}(n), 1 \geq \sigma_1(A) \geq \sigma_n(A) \geq n^{-O(1)}\}$ and $\textit{poly-det} : \mathcal{D}(\textit{poly-det}) \to \mathbb{C}$, such that, $\textit{poly-det}(\langle A\rangle) = \det(A)$. Recall that the promise problem DET, given in Definition 14, corresponds to approximating the function $\ln(|\textit{poly-det}(\cdot)|) : \mathcal{D}(\textit{poly-det}) \to \mathbb{R}_{\leq 0}$.

Lemma 29. *$\Re(\textit{poly-matinv}(\cdot))$, $|\textit{poly-matinv}(\cdot)|$, $\Re(\textit{poly-itmatprod}(\cdot))$, $|\textit{poly-itmatprod}(\cdot)|$, and $\ln(|\textit{poly-det}(\cdot)|)$ each have a $\mathsf{FLQ_UAS}$.*

Proof. By [18, Theorem 14] (and the discussion following it), the functions $\Re(\textit{poly-matinv}(\cdot))$ and $|\textit{poly-matinv}(\cdot)|$ each have a $\mathsf{FLQ_UAS}$ (to be precise, the aforementioned theorem considered only the case in which $A \in \widehat{\mathrm{Pos}}(n)$, the general case then follows from the fact that the reduction from MATINV to MATINV^+ given by Lemma 18 preserves the value of the corresponding entry of the inverse matrix, not merely its magnitude); this improved upon the earlier result of Ta-Shma [45], which showed that these functions each have a FLQAS [16]. Notice that the reduction from $\mathsf{ITMATPROD}$ to MATPOW given by Lemma 16, and the reduction from MATPOW to MATINV given by Lemma 17, both also preserve the value of the matrix entry in question, not merely its magnitude. Therefore, $\Re(\textit{poly-itmatprod}(\cdot))$ and $|\textit{poly-itmatprod}(\cdot)|$ each have a $\mathsf{FLQ_UAS}$. Finally, by Lemmas 20, 21 and 24, $\ln(|\textit{poly-det}(\cdot)|)$ has a $\mathsf{FLQ_UAS}$. $\square$

Note that, following [16], we have defined fully logarithmic (quantum, randomized, etc.) approximation schemes with respect to *additive* error ϵ; that is to say, we approximate $f(x)$ by a value y such that $\Pr[|f(x) - y| \geq \epsilon] \leq \delta$. We then define a *multiplicative* fully logarithmic (quantum, randomized, etc.) approximation scheme of a function $g : \Sigma^* \to \mathbb{R}_{\geq 0}$ as an analogous procedure that produces an approximation z such that $\Pr[z \notin [e^{-\epsilon}g(x), e^{\epsilon}g(x)]] \leq \delta$. Note that here, for convenience, we follow the convention (as used in, for example, [21]) that multiplicative approximations are defined using $e^{\pm\epsilon}$, rather than the more standard (and essentially equivalent) $(1 \pm \epsilon)$.

Lemma 30. *$\ln(|\textit{poly-det}(\cdot)|)$ has an (additive) $\mathsf{FLQ_UAS}$ (resp. $\mathsf{FLQAS}, \mathsf{FLRAS}, \mathsf{FLAS}$) if and only if $|\textit{poly-det}(\cdot)|$ has a multiplicative $\mathsf{FLQ_UAS}$ (resp. $\mathsf{FLQAS}, \mathsf{FLRAS}, \mathsf{FLAS}$). In particular, $|\textit{poly-det}(\cdot)|$ has a multiplicative $\mathsf{FLQ_UAS}$.*

Proof. The first statement follows immediately from the fact that $|\ln(|\det(A)|) - y| \geq \epsilon$ if and only if $e^y \notin [e^{-\epsilon}|\det(A)|, e^{\epsilon}|\det(A)|]$. The second statement is a consequence of the first statement and Lemma 29. $\square$

Next, recall that by [16, Theorem 6], if $\mathsf{BQL} = \mathsf{BPL}$, then every *poly*-bounded function that has a FLQAS also has a FLRAS (recall that we use BQL and BPL to denote classes of promise problems, which differs from the notation used in [16]). By combining this with the $\mathsf{BQ_UL}$-hardness of the various *poly*-conditioned promise problems (Theorem 4) and our result that $\mathsf{BQ_UL} = \mathsf{BQL}$ (Theorem 27), the following proposition is immediate; we note that a partial version of this proposition was implicit in [15].

Proposition 31. *The following statements are equivalent.*

(i) $\mathsf{BQL} = \mathsf{BPL}$.

(ii) Every poly-bounded function that has a FLQAS *also has a* FLRAS.

(iii) Every poly-bounded function that has a $\mathsf{FLQ_UAS}$ *also has a* FLRAS.

(iv) $\Re(\textit{poly-matinv}(\cdot))$ *has a* FLRAS.

(v) $|\textit{poly-matinv}(\cdot)|$ *has a* FLRAS.

(vi) $\Re(\textit{poly-itmatprod}(\cdot))$ *has a* FLRAS.

(vii) $|\textit{poly-itmatprod}(\cdot)|$ *has a* FLRAS.

(viii) $\ln(|\textit{poly-det}(\cdot)|)$ *has a* FLRAS.

(ix) $|\textit{poly-det}(\cdot)|$ *has a multiplicative* FLRAS.

Remark. In particular, the preceding proposition suggests that $|\textit{poly-det}(\cdot)|$ *does not* have a multiplicative FLRAS (as this would imply the *seemingly* unlikely statement $\mathsf{BQL} = \mathsf{BPL}$). It is natural to compare this statement with the result of Jerrum, Sinclair, and Vigoda [21] which shows the existence of a (multiplicative) FPRAS (fully *polynomial* randomized approximation scheme) for the *permanent* of a nonnegative integer matrix.

4 Well-Conditioned Singular

We next consider a well-conditioned version of the problem of determining if a matrix is singular. Additionally, following [41], we also consider well-conditioned versions of the "verification" versions of standard DET-complete problems.

Definition 32. Consider functions $m : \mathbb{N} \to \mathbb{N}$, $\kappa : \mathbb{N} \to \mathbb{R}_{\geq 1}$, and $\epsilon : \mathbb{N} \to \mathbb{R}_{>0}$.

(i) $\mathsf{SINGULAR}_{n,\epsilon^{-1}}$
Input: $A \in \widehat{\mathrm{Herm}}(n)$.
Promise: $\sigma_1(A) \leq 1$, $\sigma_n(A) \in \{0\} \cup [\epsilon(n), 1]$.
Output: 1 if $\sigma_n(A) = 0$, 0 otherwise.

(ii) $\mathsf{vMATPOW}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{Q}[i]_n$.
Promise: $\sigma_1(A^j) \leq \kappa(n), \forall j \in \{1, \ldots, m\}$, $|b| \leq \kappa(n)$, $|A^m[s,t] - b| \in \{0\} \cup [\epsilon(n), 2\kappa(n)]$.
Output: 1 if $A^m[s,t] = b$, 0 otherwise.

(iii) $\mathsf{vITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{Q}[i]_n$.
Promise: $\sigma_1(A_{j_1,j_2}) \leq \kappa(n)$ for $1 \leq j_1 \leq j_2 \leq m$, $|b| \leq \kappa(n)$, $|A_{1,m}[s,t] - b| \in \{0\} \cup [\epsilon(n), 2\kappa(n)]$.
Output: 1 if $|A_{1,m}[s,t]| = b$, 0 otherwise.

(iv) $\mathsf{vMATINV}_{n,\kappa,\epsilon^{-1}}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s,t \in \{1,\ldots,n\}$, $b \in \mathbb{Q}[i]_n$.
Promise: $\sigma_1(A) \leq 1$, $\sigma_n(A) \geq \frac{1}{\kappa(n)}$, $|b| \leq \kappa(n)$, $|A^{-1}[s,t] - b| \in \{0\} \cup [\epsilon(n), 2\kappa(n)]$.
Output: 1 if $A^{-1}[s,t] = b$, 0 otherwise.

Remark. Note that the requirement $|b| \leq \kappa(n)$ above is unnecessary due to the fact that, for any matrix M, $\max_{s,t}|M[s,t]| \leq \sigma_1(M)$ (therefore, if $|b| > \kappa(n)$, which can easily be checked, then the input is a 0 instance). We include this condition only for convenience.

We begin by exhibiting reductions between the above problems; in subsequent sections, we will use these reductions to prove new properties of quantum logspace.

