Eliminating Intermediate Measurements in Space-Bounded Quantum Computation

Bill Fefferman
The University of Chicago
Chicago, Illinois, USA
wjf@uchicago.edu

Zachary Remscrim*
The University of Chicago
Chicago, Illinois, USA
remscrim@uchicago.edu

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 eliminate intermediate measurements 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. In particular, this shows that the definition of a space-bounded quantum complexity class is robust to allowing or forbidding intermediate measurements. 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.

CCS CONCEPTS

• **Theory of computation → Quantum complexity theory**.

KEYWORDS

quantum computation, space complexity, approximation algorithms, algorithms for linear algebra

ACM Reference Format:

Bill Fefferman and Zachary Remscrim. 2021. Eliminating Intermediate Measurements in Space-Bounded Quantum Computation. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC '21), June 21–25, 2021, Virtual, Italy*. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3406325.3451051

*Corresponding author; portions of this research were completed while a member of the Department of Mathematics at MIT.

STOC '21, June 21–25, 2021, Virtual, Italy

ACM ISBN 978-1-4503-8053-9/21/06…$15.00
https://doi.org/10.1145/3406325.3451051

1 INTRODUCTION

Quantum computation has the potential to yield dramatic speedups for important problems such as quantum simulation (see, e.g., [20, 30]) and integer factorization [44]. While fault-tolerant, fully scalable 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* eliminate intermediate measurements in a quantum computation (see e.g., [18, 25, 33, 37, 46, 50–52]). While a classic result known as the "principle of safe storage"[1] states that it is always possible to *time-efficiently* defer intermediate measurements to the end of a computation [2, 35], 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 first main result solves this problem. We show that every problem solvable by a *general* quantum algorithm, which may make arbitrary use of quantum measurements, can also be solved, using the same amount of space, by a *unitary* quantum algorithm, which may not perform any intermediate measurements. As an immediate corollary, this shows that, in the space-bounded setting, unitary quantum algorithms are at least as powerful as probabilistic algorithms, resolving a longstanding open problem [33, 51].

In the process of proving this result, we also obtain the following result, which is likely of independent interest: approximating the solution of the "well-conditioned" versions of various standard linear-algebraic problems, such as the determinant problem, the matrix inversion problem, or the matrix powering problem, is complete for the class of bounded-error logspace quantum algorithms. These are a new class of natural problems on which quantum computers seem to outperform their classical counterparts.

[1] The "principle of deferred measurement" is another common name for this result.

1.1 Eliminating Intermediate Measurements

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 qubits to their initial states (for a discussion of these issues from an experimental perspective see, e.g., [14]). Secondly, unitary computations are *reversible*, whereas quantum measurement is an inherently irreversible process. The ability to "undo" a unitary subroutine, by running it in reverse, is routinely used in the design and analysis of quantum algorithms [5, 17, 18, 32, 34, 45, 54]. Moreover, reversible computations may be performed without generating heat [28]. Thirdly, by demonstrating that unitary quantum space and general quantum space are equivalent in power, we show that the definition of quantum space is quite *robust.* Allowing intermediate measurements, or even general quantum operations, does not provide any additional power in the space-bounded setting.

Let $\mathsf{BQ_USPACE}(s)$ (resp. $\mathsf{BQSPACE}(s)$) denote the set 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, [33, 41, 50, 52] for the importance of this restriction in quantum and/or probabilistic space-bounded computation. Furthermore, let $\mathsf{Q_UMASPACE}(s)$ (resp. $\mathsf{QMASPACE}(s)$) denote those promise problems recognized by a unitary (resp. general) quantum Merlin-Arthur protocol that operates in space $O(s(n))$ and time $2^{O(s(n))}$. An equivalent definition of these complexity classes may be given using quantum Turing machines; see Section 2.2 for further details.

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*

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

Remark. Note that $\mathsf{BPSPACE}(s) \subseteq \mathsf{BQ_USPACE}(s)$ was not previously known to hold, where $\mathsf{BPSPACE}(s)$ denotes the analogously defined class of language recognizable by a *probabilistic* algorithm in space $O(s)$ (and time $2^{O(s)}$); see, for instance, [33, 51] for 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 $\mathsf{BPSPACE}(s) \subseteq \mathsf{BQSPACE}(s)$, we have, as a corollary of Theorem 1, that $\mathsf{BPSPACE}(s) \subseteq \mathsf{BQ_USPACE}(s)$, resolving this question.