Lemma 33. $\mathsf{vITMATPROD} \leq^m_{\mathsf{AC}^0} \mathsf{vMATPOW}$.

Proof. Precisely analogous to the proof of Lemma 16. ☐

Lemma 34. $\mathsf{vMATPOW} \leq^m_{\mathsf{AC}^0} \mathsf{vMATINV}$.

Proof. Precisely analogous to the proof of Lemma 17. ☐

Lemma 35. $\mathsf{vMATINV} \leq^m_{\mathsf{AC}^0} \mathsf{SINGULAR}$.

Proof. Consider $\langle A,s,t,b\rangle \in \mathsf{vMATINV}_{n,\kappa,\epsilon^{-1}}$. Let $\widehat{B} = (2\lceil\kappa(n)\rceil A) \oplus \left(1 - \frac{b}{2\lceil\kappa(n)\rceil}\right)^{-1} I_1 \in \widehat{\mathrm{Mat}}(n+1)$, $u = \chi_s + \chi_{n+1}$, $v = \chi_t + \chi_{n+1}$, and $\widehat{C} = \widehat{B} - vu^\dagger \in \widehat{\mathrm{Mat}}(n+1)$. By the matrix determinant lemma,

$$\det(\widehat{C}) = (1 - u\widehat{B}^{-1}v)\det(\widehat{B}) = \left(1 - \left(\frac{A^{-1}[s,t]}{2\lceil\kappa(n)\rceil} + \left(1 - \frac{b}{2\lceil\kappa(n)\rceil}\right)\right)\right)\det(\widehat{B}) = \frac{b - A^{-1}[s,t]}{2\lceil\kappa(n)\rceil}\det(\widehat{B}).$$

If $A^{-1}[s,t] = b$, then $\det(\widehat{C}) = 0$, which implies $\sigma_{n+1}(\widehat{C}) = 0$. Next, suppose $|A^{-1}[s,t] - b| \geq \epsilon(n)$, then $|\det(\widehat{C})| \geq \frac{\epsilon(n)}{2\lceil\kappa(n)\rceil}|\det(\widehat{B})|$. By the Weyl inequalities, $\sigma_1(\widehat{C}) \leq \sigma_1(\widehat{B}) + \sigma_1(-vu^\dagger) = \sigma_1(\widehat{B}) + 1$ and, for $j \in \{2,\ldots,n+1\}$, we have $\sigma_j(\widehat{C}) \leq \sigma_{j-1}(\widehat{B}) + \sigma_2(-vu^\dagger) = \sigma_{j-1}(\widehat{B})$. Moreover, $\sigma_1(\widehat{B}) = 2\lceil\kappa(n)\rceil\sigma_1(A) \leq 2\lceil\kappa(n)\rceil$, $\sigma_n(\widehat{B}) = 2\lceil\kappa(n)\rceil\sigma_n(A) \geq 2$, and $\sigma_{n+1}(\widehat{B}) = \left|1 - \frac{b}{2\lceil\kappa(n)\rceil}\right|^{-1} \geq \frac{2}{\sqrt{5}}$. Therefore,

$$\sigma_{n+1}(\widehat{C}) = \frac{|\det(\widehat{C})|}{\sigma_1(\widehat{C})\prod_{j=2}^{n}\sigma_j(\widehat{C})} \geq \frac{\frac{\epsilon(n)}{2\lceil\kappa(n)\rceil}|\det(\widehat{B})|}{(\sigma_1(\widehat{B})+1)\prod_{j=1}^{n-1}\sigma_j(\widehat{B})} = \frac{\epsilon(n)\sigma_n(\widehat{B})\sigma_{n+1}(\widehat{B})}{2\lceil\kappa(n)\rceil(\sigma_1(\widehat{B})+1)} \geq \frac{2\epsilon(n)}{\sqrt{5}\lceil\kappa(n)\rceil(2\lceil\kappa(n)\rceil+1)}.$$

Let $\widehat{d} = (2\lceil\kappa(n)\rceil + 1)^{-1}$ and let $\widehat{H} = \widehat{d}\begin{pmatrix} 0_{n+1} & \widehat{C} \\ \widehat{C}^\dagger & 0_{n+1}\end{pmatrix} \in \widehat{\mathrm{Herm}}(2n+2)$. Notice that $\widehat{H}$ has eigenvalues $\{\pm\widehat{d}\sigma_1(\widehat{C}),\ldots,\pm\widehat{d}\sigma_{n+1}(\widehat{C})\}$. This implies $\sigma_1(\widehat{H}) = \widehat{d}\sigma_1(\widehat{C}) \leq 1$ and $\sigma_{2n+2}(\widehat{H}) = \widehat{d}\sigma_{n+1}(\widehat{C}) \in \{0\} \cup \left[\frac{2\epsilon(n)}{\sqrt{5}\lceil\kappa(n)\rceil(2\lceil\kappa(n)\rceil+1)^2}, 1\right]$. Moreover, $\sigma_{2n+2}(\widehat{H}) = 0 \Leftrightarrow A^{-1}[s,t] = b$. Therefore, $\langle\widehat{H}\rangle \in \mathsf{SINGULAR}_{2n+2,(2\epsilon(n))^{-1}\sqrt{5}\lceil\kappa(n)\rceil(2\lceil\kappa(n)\rceil+1)^2}$ and $\mathsf{vMATINV}(\langle A,s,t,b\rangle) = \mathsf{SINGULAR}(\langle\widehat{H}\rangle)$. ☐

4.1 RQSPACE vs. RQ$_\mathsf{U}$SPACE vs. RQMASPACE vs. QMASPACE$_1$

For those promise problems P given in Definition 32, we define *poly*-conditioned-P as in Definition 15.

Lemma 36. *poly-conditioned-*$\mathsf{vITMATPROD}$ *is* coRQL*-hard.*

Proof. Precisely analogous to the proof of Lemma 26. □

Lemma 37. $\mathsf{RQMAL} \subseteq \mathsf{RQ_UL}$

Proof. Apply the well-known technique of replacing Merlin's proof with the totally mixed state [30], which preserves perfect soundness [26]; then use space-efficient probability amplification for one-sided bounded-error (unitary) quantum logspace [50]. We briefly sketch the details.

Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{RQMAL}$. By definition, there is a L-uniform family of (unitary) quantum circuits $\{V_w : w \in \mathsf{P}\}$, where V_w acts on $m_w + h_w = O(\log|w|)$ qubits and has $t_w = poly(|w|)$ gates, such that $w \in \mathsf{P}_1 \Rightarrow \exists|\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] \geq c = \frac{1}{2}$, and $w \in \mathsf{P}_0 \Rightarrow \forall|\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] = k = 0$, where $\Pr[V_w \text{ accepts } w, |\psi\rangle] = \|\Pi_1 V_w(|\psi\rangle \otimes |0^{h_w}\rangle)\|^2$.

As in the proof of [30, Theorem 3.8], let $A_w = (I_{2^{m_w}} \otimes \langle 0^{h_w}|)V_w^\dagger \Pi_1 V_w(I_{2^{m_w}} \otimes |0^{h_w}\rangle) \in \mathrm{Pos}(2^{m_w})$; then $w \in \mathsf{P}_1 \Rightarrow \mathrm{tr}(A_w) \geq c = \frac{1}{2}$ and $w \in \mathsf{P}_0 \Rightarrow \mathrm{tr}(A_w) \leq 2^{m_w}k = 0 \Rightarrow \mathrm{tr}(A_w) = 0$. Similar to the proof of [26, Theorem 14] (cf. [30, Theorem 3.10]), we define a L-uniform family of (unitary) quantum circuits $\{Q_w : w \in \mathsf{P}\}$, where Q_w acts on $2m_w + h_w = O(\log|w|)$ qubits and has $t_w + O(m_w) = poly(|w|)$ gates, such that, when Q_w is applied to the state $|0^{2m_w+h_w}\rangle$, it simulates V_w on $|q\rangle \otimes |0^{h_w}\rangle$, where $|q\rangle \in \Psi_{m_w}$ is drawn uniformly at random from the 2^{m_w} standard basis elements of Ψ_{m_w}. We have $\Pr[Q_w \text{ accepts } w] = \mathrm{tr}(A_w 2^{-m_w} I_{2^{m_w}}) = 2^{-m_w}\,\mathrm{tr}(A_w)$; thus,

$$w \in \mathsf{P}_1 \Rightarrow \Pr[Q_w \text{ accepts } w] = 2^{-m_w}\,\mathrm{tr}(A_w) \geq 2^{-(m_w+1)} = 1/poly(|w|),$$

$$\text{and } w \in \mathsf{P}_0 \Rightarrow \Pr[Q_w \text{ accepts } w] = 2^{-m_w}\,\mathrm{tr}(A_w) = 0.$$