Remark. To further clarify the parameters of our result, given a *general* quantum algorithm that operates in space s and time t, we produce a *unitary* quantum algorithm that operates in space $O(s+\log t)$ and time $poly(t2^s)$. Note that these parameters coincide with that of the space-efficient simulation of a deterministic algorithm by a (classical) reversible algorithm [29]. Further note that in the extreme (but natural) setting in which $t = 2^{O(s)}$ (e.g. quantum logspace), this procedure is simultaneously space-efficient and time-efficient. On the other hand, in the opposite extreme setting in which $t = poly(s)$, this procedure is no longer time-efficient; of course, in this setting, the standard "principle of deferred measurement" is simultaneously space-efficient and time-efficient. Between these two extremes, one has time-space tradeoffs analogous to those of the simulation of deterministic algorithms by reversible algorithms [10].

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 establish analogous results concerning the relationship between the unitary and general versions of these classes; see Section 5 for a formal statement of these results.

1.2 Exact and Approximate Linear Algebra

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. 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. Before our work, the following relationships were known [50, 52]: $\mathsf{BQ_UL} \subseteq \mathsf{BQL} \subseteq \mathsf{DET}^*$ and $\mathsf{BPL} \subseteq \mathsf{BQL}$. Many natural linear-algebraic problems are DET^*-complete, such as intDET, $\mathsf{intMATINV}$ (the problem of computing a single entry of the inverse of a matrix), and $\mathsf{intITMATPROD}$ (the problem of computing a single entry of the product of polynomially-many matrices). 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 such linear-algebraic problems. Let *poly*-conditioned-MATINV denote the promise problem of approximating, to additive $1/poly(n)$ precision, a single entry of the inverse of an $n \times n$ matrix A with *condition number* $\kappa(A) = poly(n)$ (see Section 3 for a precise definition). Ta-Shma [46], building on the landmark result of Harrow, Hassidim, and Lloyd [22], showed *poly*-conditioned-$\mathsf{MATINV} \in \mathsf{BQL}$. Fefferman and Lin [18] improved upon this result by presenting a *unitary* quantum logspace algorithm and proving a matching $\mathsf{BQ_UL}$-hardness result, thereby exhibiting the first known natural $\mathsf{BQ_UL}$-complete (promise) problem. We further extend this line of research by proving the following theorem, which demonstrates an intriguing relationship between $\mathsf{BQ_UL}$ and DET^*:

Theorem 2. *All of the poly-conditioned versions of the "standard" DET^*-complete problems, given in Definitions 9 and 10 are $\mathsf{BQ_UL}$-complete.*

This shows that several natural linear-algebraic problems are in $\mathsf{BQ_UL}$, and, moreover, are not in BPL (unless $\mathsf{BQ_UL} = \mathsf{BPL}$). In particular, the above theorem shows *poly*-conditioned-$\mathsf{ITMATPROD} \in \mathsf{BQ_UL}$. We also show that this problem is BQL-hard, which implies $\mathsf{BQL} = \mathsf{BQ_UL}$; Theorem 1, which states the more general equivalence for any larger space bound, then follows from a standard padding argument.

We next exhibit several other applications of this theorem. Firstly, in Section 4, we consider *fully logarithmic approximation schemes*, whose study was initiated by Doron and Ta-Shma [16]. Using the preceding theorem, we 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. Secondly, consider κ-conditioned-DET, the problem of approximating $\ln(|\det(A)|)$, to $1/poly(n)$ precision, for an $n \times n$ matrix A with condition number $\kappa(n)$. Boix-Adserà, Eldar, and Mehraban [8] have recently shown that κ-conditioned-DET $\in$ DSPACE$(\log(n)\log(\kappa(n))poly(\log\log n))$. They also raised the following question: is *poly*-conditioned-DET BQL-complete? As an immediate consequence of Theorem 2, we answer their question in the affirmative.