Therefore, $\mathsf{P} \in \mathsf{Q_USPACE}(\log n)_{\frac{1}{poly(n)},0}$. By [50, Lemma 5.1] $\mathsf{Q_USPACE}(\log n)_{\frac{1}{poly(n)},0} = \mathsf{RQ_UL}$, which implies $\mathsf{P} \in \mathsf{RQ_UL}$. □

Lemma 38. *poly-conditioned-*$\mathsf{SINGULAR}$ *is* QMAL_1*-hard.*

Proof. Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{QMAL}_1$. By definition, there is a L-uniform family of (unitary) quantum circuits $\{V_w = (V_{w,1}, \ldots, V_{w,t_w}) : w \in \mathsf{P}\}$, where V_w acts on $m_w + h_w = O(\log|w|)$ qubits and has $t_w = poly(|w|)$ gates, such that $w \in \mathsf{P}_1 \Rightarrow \exists|\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] \geq c = 1$, and $w \in \mathsf{P}_0 \Rightarrow \forall|\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] \leq k = \frac{1}{2}$, where $\Pr[V_w \text{ accepts } w, |\psi\rangle] = \|\Pi_1 V_w(|\psi\rangle \otimes |0^{h_w}\rangle)\|^2$.

We make use of the Kitaev clock Hamiltonian construction [24, Section 14.4], in a manner similar to [18, Lemma 21] (though, without the need to first apply space-efficient probability amplification techniques). Let $d_w = 2^{m_w+h_w}(t_w+1) = poly(|w|)$, define the d_w-dimensional Hilbert space $\mathcal{H}_w = \mathbb{C}^{2^{m_w}} \otimes \mathbb{C}^{2^{h_w}} \otimes \mathbb{C}^{t_w+1}$, and let $\Pi_b = I_{2^{b-1}} \otimes |1\rangle\langle 1| \otimes I_{2^{m_w+h_w-b}} \in \widehat{\mathrm{Proj}}(2^{m_w+h_w})$ denote the projection onto the subspace of $\mathbb{C}^{2^{m_w}} \otimes \mathbb{C}^{2^{h_w}}$ spanned by states in which the b^{th} qubit is 1. We define the Hamiltonians $H_w^{prop}, H_w^{in}, H_w^{out}, H_w \in \widehat{\mathrm{Pos}}(d_w)$ on $\mathcal{H}_w$ as follows:

$$H_w^{prop} = \frac{1}{2}\sum_{j=1}^{t_w}\Big(-V_{w,j} \otimes |j\rangle\langle j-1| - V_{w,j}^\dagger \otimes |j-1\rangle\langle j| + I_{2^{m_w+h_w}} \otimes (|j\rangle\langle j| + |j-1\rangle\langle j-1|)\Big)$$

$$H_w^{in} = \sum_{b=m_w+1}^{m_w+h_w} (\Pi_b \otimes |0\rangle\langle 0|), \quad H_w^{out} = \Pi_1 \otimes |t_w\rangle\langle t_w|, \quad \text{and} \quad H_w = H_w^{in} + H_w^{prop} + H_w^{out}.$$

By [24, Section 14.4], $\exists r_0, r_1 \in \mathbb{R}_{>0}$, such that $\sigma_1(H_w) \leq r_0, \forall w \in \mathsf{P}$, $\sigma_{d_w}(H_w) \leq \frac{1-c}{t_w+1} = 0, \forall w \in \mathsf{P}_1$ and $\sigma_{d_w}(H_w) \geq r_1 \frac{1-\sqrt{k}}{t_w+1} = 1/poly(d_w), \forall w \in \mathsf{P}_0$. Therefore, $\langle H_w \rangle \in$ *poly*-conditioned-SINGULAR and $\mathsf{P}(w) = \mathsf{SINGULAR}(\langle H_w \rangle)$; as $\{V_w : w \in \mathsf{P}\}$ is L-uniform, we see that H_w may be constructed from w in L, which implies $\mathsf{P} \leq_{\mathsf{L}}^m$ *poly*-conditioned-SINGULAR. □

Lemma 39. *poly-conditioned-*SINGULAR $\in \mathsf{QMAL}_1$.

Proof. Follows from using the quantum walk based Hamiltonian simulation technique of Childs [7, 11] to allow the phase estimation of [18, Lemma 19] to be carried out with one-sided error, in the style of [18, Proposition 32], we omit the straightforward details. □

Lemma 40. *poly-conditioned-*SINGULAR *is* QMAL_1*-complete.*

Proof. Follows immediately from Lemmas 38 and 39 □

Theorem 41. $\mathsf{RQMAL} = \mathsf{RQ_UL} \subseteq \mathsf{RQL} \subseteq \mathsf{coQMAL}_1$.

Proof. Trivially, $\mathsf{RQ_UL} \subseteq \mathsf{RQL}$ and $\mathsf{RQ_UL} \subseteq \mathsf{RQMAL}$. By Lemma 37, $\mathsf{RQMAL} \subseteq \mathsf{RQ_UL}$, which implies $\mathsf{RQMAL} = \mathsf{RQ_UL}$. By Lemma 36, *poly*-conditioned-vITMATPROD is coRQL-hard; thus, by Lemmas 33 to 35, *poly*-conditioned-SINGULAR is coRQL-hard. Finally, by Lemma 39, we have *poly*-conditioned-SINGULAR $\in \mathsf{QMAL}_1$, which implies $\mathsf{coRQL} \subseteq \mathsf{QMAL}_1$; therefore, $\mathsf{RQL} \subseteq \mathsf{coQMAL}_1$. □

We now prove Theorem 2, stated in the introduction, which we restate here for convenience.

Theorem 2. *For any space-constructible function* $s : \mathbb{N} \to \mathbb{N}$*, where* $s(n) = \Omega(\log n)$*, we have*

$$\mathsf{RQMASPACE}(s(n)) = \mathsf{RQ_USPACE}(s(n)) \subseteq \mathsf{RQSPACE}(s(n)) \subseteq \mathsf{coQMASPACE}_1(s(n)).$$

Proof. Follows from Theorem 41 and a padding argument analogous to that of Theorem 1. □

4.2 NQSPACE vs. NQ$_\mathsf{U}$SPACE vs. NQMASPACE vs. PreciseQMASPACE$_1$

We next consider variants of the well-conditioned problems of Definition 32, in which $\epsilon(n) = 0$, $\forall n \in \mathbb{N}$.

Definition 42. Consider functions $m : \mathbb{N} \to \mathbb{N}$ and $\kappa : \mathbb{N} \to \mathbb{R}_{\geq 1}$.

(i) PreciseSINGULAR$_n$
Input: $A \in \widehat{\mathrm{Herm}}(n)$.
Promise: $\sigma_1(A) \leq 1$, $\sigma_n(A) \in [0, 1]$.
Output: 1 if $\sigma_n(A) = 0$, 0 otherwise.

(ii) vPreciseMATPOW$_{n,m,\kappa}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{Q}[i]_n$.
Promise: $\sigma_1(A^j) \leq \kappa(n), \forall j \in \{1, \ldots, m\}$, $|b| \leq \kappa(n)$, $|A^m[s, t] - b| \in [0, 2\kappa(n)]$.
Output: 1 if $A^m[s, t] = b$, 0 otherwise.

(iii) $\mathsf{vITMATPROD}_{n,m,\kappa}$
Input: $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{Q}[i]_n$.
Promise: $\sigma_1(A_{j_1,j_2}) \leq \kappa(n)$ for $1 \leq j_1 \leq j_2 \leq m$, $|b| \leq \kappa(n)$, $|A_{1,m}[s,t] - b| \in [0, 2\kappa(n)]$.
Output: 1 if $|A_{1,m}[s,t]| = b$, 0 otherwise.

(iv) $\mathsf{vPreciseMATINV}_{n,\kappa}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{Q}[i]_n$.
Promise: $\sigma_1(A) \leq 1$, $\sigma_n(A) \geq \frac{1}{\kappa(n)}$, $|b| \leq \kappa(n)$, $|A^{-1}[s,t] - b| \in [0, 2\kappa(n)]$.
Output: 1 if $A^{-1}[s,t] = b$, 0 otherwise.

By arguments precisely analogous to those of Section 4.1, we obtain the following lemmas; we omit the proofs.

Lemma 43. *poly-conditioned-*$\mathsf{vPreciseITMATPROD}$ *is* coNQL*-hard.*

Lemma 44. $\mathsf{NQMAL} \subseteq \mathsf{NQ_UL}$

Lemma 45. $\mathsf{PreciseSINGULAR}$ *is* $\mathsf{PreciseQMAL_1}$*-complete.*

Theorem 46. $\mathsf{NQMAL} = \mathsf{NQ_UL} = \mathsf{NQL} = \mathsf{coPreciseQMAL_1} = \mathsf{coC_=L}$.

Proof. Trivially, $\mathsf{NQ_UL} \subseteq \mathsf{NQL}$ and $\mathsf{NQ_UL} \subseteq \mathsf{NQMAL}$. By Lemma 44, $\mathsf{NQMAL} \subseteq \mathsf{NQ_UL}$, which implies $\mathsf{NQMAL} = \mathsf{NQ_UL}$. By Lemma 43, *poly*-conditioned-$\mathsf{vPreciseITMATPROD}$ is coNQL-hard; thus, by Lemmas 33 to 35, $\mathsf{PreciseSINGULAR}$ is coNQL-hard. By Lemma 45, *poly*-conditioned-$\mathsf{SINGULAR} \in \mathsf{PreciseQMAL_1}$, which implies $\mathsf{NQL} \subseteq \mathsf{coPreciseQMAL_1}$. By [5, Theorem 14], $\mathsf{PreciseSINGULAR}$ is $\mathsf{C_=L}$-complete, and by Lemma 45, $\mathsf{PreciseSINGULAR}$ is $\mathsf{PreciseQMAL_1}$-complete; therefore, $\mathsf{C_=L} = \mathsf{PreciseQMAL_1}$. Thus far, we have shown $\mathsf{NQMAL} = \mathsf{NQ_UL} \subseteq \mathsf{NQL} \subseteq \mathsf{coPreciseQMAL_1} = \mathsf{coC_=L}$. To complete the proof, note that, by [49, Theorem 4.14], $\mathsf{NQ_UL} = \mathsf{coC_=L}$. ☐

We now prove Theorem 3, stated in the introduction, which we restate here for convenience.

Theorem 3. *For any space-constructible function* $s : \mathbb{N} \to \mathbb{N}$*, where* $s(n) = \Omega(\log n)$*, we have*

$$\mathsf{NQMASPACE}(s(n)) = \mathsf{NQ_USPACE}(s(n)) = \mathsf{NQSPACE}(s(n))$$

$$= \mathsf{coPreciseQMA_1SPACE}(s(n)) = \mathsf{coC_=SPACE}(s(n)).$$

Proof. Follows from Theorem 46 and a padding argument analogous to that of Theorem 1. ☐

5 Well-Conditioned Minimum Eigenvalue and Circuit Trace

In Section 3.1, a "scaled-down" version of the well-conditioned matrix inversion problem was shown to be $\mathsf{BQ_UL}$-complete, where the "standard" version of this problem is BQP-complete [20]. We next consider "scaled-down" versions of the minimum eigenvalue problem and of the unitary circuit trace estimation problem, the "standard" versions of which are QMA-complete [24] and $\mathsf{DQC1}$-complete [25, 42, 44], respectively. Interestingly, the "scaled-down" versions are both $\mathsf{BQ_UL}$-complete.

Recall that $\mathsf{DQC1} \subseteq \mathsf{BQP} \subseteq \mathsf{QMA}$. It seems reasonable to suspect that both of the preceding inclusions are proper. However, the fact that the "scaled-down" versions of a $\mathsf{DQC1}$-complete problem, a BQP-complete problem, and a QMA-complete are all $\mathsf{BQ_UL}$-complete shows that the differences between the time-bounded classes (if these classes are, indeed, different) disappear in the space-bounded setting.

In the following, we say that a function $f : D \to [-1,1]$, where $D \subseteq \mathbb{R}$, is *κ-Lipschitz* if $|f(x) - f(y)| \leq \kappa|x-y|, \forall x, y \in D$. For some $H \in \mathrm{Herm}(n)$, with eigendecomposition $H = \sum_{j=1}^{n} \lambda_j(H) v_j v_j^\dagger$, and for some function $f : D \to [-1,1]$, where $\{\lambda_j(H) : 1 \leq j \leq n\} \subseteq D$, let $f(H) = \sum_{j=1}^{n} f(\lambda_j(H)) v_j v_j^\dagger \in \mathrm{Herm}(n)$. We use the term L-transducer to refer to a deterministic logspace Turing machine that computes a function. As before, for some unitary circuit $Q = (Q_1, \ldots, Q_t)$, where each $Q_j \in \mathrm{U}(n)$ (i.e., Q_j is a unitary gate acting on $\log n$ qubits), we also use Q to denote the element $Q_t \cdots Q_1 \in \mathrm{U}(n)$ that corresponds to applying the gates $Q_1, \ldots, Q_t$ sequentially.

Definition 47. Consider functions $\kappa : \mathbb{N} \to \mathbb{R}_{\geq 1}$ and $\epsilon : \mathbb{N} \to \mathbb{R}_{>0}$.

(i) $\mathsf{MINEIGENVALUE}^{+}_{n,\epsilon^{-1}}$
Input: $H \in \widehat{\mathrm{Pos}}(n)$, $b \in [0,1]$.
Promise: $\lambda_n(H) \in [0,b] \cup [b+\epsilon(n), 1]$.
Output: 1 if $\lambda_n(H) \leq b$, 0 otherwise.

(ii) $\mathsf{MINEIGENVALUE}^{+,gap}_{n,\epsilon^{-1}}$
Input: $H \in \widehat{\mathrm{Pos}}(n)$, $b \in [0,1]$.
Promise: $\lambda_n(H) \in [0,b] \cup [b+\epsilon(n), 1]$, $\lambda_{n-1}(H) \geq b + \epsilon(n)$, $\lambda_{n-1}(H) - \lambda_n(H) \geq \epsilon(n)$.
Output: 1 if $\lambda_n(H) \leq b$, 0 otherwise.

(iii) $\mathsf{CIRCUITTRACE}_{n,\epsilon^{-1}}$
Input: $Q = (Q_1, \ldots, Q_t)$, where each $Q_j \in \mathrm{U}(n)$, $b \in [-1,1]$.
Promise: $\frac{1}{n}\mathrm{tr}(Q) \in [-1, b-\epsilon(n)] \cup [b, 1]$.
Output: 1 if $\frac{1}{n}\mathrm{tr}(Q) \geq b$, 0 otherwise.

(iv) $\mathsf{FILTEREDMATTRACE}_{n,\kappa,\epsilon^{-1}}$
Input: $H \in \widehat{\mathrm{Herm}}(n)$, $b \in [-1,1]$, M_f a L-transducer that computes $f : D \to [-1,1]$.
Promise: $\{\lambda_j(H) : 1 \leq j \leq n\} \subseteq D$, f is $\kappa(n)$-Lipschitz, $\frac{1}{n}\mathrm{tr}(f(H)) \in [-1, b-\epsilon(n)] \cup [b,1]$.
Output: 1 if $\frac{1}{n}\mathrm{tr}(f(H)) \geq b$, 0 otherwise.

(v) $\mathsf{MATINV}^{+,\geq 0}_{n,\kappa,\epsilon^{-1}}$
Input: $H \in \widehat{\mathrm{Pos}}(n)$, $s, t \in \{1, \ldots, n\}$, $b \in \mathbb{R}_{\geq 1+\epsilon(n)}$.
Promise: $\sigma_1(H) \leq 1$, $\sigma_n(H) \geq \frac{1}{\kappa(n)}$, $H^{-1}[s,t] \in [0, b-\epsilon(n)] \cup [b, \kappa(n)]$.
Output: 1 if $H^{-1}[s,t] \geq b$, 0 otherwise.

(vi) $\mathsf{SUMMATINV}^{\geq 0}_{n,\kappa,\epsilon^{-1}}$
Input: $H \in \widehat{\mathrm{Herm}}(n)$, $S \subseteq \{1, \ldots, n\}$, $b \in \mathbb{R}_{\geq 1+\epsilon(n)}$.
Promise: $\sigma_1(H) \leq 1$, $\sigma_n(H) \geq \frac{1}{\kappa(n)}$, $\det(H) > 0$, $\frac{1}{|S|}\sum_{(s,t)\in S^2} H^{-1}[s,t] \in [0, b-\epsilon(n)] \cup [b, \kappa(n)]$.
Output: 1 if $\frac{1}{|S|}\sum_{(s,t)\in S^2} H^{-1}[s,t] \geq b$, 0 otherwise.