Corollary 3. *poly-conditioned-*DET *is* BQL(= $\mathsf{BQ_UL}$)*-complete.*

To see the significance of the previous corollary, recall the well-known "dequantumization" result given by Watrous [52]: BQL $\subseteq$ DSPACE$(\log^2 n)$. It is natural to ask if a stronger upper bound on BQL can be established. We note that the strongest currently known "derandomization" result of this type, given by Saks and Zhou [42], states BPL $\subseteq$ DSPACE$(\log^{\frac{3}{2}} n)$. Note that the statement BQL $\subseteq$ DSPACE$(\log^{2-\epsilon} n)$ would follow from either a small improvement in the result of Boix-Adserà, Eldar, and Mehraban (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 κ-conditioned-DET remains BQL-hard for *subpolynomial* $\kappa(n)$). Therefore, if BQL $\nsubseteq$ DSPACE$(\log^{2-\epsilon} n)$, $\forall \epsilon > 0$, then both our result and their result are essentially optimal (in terms of the dependence on $\kappa(n)$).

In Section 5, we study well-conditioned versions of the "standard" $\mathsf{C_=L}$-complete problems. We establish a result, very much analogous to Theorem 2, which shows that these problems are complete for the *one*-sided error versions of space-bounded quantum complexity classes. This enables us to prove results, analogous to Theorem 1, concerning the relative power of unitary and general quantum space in the one-sided error cases. We conclude by stating several open problems related to our work in Section 6.

1.3 Techniques

We briefly discuss the techniques used to prove Theorem 2, which states that the *poly*-conditioned versions of the "standard" DET*-complete problems are $\mathsf{BQ_UL}$-complete. As discussed earlier, Fefferman and Lin [18] showed that *poly*-conditioned-MATINV is $\mathsf{BQ_UL}$-complete. In order to establish the $\mathsf{BQ_UL}$-completeness of the other *poly*-conditioned problems, we exhibit a long cycle of reductions through these problems. Note that reductions between the standard versions of these problems (i.e., where there is no assumption of being well-conditioned) are well-known [4, 6, 12, 13, 31, 47–49]. However, these reductions, generally, *do not* preserve the property of being *poly*-conditioned. Therefore, we must exhibit reductions that are rather different from the "standard" reductions.

As a motivating example, consider *poly*-conditioned-DET$^+$ and *poly*-conditioned-SUMITMATPROD. While Berkowitz's algorithm [6] provides a reduction from DET$^+$ to SUMITMATPROD, this reduction does not preserve the property of being *poly*-conditioned. We now provide a brief sketch of a reduction which does preserve this property; see Lemma 19 for a formal proof. Consider a positive definite $n \times n$ matrix H, with $\sigma_1(H) \leq 1$ and $\kappa(H) = poly(n)$. We wish to obtain an additive $1/poly(n)$ approximation of $\ln(\det(H))$. By Jacobi's formula, $\ln(\det(H)) = \mathrm{tr}(\ln(H))$, where $\ln(H)$ denotes the matrix logarithm. We have $\sigma_1(I - H) \leq 1 - 1/poly(n) < 1$, which implies that the series $-\sum_{k=1}^{\infty} \frac{(I-H)^k}{k}$ converges to $\ln(H)$. Therefore, $\ln(\det(H)) = -\sum_{k=1}^{\infty} \frac{\mathrm{tr}((I-H)^k)}{k}$. Moreover, as $\sigma_1(I - H) \leq 1 - 1/poly(n)$, the aforementioned series converges "quickly," which implies that, for some $m = poly(n)$, the quantity $-\sum_{k=1}^{m} \frac{\mathrm{tr}((I-H)^k)}{k}$ is a sufficiently good approximation of $\ln(\det(H))$. Estimating this quantity corresponds to an instance of *poly*-conditioned-SUMITMATPROD.

In Section 3.2, we exhibit a collection of reductions between these various linear-algebraic problems, which use a variety of techniques to preserve the property of being *poly*-conditioned.

Moreover, we note that our paper and that of Boix-Adserà, Eldar, and Mehraban [8] use power series techniques to produce space-efficient algorithms for *poly*-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.

1.4 Related Work

Simultaneously and independently of our work, Girish, Raz, and Zhan [21] proved the following weaker version of our Theorem 2: *contraction*-MATPOW $\in \mathsf{BQ_UL}$, where *contraction*-MATPOW is a special case of our *poly*-conditioned-MATPOW. We note that the techniques used in their proof differed substantially from ours. As a consequence of this result, they then obtain the following weaker version of our Theorem 1: $\mathsf{BQ_UL} = \mathsf{BQ_QL}$, where $\mathsf{BQ_QL} \subseteq$ BQL is a version of quantum logspace that allows a special type of intermediate measurements to be performed, but does not allow using the (classical) result of earlier measurements to control (in a general fashion) later steps of the computation.