Fefferman and Lin [18] showed that the *poly*-conditioned-$\mathsf{MINEIGENVALUE}^{+}$ problem and the *poly*-conditioned-$\mathsf{MATINV}^{+,\geq 0}$ problem are both $\mathsf{BQ_UL}$-complete. We next show, directly, that $\mathsf{MATINV}^{+,\geq 0} \leq^m_{\mathsf{AC}^0} \mathsf{MINEIGENVALUE}^{+,gap}$. This statement (combined with the fact that, trivially, $\mathsf{MINEIGENVALUE}^{+,gap} \leq^m_{\mathsf{AC}^0} \mathsf{MINEIGENVALUE}^{+}$) provides alternate proofs of the $\mathsf{BQ_UL}$-hardness of *poly*-conditioned-$\mathsf{MINEIGENVALUE}^{+}$ and of *poly*-conditioned-$\mathsf{MATINV}^{+,\geq 0} \in \mathsf{BQ_UL}$.

In particular, the previous proof of the $\mathsf{BQ_UL}$-hardness of *poly*-conditioned-$\mathsf{MINEIGENVALUE}^+$ [18, Lemma 21] required the use of "strong" in-place error reduction for $\mathsf{BQ_UL}$ [18, Lemma 28], whereas *poly*-conditioned-$\mathsf{MATINV}^{+,\geq 0}$ was shown to be $\mathsf{BQ_UL}$-hard by a much simpler argument that did not require the use of any such error reduction machinery [18, Theorem 13]. As a consequence of our direct proof of the fact that $\mathsf{MATINV}^{+,\geq 0} \leq^m_{\mathsf{AC}^0} \mathsf{MINEIGENVALUE}^{+,gap}$, we then obtain a simple proof of the $\mathsf{BQ_UL}$-hardness of *poly*-conditioned-$\mathsf{MINEIGENVALUE}^+$.

We then study $\mathsf{CIRCUITTRACE}$ and $\mathsf{FILTEREDMATTRACE}$. We show that the *poly*-conditioned versions of both problems are $\mathsf{BQ_UL}$-complete. In particular, we show $\mathsf{MINEIGENVALUE}^{+,gap} \leq^m_{\mathsf{AC}^0}$ $\mathsf{FILTEREDMATTRACE}$, which, combined with the above results, provides an extremely simple proof of the fact that *poly*-conditioned-$\mathsf{MINEIGENVALUE}^{+,\mathsf{gap}}$, *poly*-conditioned-$\mathsf{MATINV}^{+,\geq 0} \in \mathsf{BQ_UL}$. Note that $\mathsf{FILTEREDMATTRACE}$ is a "scaled-down" version of a problem that was shown to be in $\mathsf{DQC1}$ by Cade and Montanaro [10]; their technique has many common elements with the technique used earlier by Harrow, Hassidim, and Lloyd [20] in their proof of the fact that "scaled-up" matrix inversion is in BQP. We use the term "filtered" in reference to the "filter functions" that appear in these proofs.

Lemma 48. $\mathsf{MATINV}^{+,\geq 0} \leq^m_{\mathsf{AC}^0} \mathsf{SUMMATINV}^{\geq 0}$.

Proof. Consider $\langle H, s, t, b\rangle \in \mathsf{MATINV}^{+,\geq 0}_{n,\kappa,\epsilon^{-1}}$. First, suppose $s = t$; then $\langle H, \{s\}, b\rangle \in \mathsf{SUMMATINV}^{\geq 0}_{n,\kappa,\epsilon}$ and $\mathsf{MATINV}(\langle H, s, s, b\rangle) = \mathsf{SUMMATINV}(\langle H, \{s\}, b\rangle)$. Next, suppose instead that $s \neq t$. If n is even, let $\widehat{A} = H$ and $m = n$; if n is odd, let $\widehat{A} = H \oplus I_1$ and $m = n + 1$. Let $\widehat{H} = \begin{pmatrix} 0_m & \widehat{A}^\dagger \\ \widehat{A} & 0_m \end{pmatrix} \in \widehat{\mathrm{Herm}}(2m)$, $\widehat{S} = \{s, t\}$, and $\widehat{b} = b$. Then $\langle \widehat{H}, \widehat{S}, \widehat{b}\rangle \in \mathsf{SUMMATINV}^{\geq 0}_{m,\kappa(n),\epsilon^{-1}(n)}$ and $\mathsf{MATINV}(\langle H, s, t, b\rangle) = \mathsf{SUMMATINV}(\langle \widehat{H}, \widehat{S}, \widehat{b}\rangle)$. $\square$

Lemma 49. $\mathsf{SUMMATINV}^{\geq 0} \leq^m_{\mathsf{AC}^0} \mathsf{MINEIGENVALUE}^{+,gap}$.

Proof. Consider $\langle H, S, b\rangle \in \mathsf{SUMMATINV}^{\geq 0}_{n,\kappa,\epsilon^{-1}}$. Let $v = \frac{1}{\sqrt{|S|}} \sum_{s\in S} \chi_s$, $\widehat{C} = \frac{2}{2b-\epsilon(n)} vv^\dagger \in \widehat{\mathrm{Herm}}(n)$, $\widehat{A} = H - \widehat{C} \in \widehat{\mathrm{Herm}}(n)$, and $x = v^\dagger H^{-1} v = \frac{1}{|S|} \sum_{(s,t)\in S^2} H^{-1}[s,t]$. By the matrix determinant lemma, $\det(\widehat{A}) = (1 - \frac{2x}{2b-\epsilon(n)})$. If $x \geq b$, then $\det(\widehat{A}) \leq \frac{-\epsilon(n)}{2\kappa(n)} \det(H)$; if $x \leq b - \epsilon(n)$, then $\det(\widehat{A}) \geq \frac{\epsilon(n)}{2\kappa(n)} \det(H)$.

By the Weyl eigenvalue inequalities, for any $j \in \{1, \dots, n-1\}$, $\lambda_j(\widehat{A}) \geq \lambda_{j+1}(H) - \lambda_2(\widehat{C}) = \lambda_{j+1}(H) > 0$ and $\lambda_j(\widehat{A}) \leq \lambda_j(H) + \lambda_1(-\widehat{C}) = \lambda_j(H) \leq 1$. This implies,

$$\det(H)\kappa(n) \geq \det(H)\lambda_n^{-1}(H) = \prod_{j=1}^{n-1} \lambda_j(H) \geq \prod_{j=1}^{n-1} \lambda_j(\widehat{A}) \geq \prod_{j=1}^{n-1} \lambda_{j+1}(H) = \det(H)\lambda_1^{-1}(H) \geq \det(H) > 0.$$

Clearly, $\det(\widehat{A}) = \prod_{j=1}^{n} \lambda_j(\widehat{A}) = \lambda_n(\widehat{A}) \prod_{j=1}^{n-1} \lambda_j(\widehat{A})$. Therefore, if $\det(\widehat{A}) \leq \frac{-\epsilon(n)}{2\kappa(n)} \det(H)$, then $\lambda_n(\widehat{A}) \leq \frac{-\epsilon(n)}{2\kappa(n)}$; similarly, if $\det(\widehat{A}) \geq \frac{\epsilon(n)}{2\kappa(n)} \det(H)$, then $\lambda_n(\widehat{A}) \geq \frac{\epsilon(n)}{2\kappa(n)^2}$. By the promise, $b \geq 1 + \epsilon(n)$, which implies $\lambda_n(\widehat{A}) \geq \lambda_n(H) + \lambda_n(-\widehat{C}) = \lambda_n(H) - \frac{2}{2b-\epsilon(n)} \geq -1$. Moreover, $\lambda_n(\widehat{A}) \leq \lambda_n(H) \leq 1$.