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.) consisting of those matrices whose entries are all in $\mathbb{Q}[i]_n$. We define $\mathrm{Mat}(n,c,d) = \{A \in \mathrm{Mat}(n) : c \le \sigma_n(A) \le \sigma_1(A) \le d\}$ and we also analogously define $\widehat{\mathrm{Mat}}(n,c,d)$, $\mathrm{Herm}(n,c,d)$, etc. Let $[n] = \{1,\ldots,n\}$.

We assume that the reader has familiarity with quantum computation and the theory of quantum information; see, for instance, [26, 35, 55] 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 $A = \sum_i p_i |\psi_i\rangle\langle\psi_i| \in \mathrm{Den}(2^s)$.

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 $\mathbf{1}_n \in \mathrm{T}(n)$ 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 $\Phi \otimes \mathbf{1}_r$ 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 briefly recall the definitions of several needed space-bounded quantum complexity classes. Note that, in many of the previous papers that considered space-bounded quantum computation [25, 33, 37, 46, 50–53], 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 TM model.

Definition 4. 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 $\mathsf{DSPACE}(s)$-uniform if there is a deterministic TM 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 5. 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 $\mathsf{DSPACE}(s)$. Let $\mathsf{Q_U SPACE}(s)_{c,k}$ denote the collection of all promise problems $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0)$ such that there is a $\mathsf{DSPACE}(s)$-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, which 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. We require that the following conditions hold:

Completeness: $w \in \mathsf{P}_1 \Rightarrow \Pr[Q_w \text{ accepts } w] \ge c(|w|)$.
Soundness: $w \in \mathsf{P}_0 \Rightarrow \Pr[Q_w \text{ accepts } w] \le k(|w|)$.

Let $\mathsf{BQ_U SPACE}(s) = \mathsf{Q_U SPACE}(s)_{\frac{2}{3},\frac{1}{3}}$ denote (two-sided) bounded-error *unitary* quantum space s and let $\mathsf{BQ_U L} = \mathsf{BQ_U SPACE}(\log n)$ denote unitary quantum *logspace*.

Note that, in the preceding definition, Q_w has $2^{O(s(|w|))}$ gates (this is 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 [41] 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, [33, 50, 52] for discussions of the analogous issue for *quantum* computation. Note that the particular choice of universal gate set does not affect the definition of $\mathsf{BQ_U SPACE}(s)$ due to the space-efficient version [33] of the Solovay-Kitaev theorem. Furthermore, note that the class $\mathsf{BQ_U L}$ would remain the same if defined as $\mathsf{BQ_U L} = \mathsf{Q_U SPACE}(\log n)_{c,k}$ 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 $c(n) - k(n) \ge \frac{1}{q(n)}, \forall n$ [17].

We next consider *general* space-bounded quantum computation. Most basically, we desire 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, 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 equivalency of unitary and general space-bounded quantum computation) to be as strong as possible, we want to use a model of general space-bounded quantum computation that is as powerful as possible. To that end, we consider 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 6. 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 $\mathsf{DSPACE}(s)$-uniform if there is a deterministic TM 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 $K(\Phi_{w,j}) \in \widehat{\mathrm{Mat}}(2^{2h})$.

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, the process of performing any (partial or full) quantum measurement in the computational basis is described by a quantum channel, and 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, e.g., non-computable numbers to be used in defining each gate (see [33, 38, 39, 52] for discussion of this issue).

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 $\mathsf{DSPACE}(s)$. Let $\mathsf{QSPACE}(s)_{c,k}$ denote the collection of all promise problems $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0)$ such that there is a $\mathsf{DSPACE}(s)$-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. We require that $w \in \mathsf{P}_1 \Rightarrow \Pr[\Phi_w \text{ accepts } w] \geq c(|w|)$ and $w \in \mathsf{P}_0 \Rightarrow \Pr[\Phi_w \text{ accepts } w] \leq k(|w|)$. We define *general* quantum space $\mathsf{BQSPACE}(s) = \mathsf{QSPACE}(s)_{\frac{2}{3},\frac{1}{3}}$ and $\mathsf{BQL} = \mathsf{BQSPACE}(\log n)$.

We note that this general quantum circuit model is equivalent to the space-bounded (general) quantum TM model of Watrous [52]. 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 [33, 37, 46, 52]. We refer the reader to [33, 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 next define space-bounded quantum Merlin-Arthur proof systems, essentially following [18].

Definition 8. 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 $\mathsf{DSPACE}(s)$. Let $\mathsf{Q_UMASPACE}(s)_{c,k}$ (resp. $\mathsf{QMASPACE}(s)_{c,k}$) denote the collection of all promise problems $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0)$ such that there is a $\mathsf{DSPACE}(s)$-uniform family of unitary (resp. general) 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 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.* We require that $w \in \mathsf{P}_1 \Rightarrow \exists |\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] \geq c(|w|)$ and $w \in \mathsf{P}_0 \Rightarrow \forall |\psi\rangle \in \Psi_{m_w}, \Pr[V_w \text{ accepts } w, |\psi\rangle] \leq k(|w|)$. We then define $\mathsf{Q_UMASPACE}(s) = \mathsf{Q_UMASPACE}(s)_{\frac{2}{3},\frac{1}{3}}$, $\mathsf{QMASPACE}(s) = \mathsf{QMASPACE}(s)_{\frac{2}{3},\frac{1}{3}}$, $\mathsf{Q_UMAL} = \mathsf{Q_UMASPACE}(\log n)$, and $\mathsf{QMAL} = \mathsf{QMASPACE}(\log n)$.

Lastly we define analogues of these classes for other error types. One-sided bounded error: $\mathsf{RQMASPACE}(s) = \mathsf{QMASPACE}(s)_{\frac{1}{2},0}$, $\mathsf{RQ_USPACE}(s) = \mathsf{Q_USPACE}(s)_{\frac{1}{2},0}$, etc. One-sided unbounded error: $\mathsf{NQSPACE}(s) = \bigcup_{c:\mathbb{N}\to(0,1]} \mathsf{QSPACE}(s)_{c,0}$, $\mathsf{NQMASPACE}(s) = \bigcup_{c:\mathbb{N}\to(0,1]} \mathsf{QMASPACE}(s)_{c,0}$, etc. Note that the classes RQMASPACE and NQMASPACE have perfect soundness. We define QMASPACE with perfect completeness: $\mathsf{QMASPACE}_1(s) = \mathsf{QMASPACE}(s)_{1,\frac{1}{2}}$ and $\mathsf{PreciseQMASPACE}_1(s) = \bigcup_{k:\mathbb{N}\to[0,1)} \mathsf{QMASPACE}(s)_{1,k}$.

3 WELL-CONDITIONED DETERMINANT

We define the following well-conditioned versions of the standard DET^*-complete problems [12]. 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 first define well-conditioned versions of DET and MATINV. The input to each problem consists of a matrix A (among other values) which is promised to be well-conditioned (among other promises): $A \in \widehat{\mathrm{Mat}}(n, 1/\kappa(n), 1)$. Recall that, by definition, this means that $\sigma_1(A) \leq 1$ and $\sigma_n(A) \geq 1/\kappa(n)$, which implies A has condition number at most $\kappa(n)$.

Definition 9. Consider functions $\kappa : \mathbb{N} \to \mathbb{R}_{\geq 1}$ and $\epsilon : \mathbb{N} \to \mathbb{R}_{>0}$.

$\mathsf{DET}_{n,\kappa,\epsilon^{-1}}$

Input: $A \in \widehat{\mathrm{Mat}}(n)$ and $b \in \mathbb{R}_{\leq 0}$

Promise: $A \in \widehat{\mathrm{Mat}}(n, 1/\kappa(n), 1)$, $|\det(A)| \in [0, e^{b-\epsilon(n)}] \cup [e^b, 1]$

Output: 1 if $|\det(A)| \geq e^b$, 0 otherwise

$\mathsf{DET}^+_{n,\kappa,\epsilon^{-1}}$

Input: $A \in \widehat{\mathrm{Mat}}(n)$ and $b \in \mathbb{R}_{\leq 0}$

Promise: $A \in \widehat{\mathrm{Pos}}(n, 1/\kappa(n), 1)$, $\det(A) \in [0, e^{b-\epsilon(n)}] \cup [e^b, 1]$