Let $\widehat{H} = \frac{1}{2}(I_{n+1} + (\widehat{A} \oplus 0_1)) \in \widehat{\mathrm{Pos}}(n+1)$, where 0_1 denotes the 1×1 zero matrix. By the above, $\lambda_{n+1}(\widehat{H}) \in \left[0, \frac{1}{2}\left(1 - \frac{\epsilon(n)}{2\kappa(n)}\right)\right] \cup \left\{\frac{1}{2}\right\}$, $\lambda_n(H) - \lambda_{n+1}(H) \geq \frac{\epsilon(n)}{2\kappa(n)}$, and $\lambda_{n+1}(\widehat{H}) \in \left[0, \frac{1}{2}\left(1 - \frac{\epsilon(n)}{2\kappa(n)}\right)\right] \Leftrightarrow x \geq b$. Let $\widehat{b} = \frac{1}{2}\left(1 - \frac{\epsilon(n)}{2\kappa(n)}\right)$. Then $\langle \widehat{H}, \widehat{b}\rangle \in \mathsf{MINEIGENVALUE}^{+,gap}_{n+1,4\kappa(n)\epsilon^{-1}(n)}$ and $\mathsf{SUMMATINV}(\langle H, S, b\rangle) = \mathsf{MINEIGENVALUE}(\langle \widehat{H}, \widehat{b}\rangle)$. $\square$

Lemma 50. *poly-conditioned-*$\mathsf{MINEIGENVALUE}^{+,gap}$ *is* $\mathsf{BQ_UL}$*-complete.*

Proof. By [18, Theorem 13], *poly*-conditioned-$\mathsf{MATINV}^{+,\geq 0}$ is $\mathsf{BQ_UL}$-hard, which, by Lemmas 48 and 49, implies *poly*-conditioned-$\mathsf{MINEIGENVALUE}^{+,gap}$ is $\mathsf{BQ_UL}$-hard. By [18, Lemmas 19 and 20], *poly*-conditioned-$\mathsf{MINEIGENVALUE}^{+} \in \mathsf{BQ_UL}$; trivially, $\mathsf{MINEIGENVALUE}^{+,gap} \leq^m_{\mathsf{AC}^0} \mathsf{MINEIGENVALUE}^{+}$, which implies *poly*-conditioned-$\mathsf{MINEIGENVALUE}^{+,gap} \in \mathsf{BQ_UL}$. ☐

Lemma 51. *poly-conditioned-*$\mathsf{CIRCUITTRACE}$ *is* $\mathsf{BQ_UL}$*-complete.*

Proof. Follows straightforwardly from the same technique used in the "standard" proof that the "scaled-up" circuit trace estimation problem is $\mathsf{DQC1}$-complete [25, 42, 44]; we omit the details. ☐

Lemma 52. $\mathsf{MINEIGENVALUE}^{+,gap} \leq^m_{\mathsf{AC}^0} \mathsf{FILTEREDMATTRACE}$

Proof. Consider $\langle H, b\rangle \in \mathsf{MINEIGENVALUE}^{+,gap}_{n,\epsilon^{-1}}$. Let $\widehat{D} = [0, b] \cup [b + \epsilon(n), 1]$ and let $\widehat{f} : \widehat{D} \to \{0, 1\}$ be defined such that $\widehat{f}(x) = 1$ if $x \leq b$, and $\widehat{f}(x) = 0$ if $x \geq b + \epsilon(n)$. Clearly, $\{\lambda_j(H) : 1 \leq j \leq n\} \subseteq \widehat{D}$ and $\widehat{f}$ is $\epsilon^{-1}(n)$-Lipschitz. Moreover,

$$\operatorname{tr}(\widehat{f}(H)) = \sum_{j=1}^{n} \widehat{f}(\lambda_j(H)) = |\{j : \lambda_j(H) \leq b\}| = \begin{cases} 1, & \lambda_n(H) \leq b \\ 0, & \text{otherwise.} \end{cases}$$

Let $\widehat{b} = \frac{1}{n}$. Then $\frac{1}{n}\operatorname{tr}(\widehat{f}(H)) \in \{0, \widehat{b}\}$ and $\frac{1}{n}\operatorname{tr}(\widehat{f}(H)) = \widehat{b} \Leftrightarrow \lambda_n(H) \leq b$. Thus, $\langle \widehat{H}, \widehat{b}, M_{\widehat{f}}\rangle \in \mathsf{FILTEREDMATTRACE}_{n,\epsilon^{-1},\epsilon^{-1}}$ and $\mathsf{MINEIGENVALUE}(\langle H, b\rangle) = \mathsf{FILTEREDMATTRACE}(\langle \widehat{H}, \widehat{b}, M_{\widehat{f}}\rangle)$. ☐

Lemma 53. *poly-conditioned-*$\mathsf{FILTEREDMATTRACE}$ *is* $\mathsf{BQ_UL}$*-complete.*

Proof. By Lemmas 50 and 52, *poly*-conditioned-$\mathsf{FILTEREDMATTRACE}$ is $\mathsf{BQ_UL}$-hard. Moreover, *poly*-conditioned-$\mathsf{FILTEREDMATTRACE} \in \mathsf{BQ_UL}$ follows, straightforwardly, from the same technique used to prove [10, Lemma 1] (which shows that a "scaled-up" version of this problem is in $\mathsf{DQC1}$); we omit the details. ☐

6 Discussion

We conclude by stating a few interesting open problems related to our work. In Theorem 1 we established the equivalence of unitary quantum space, general quantum space, and space-bounded quantum Merlin-Arthur proof systems, in the two-sided bounded-error case. We obtained an analogous equivalence for one-sided unbounded-error in Theorem 3. However, in the case of one-sided bounded-error, we only have the partial results of Theorem 2. In particular, specializing to the case of logspace, we have $\mathsf{BQL} = \mathsf{BQ_UL} = \mathsf{QMAL}$ in the two-sided bounded-error case (Theorem 27), and we have $\mathsf{RQMAL} = \mathsf{RQ_UL} \subseteq \mathsf{RQL} \subseteq \mathsf{coQMAL_1}$ in the one-sided bounded-error case (Theorem 41). It is naturally to ask if the analogues of results known to hold for two-sided bounded-error also hold for one-sided bounded-error.

Open Problem 1. Is $\mathsf{RQ_UL} = \mathsf{RQL}$? Is $\mathsf{RQL} = \mathsf{coQMAL_1}$?

By the well-known result of Zachos and Fürer [55], $\mathsf{MA} = \mathsf{MA_1}$; that is to say, it is possible to achieve perfect completeness for *classical* (polynomial time) Merlin-Arthur proof systems. On the other hand, the question of whether or not it is possible to achieve perfect completeness for *quantum* (polynomial time) Merlin-Arthur proof systems (i.e., is $\mathsf{QMA} = \mathsf{QMA_1}$?) remains open (see, for instance, [1, 3, 9, 22] for previous discussion). We next consider the logspace analogue of this question.

Open Problem 2. Is $\mathsf{QMAL} = \mathsf{QMAL_1}$?

A possible explanation for the difficulty of proving $\mathsf{QMA} = \mathsf{QMA_1}$ (if these classes are indeed equal) was provided by Aaronson's result [1] that there is a *quantum* oracle $\mathcal{U}$ such that $\mathsf{QMA}^{\mathcal{U}} \neq \mathsf{QMA_1}^{\mathcal{U}}$; therefore, any proof of $\mathsf{QMA} = \mathsf{QMA_1}$ must use a technique that is *quantumly nonrelativizing*. Note that the technique used by Zachos and Fürer [55] to show $\mathsf{MA} = \mathsf{MA_1}$ is (classically) relativizing. It is not hard to see that Aaronson's argument can also be used to produce a quantum oracle $\mathcal{U}$ such that $\mathsf{QMAL}^{\mathcal{U}} \neq \mathsf{QMAL_1}^{\mathcal{U}}$, and so any proof of $\mathsf{QMAL} = \mathsf{QMAL_1}$ must also use quantumly nonrelativizing techniques. We emphasize that the techniques used in this paper to show our results concerning new inclusions between various complexity classes (i.e., the various reductions between linear-algebraic problems shown in this paper) are *quantumly nonrelativizing.*

Moreover, it is known that it is possible to achieve perfect completeness in quantum Merlin-Arthur proof systems that have a *classical witness*; that is to say, $\mathsf{QCMA} = \mathsf{QCMA_1}$ [22]. Note that, trivially, $\mathsf{BQ_UL} \subseteq \mathsf{QCMAL} \subseteq \mathsf{QMAL}$. Thus, the known equality $\mathsf{BQ_UL} = \mathsf{QMAL}$ immediately implies $\mathsf{QCMAL} = \mathsf{QMAL}$. Therefore, $\mathsf{QMAL} = \mathsf{QMAL_1} \Leftrightarrow \mathsf{QCMAL} = \mathsf{QMAL_1} \Leftarrow \mathsf{QCMAL} = \mathsf{QCMAL_1}$.