Output: 1 if $\det(A) \geq e^b$, 0 otherwise

$\mathsf{MATINV}_{n,\kappa,\epsilon^{-1}}$

Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in [n]$, $b \in \mathbb{R}_{\geq 0}$

Promise: $A \in \widehat{\mathrm{Mat}}(n, 1/\kappa(n), 1)$, $|A^{-1}[s,t]| \in [0, b - \epsilon(n)] \cup [b, \infty)$

Output: 1 if $|A^{-1}[s,t]| \geq b$, 0 otherwise

$\mathsf{MATINV}^+_{n,\kappa,\epsilon^{-1}}$

Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in [n]$, $b \in \mathbb{R}_{\geq 0}$

Promise: $A \in \widehat{\mathrm{Pos}}(n, 1/\kappa(n), 1)$, $A^{-1}[s,t] \in [0, b - \epsilon(n)] \cup [b, \infty)$

Output: 1 if $A^{-1}[s,t] \geq b$, 0 otherwise

We next define well-conditioned versions of the various matrix multiplication problems; here, "well-conditioned" has a somewhat different definition. 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 require that all partial products have small singular values $\sigma_1(A_{j_1,j_2}) \leq \kappa(n)$.

Definition 10. Consider functions $m : \mathbb{N} \to \mathbb{N}$, $\kappa : \mathbb{N} \to \mathbb{R}_{\geq 1}$, and $\epsilon : \mathbb{N} \to \mathbb{R}_{>0}$.

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

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

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

$\mathsf{SUMITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$, $E \subseteq [n]^2$, $b \in \mathbb{R}_{\geq 0}$
Promise: $\sigma_1(A_{j_1,j_2}) \leq \kappa(n)\, \forall 1 \leq j_1 \leq j_2 \leq m$,
$$\left|\sum_{(s,t)\in E} A_{1,m}[s,t]\right| \in [0, b-\epsilon(n)] \cup [b, \infty)$$
Output: 1 if $\left|\sum_{(s,t)\in E} A_{1,m}[s,t]\right| \geq b$, 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 3.2 and Section 4, 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, the problems as defined above are somewhat "over parameterized." For example, if $\langle A,s,t,b\rangle \in \mathsf{MATINV}_{n,\kappa(n),\epsilon^{-1}(n)}$, then $\mathsf{MATINV}(\langle A,s,t,b\rangle) = \mathsf{MATINV}(\langle \epsilon(n)A, s, t, \epsilon^{-1}(n)b\rangle)$ and $\langle \epsilon(n)A, s, t, \epsilon^{-1}(n)b\rangle \in \mathsf{MATINV}_{n,\kappa(n)\epsilon^{-1}(n),1}$. These additional parameters are convenient as they allow us to express certain results more cleanly.

Definition 11. For each promise problem $\mathsf{P}_{n,\kappa,\epsilon^{-1}}$ (resp. $\mathsf{P}_{n,m,\kappa,\epsilon^{-1}}$) in Definitions 9 and 10, we define *poly*-conditioned-P to be the promise problem $\mathsf{P}_{n,n^{O(1)},n^{O(1)}}$ (resp. $\mathsf{P}_{n,n^{O(1)},n^{O(1)},n^{O(1)}}$). For example,

poly-conditioned-DET
Input: $A \in \widehat{\mathrm{Mat}}(n)$ and $b \in \mathbb{R}_{\leq 0}$
Promise: $A \in \widehat{\mathrm{Mat}}(n, n^{-O(1)}, 1)$, $|\det(A)| \in [0, e^{b-n^{-O(1)}}] \cup [e^b, 1]$
Output: 1 if $|\det(A)| \geq e^b$, 0 otherwise

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 the following conditions hold: (1) $\mathsf{P}(\langle X\rangle) = \mathsf{P}'(g_n(\langle X\rangle))$, $\forall \langle X\rangle \in \mathsf{P}_{n,f_1,\ldots,f_h}$, (2) $m = p_0(n, f_1(n), \ldots, f_h(n))$, and (3) $f'_j(m) = p_j(n, f_1(n), \ldots, f_h(n))$, $\forall j$. 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 C, we say that P' is C-complete if (1) $\mathsf{P}' \in C$ and (2) $\mathsf{P} \leq^m_{\mathsf{L}} \mathsf{P}'$, $\forall \mathsf{P} \in C$.