Returning to an issue discussed in Section 1.1, Boix-Adserà, Eldar, and Mehraban [8] showed $\kappa(n)$-conditioned-$\mathsf{DET} \in \mathsf{DSPACE}(\log(n)\log(\kappa(n))poly(\log\log n))$. Moreover, they asked the question "is *poly*-conditioned-DET BQL-complete?" In this paper, we answered their question in the affirmative. Therefore, if $\mathsf{BQL} \nsubseteq \mathsf{DSPACE}(\log^{2-\epsilon} n)$, $\forall \epsilon > 0$, then both our result and the result of Boix-Adserà, Eldar, and Mehraban are essentially optimal (in terms of the relationship between condition number and needed space). Recall that, by the classic result of Watrous [51], $\mathsf{BQL} \subseteq \mathsf{DSPACE}(\log^2 n)$.

Moreover, we note that our paper and that of Boix-Adserà, Eldar, and Mehraban used similar power series techniques to produce space-efficient algorithms for $\kappa(n)$-conditioned-DET. However, our quantum algorithm can make use of a power series with an *exponentially* larger number of terms than seems possible for their (or any other) classical algorithm. This suggests a possible mechanism for explaining the supposed advantage of quantum computers over classical computers in the space-bounded setting. We conclude with a general question.

Open Problem 3. What further relationships can be established between BQL and other natural logspace complexity classes (e.g., $\#\mathsf{L}, \mathsf{GapL}, \mathsf{L}/poly$, etc.)?

Acknowledgments

We would like to thank Dieter van Melkebeek for many helpful comments on a preliminary draft.

References

[1] Scott Aaronson. On perfect completeness for qma. *Quantum Information and Computation*, 9(1):0081–0089, 2009.

[2] Dorit Aharonov, Alexei Kitaev, and Noam Nisan. Quantum circuits with mixed states. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 20–30, 1998.

[3] Dorit Aharonov and Tomer Naveh. Quantum np-a survey. *arXiv preprint quant-ph/0210077*, 2002.

[4] Eric Allender, Vikraman Arvind, and Meena Mahajan. Arithmetic complexity, kleene closure, and formal power series. *Theory of Computing Systems*, 36(4):303–328, 2003.

[5] Eric Allender and Mitsunori Ogihara. Relationships among pl, #l, and the determinant. *RAIRO-Theoretical Informatics and Applications*, 30(1):1–21, 1996.

[6] Charles H Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM journal on Computing*, 26(5):1510–1523, 1997.

[7] Dominic W Berry, Andrew M Childs, Richard Cleve, Robin Kothari, and Rolando D Somma. Exponential improvement in precision for simulating sparse hamiltonians. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 283–292, 2014.

[8] Enric Boix-Adserà, Lior Eldar, and Saeed Mehraban. Approximating the determinant of well-conditioned matrices by shallow circuits. *arXiv preprint arXiv:1912.03824*, 2019.

[9] Sergey Bravyi. Efficient algorithm for a quantum analogue of 2-sat. 2011.

[10] Chris Cade and Ashley Montanaro. The quantum complexity of computing schatten p-norms. In *13th Conference on the Theory of Quantum Computation, Communication and Cryptography*, 2018.

[11] Andrew M Childs. On the relationship between continuous-and discrete-time quantum walk. *Communications in Mathematical Physics*, 294(2):581–603, 2010.

[12] Stephen A Cook. A taxonomy of problems with fast parallel algorithms. *Information and Control*, 64(1-3):2–22, 1985.

[13] C Damm. Det= l# l. informatik-preprint 8. *Fachbereich Informatik der Humboldt-Universitat zu, Berlin*, 1991.

[14] David P DiVincenzo. The physical implementation of quantum computation. *Fortschritte der Physik: Progress of Physics*, 48(9-11):771–783, 2000.

[15] Dean Doron, Amir Sarid, and Amnon Ta-Shma. On approximating the eigenvalues of stochastic matrices in probabilistic logspace. *computational complexity*, 26(2):393–420, 2017.

[16] Dean Doron and Amnon Ta-Shma. On the de-randomization of space-bounded approximate counting problems. *Information Processing Letters*, 115(10):750–753, 2015.

[17] Bill Fefferman, Hirotada Kobayashi, Cedric Yen-Yu Lin, Tomoyuki Morimae, and Harumichi Nishimura. Space-efficient error reduction for unitary quantum computations. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[18] Bill Fefferman and Cedric Yen-Yu Lin. A complete characterization of unitary quantum space. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[19] Richard P Feynman. Simulating physics with computers. *Int. J. Theor. Phys*, 21(6/7), 1999.

[20] Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical review letters*, 103(15):150502, 2009.

[21] Mark Jerrum, Alistair Sinclair, and Eric Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with nonnegative entries. *Journal of the ACM (JACM)*, 51(4):671–697, 2004.

[22] Stephen P Jordan, Hirotada Kobayashi, Daniel Nagaj, and Harumichi Nishimura. Achieving perfect completeness in classical-witness quantum merlin-arthur proof systems. *Quantum Information & Computation*, 12(5-6):461–471, 2012.

[23] Richard Jozsa, Barbara Kraus, Akimasa Miyake, and John Watrous. Matchgate and space-bounded quantum computations are equivalent. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 466(2115):809–830, 2010.

[24] Aleksei Yur'evich Kitaev. Quantum computations: algorithms and error correction. *Uspekhi Matematicheskikh Nauk*, 52(6):53–112, 1997.

[25] Emanuel Knill and Raymond Laflamme. Power of one bit of quantum information. *Physical Review Letters*, 81(25):5672, 1998.

[26] Hirotada Kobayashi, Keiji Matsumoto, and Tomoyuki Yamakami. Quantum merlin-arthur proof systems: Are multiple merlins more helpful to arthur? In *International Symposium on Algorithms and Computation*, pages 189–198. Springer, 2003.

[27] Rolf Landauer. Irreversibility and heat generation in the computing process. *IBM journal of research and development*, 5(3):183–191, 1961.

[28] Seth Lloyd. Universal quantum simulators. *Science*, pages 1073–1078, 1996.

[29] Meena Mahajan and V Vinay. Determinant: Combinatorics, algorithms, and complexity. *Chicago Journal of Theoretical Computer Science*, 1997.

[30] Chris Marriott and John Watrous. Quantum arthur–merlin games. *Computational Complexity*, 14(2):122–152, 2005.

[31] Dieter van Melkebeek and Thomas Watson. Time-space efficient simulations of quantum computations. *Theory of Computing*, 8(1):1–51, 2012.

[32] Daniel Nagaj, Pawel Wocjan, and Yong Zhang. Fast amplification of qma. *Quantum Information & Computation*, 9(11):1053–1068, 2009.

[33] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

[34] Noam Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.

[35] Simon Perdrix and Philippe Jorrand. Classically controlled quantum computation. *Mathematical Structures in Computer Science*, 16(4):601–620, 2006.

[36] Zachary Remscrim. Lower bounds on the running time of two-way quantum finite automata and sublogarithmic space quantum turing machines. *Electronic Colloquium on Computational Complexity (ECCC)*, 26:182, 2019.

[37] Zachary Remscrim. The power of a single qubit: Two-way quantum finite automata and the word problem. *Electronic Colloquium on Computational Complexity (ECCC)*, pages 139:1–139:18, 2020.

[38] Wojciech Roga, Zbigniew Puchala, Lukasz Rudnicki, and Karol Zyczkowski. Entropic trade-off relations for quantum operations. *Physical Review A*, 87(3):032308, 2013.

[39] Michael Saks. Randomization and derandomization in space-bounded computation. In *Proceedings of Computational Complexity (Formerly Structure in Complexity Theory)*, pages 128–149. IEEE, 1996.

[40] Michael Saks and Shiyu Zhou. Bphspace (s)/spl sube/dspace (s3/2). *Journal of computer and system sciences*, 58(2):376–403, 1999.

[41] Miklos Santha and Sovanna Tan. Verifying the determinant in parallel. *Computational Complexity*, 7(2):128–151, 1998.

[42] Dan Shepherd. Computation with unitaries and one pure qubit. *arXiv preprint quant-ph/0608132*, 2006.

[43] Peter W Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994.

[44] Peter W Shor and Stephen P Jordan. Estimating jones polynomials is a complete problem for one clean qubit. *Quantum Information & Computation*, 8(8):681–714, 2008.

[45] Amnon Ta-Shma. Inverting well conditioned matrices in quantum logspace. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 881–890, 2013.

[46] Seinosuke Toda. Counting problems computationally equivalent to the determinant, 1991.

[47] Leslie G Valiant. Why is boolean complexity theory difficult. *Boolean Function Complexity*, 169:84–94, 1992.

[48] V Vinay. Counting auxiliary pushdown automata and semi-unbounded arithmetic circuits. In *1991 Proceedings of the Sixth Annual Structure in Complexity Theory Conference*, pages 270–284. IEEE.

[49] John Watrous. Space-bounded quantum complexity. *Journal of Computer and System Sciences*, 59(2):281–326, 1999.

[50] John Watrous. Quantum simulations of classical random walks and undirected graph connectivity. *Journal of computer and system sciences*, 62(2):376–391, 2001.

[51] John Watrous. On the complexity of simulating space-bounded quantum computations. *Computational Complexity*, 12(1-2):48–84, 2003.

[52] John Watrous. Encyclopedia of complexity and system science, chapter quantum computational complexity, 2009.

[53] John Watrous. Zero-knowledge against quantum attacks. *SIAM Journal on Computing*, 39(1):25–58, 2009.

[54] John Watrous. *The theory of quantum information*. Cambridge University Press, 2018.

[55] Stathis Zachos and Martin Furer. Probabilistic quantifiers vs. distrustful adversaries. In *International Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 443–455. Springer, 1987.

A A QTM-based Proof of BQL=$\mathbf{BQ_U L}$

In this appendix, we show that the technique that was used to show $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$, in Proposition 25, also immediately shows $\mathsf{BQL} \subseteq \mathsf{BQ_UL}$ (and, therefore, $\mathsf{BQL} = \mathsf{BQ_UL}$). Here, BQL is defined in terms of a logspace *quantum Turing machine* (QTM), as was the case in, for instance [23,31,35,45,49–52], rather than the equivalent model of an L-uniform family of *general quantum circuits* used in this paper.

For concreteness, we use the *classically controlled* logspace (general) QTM defined by Watrous [51] (with the minor alteration that we require all transition amplitudes of the QTM to be computable in L); however, we note that our result would apply equally well to any "reasonable" logspace QTM model that is classically controlled (this includes all models considered in all of the papers cited above). In brief, such a QTM M consists of a (classical) finite control, an internal quantum register of constant size, a classical "measurement" register of constant size, and three tapes: (1) a read-only input tape that, on any input w, contains the string $\#_L w \#_R$, where $\#_L$ and $\#_R$ are special symbols that serve as left and right end-markers, (2) a read/write classical work tape consisting of $s(|w|) = O(\log|w|)$ cells, each of which holds a symbol from some finite alphabet Γ, and (3) a read/write quantum work tape, consisting of $s(|w|) = O(\log|w|)$ qubits. Each of the tapes has a single bidirectional head. At the start of the computation, both work-tapes are "blank" (to be precise, each cell of the classical work tape contains some specified blank-symbol in Γ and each qubit of the quantum work tape is in the state $|0\rangle$); each qubit of the internal quantum register is also in the state $|0\rangle$. Each step of the computation of M involves applying a *selective quantum operation* to the combined register consisting of the internal quantum register and the single qubit that is currently under the head of the quantum work tape; the particular choice of which selective quantum operation to perform may depend on the state of the finite control and the symbols currently under the heads of the input tape and classical work tape. The (classical) result of this quantum operation is stored in the measurement register. Then, depending on this result, as well as on the state of the finite control and the symbols currently under the heads of the input tape and classical work tape, the classical configuration of the machine evolves; to be precise, the state of the finite control is updated, a symbol is written on the classical work-tape, and the head of each work tape moves up to one cell in either direction. The machine accepts (resp.) rejects its input by entering a special (classical) accepting (resp. rejecting state). See [51] for a complete definition.

Proposition 54. $\mathsf{BQL} = \mathsf{BQ_UL}$.

Proof. Trivially, $\mathsf{BQL} \supseteq \mathsf{BQ_UL}$. We next show $\mathsf{BQL} \subseteq \mathsf{BQ_UL}$. Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{BQL}$. By definition, there is some QTM M such that the following conditions are satisfied: (1) on any input $w \in \mathsf{P}$ of length $n = |w|$, M runs in space at most $s(n) = O(\log n)$ (and hence time $t(n) = 2^{O(s(n))}$), (2) if $w \in \mathsf{P}_1$, then $\Pr[M \text{ accepts } w] \geq \frac{2}{3}$, and (3) if $w \in \mathsf{P}_0$, then $\Pr[M \text{ accepts } w] \leq \frac{1}{3}$.

Consider running M on some input of length n. At any particular point in time, the configuration of (a single probabilistic branch of) M consists of the current (classical) state of the finite control, the (quantum) contents of the internal quantum register, the (classical) contents of the measurement register, the (classical) positions of the heads on the read only input-tape and the classical and quantum work-tapes, the current (classical) contents of the classical work-tape, and the current (quantum) contents of the quantum work-tape. Let $|M|$ denote the size of the finite control, let b_m denote the number of bits of the measurement register, let b_q denote the number of qubits of the internal quantum register, and let Γ denote the classical work-tape alphabet. Let C_n denote the set of all possible classical configurations of M on inputs of length n, where $|C_n| = |M|2^{b_m}(n+2)s(n)^2|\Gamma|^{s(n)} = n^{O(1)}$. Each classical configuration $c \in C_n$ corresponds to the element $|c\rangle$ in the natural orthonormal basis of the Hilbert space $\mathbb{C}^{C_n}$. Let Q_n denote the set of $|Q_n| = 2^{s(n)+b_q} = n^{O(1)}$ quantum basis states corresponding to the quantum work-tape and internal quantum register. The contents of the quantum work-tape and the internal quantum register is then described by some $|\psi\rangle \in \mathbb{C}^{Q_n}$. Then each configuration of M on an input of length n corresponds to an element $|c\rangle|\psi\rangle$ of the Hilbert space $\mathcal{H}_{M,n} = \mathbb{C}^{C_n} \otimes \mathbb{C}^{Q_n}$. Let $d(n) = \dim(\mathcal{H}_{M,n}) = |C_n||Q_n| = n^{O(1)}$.

Consider some input $w \in \mathsf{P}$. Let $n = |w|$ denote the length of w, let $\Phi_{M,w} \in \mathrm{Chan}(\mathcal{H}_{M,n})$ denote the quantum channel that corresponds to a single step of the computation of M on w, and let $K(\Phi_{M,w}) \in \widehat{\mathrm{Mat}}(d^2(n))$ denote the natural representation of $\Phi_{M,w}$. For any $t \in \mathbb{N}$, we have $\Phi^t_{M,w} \in \mathrm{Chan}(\mathcal{H}_{M,n})$, which implies $\sigma_1((K(\Phi_{M,w}))^t) = \sigma_1(K(\Phi^t_{M,w})) \leq \sqrt{d(n)} = n^{O(1)}$ [38, Theorem 1]. Let $|\psi^n_{start}\rangle = |c^n_{start}\rangle|q^n_{start}\rangle \in \mathcal{H}_{M,n}$ denote the starting configuration of M on an input of length n, where $c^n_{start} \in C_n$ is the classical part of the starting configuration, and $|q^n_{start}\rangle = |0^{s(n)+b_q}\rangle \in \mathbb{C}^{Q_n}$ is the quantum part. Without loss of generality we may, for convenience, assume that M "cleans-up" its workspace at the end of the computation, by returning both its classical and quantum work tapes to the "blank" configuration described above; in particular, this implies that M has a unique accepting configuration $|\psi^n_{accept}\rangle = |c^n_{accept}\rangle|q^n_{start}\rangle \in \mathcal{H}_{M,n}$ on any input of length n. Let $A_w = K(\Phi_{M,w}) \in \widehat{\mathrm{Mat}}(d^2(n))$, $x_w = \mathrm{vec}(|\psi^n_{accept}\rangle\langle\psi^n_{accept}|) \in \{1, \ldots, d^2(n)\}$, and $y_w = \mathrm{vec}(|\psi^n_{start}\rangle\langle\psi^n_{start}|) \in \{1, \ldots, d^2(n)\}$. Then $A^t_w[x_w, y_w]$ is precisely the probability that M accepts w within t steps.

Therefore, $\langle A_w, x_w, y_w, \frac{2}{3}\rangle \in \mathsf{MATPOW}_{d^2(n),t(n),\sqrt{d(n)},3}$ and $\mathsf{MATPOW}(\langle A_w, x_w, y_w, \frac{2}{3}\rangle) = \mathsf{P}(w)$. By Theorem 4, $\mathsf{MATPOW}_{d^2(n),t(n),\sqrt{d(n)},3} \in \mathsf{BQ_UL}$, which implies $\mathsf{P} \in \mathsf{BQ_UL}$. □

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il