Fefferman and Lin [18] showed that *poly*-conditioned-MATINV is $\mathsf{BQ_UL}$-complete. We extend their result by showing that by showing that all of the above *poly*-conditioned problems are $\mathsf{BQ_UL}$-complete. Along the way to proving this result, we also show that $\mathsf{BQ_UL} = \mathsf{BQL}$. To accomplish this, we will prove several lemmas that exhibit reductions between particular problems in Definitions 9 and 10. 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 $\mathsf{P} \leq^m_{\mathsf{L}} \mathsf{P}' \Rightarrow$ *poly*-conditioned-$\mathsf{P} \leq^m_{\mathsf{L}}$ *poly*-conditioned-P'.

3.1 Eliminating Intermediate Measurements

In this section, we show that intermediate measurements may be eliminated without any increase in needed space. We begin by showing that *poly*-conditioned-ITMATPROD is BQL-hard and is in $\mathsf{BQ_UL}$, which implies $\mathsf{BQL} = \mathsf{BQ_UL}$; the general equivalence then follows from a standard padding argument. 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 $r, c \in [m]$, define $F_{m,r,c} \in \widehat{\mathrm{Mat}}(m)$ such that $F_{m,r,c}[r,c] = 1$ and $F_{m,r,c}[r',c'] = 0$, $\forall (r',c') \neq (r,c)$.

Lemma 12. $\mathsf{ITMATPROD} \leq^m_{\mathsf{AC}^0} \mathsf{MATPOW}$.

Proof. Consider $\langle A_1, \ldots, A_m, s, t, b\rangle \in \mathsf{ITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$. Following [12], let $\widehat{A} = \sum_{r=1}^{m} F_{m+1,r,r+1} \otimes A_r \in \widehat{\mathrm{Mat}}(nm+n)$ 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. For $j \in [m]$, we have
$$\widehat{A}^j = \sum_{r=1}^{m+1-j} F_{m+1,r,r+j} \otimes A_{r,r+j-1}.$$
Let $\widehat{s} = s$, $\widehat{t} = nm + t$, $\widehat{b} = b$. Then $\widehat{A}^m[\widehat{s},\widehat{t}] = A_{1,m}[s,t]$, which implies $\mathsf{ITMATPROD}(\langle A_1, \ldots, A_m, s, t, b\rangle) = \mathsf{MATPOW}(\langle \widehat{A}, \widehat{s}, \widehat{t}, \widehat{b}\rangle)$. Moreover, $\forall j \in [m]$, we have
$$\sigma_1(\widehat{A}^j) = \sigma_1\left(\sum_{r=1}^{m+1-j} F_{m+1,r,r+j} \otimes A_{r,r+j-1}\right) = \sigma_1\left(\bigoplus_{r=1}^{m+1-j} A_{r,r+j-1}\right)$$
$$= \max_r \sigma_1(A_{r,r+j-1}) \leq \kappa(n).$$
Therefore, $\langle \widehat{A}, \widehat{s}, \widehat{t}, \widehat{b}\rangle \in \mathsf{MATPOW}_{nm+n,m,\kappa(n),\epsilon^{-1}(n)}$. □

Lemma 13. $\mathsf{MATPOW} \leq^m_{\mathsf{AC}^0} \mathsf{MATINV}$.

Proof. Suppose $\langle A, s, t, b\rangle \in \mathsf{MATPOW}_{n,m,\kappa,\epsilon^{-1}}$. Following [12], let $G_j = \sum_{r=1}^{m+1-j} F_{m+1,r,r+j} \in \widehat{\mathrm{Mat}}(m+1), \forall j \in [m]$. 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

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

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

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

$$\le 1 + \sum_{j=1}^{m} \kappa(n) \le 1 + m\kappa(n).$$

This implies $\sigma_{nm+n}(Z) = \sigma_1(Z^{-1})^{-1} \ge (1 + m\kappa(n))^{-1}$. 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 14. $\mathsf{MATINV} \le^m_{\mathsf{NC}^1} \mathsf{MATINV}^+$.

Proof. Consider $\langle A, s, t, b\rangle \in \mathsf{MATINV}_{\kappa,\epsilon^{-1}}$. We define $\widehat{H} = \frac{1}{3}\begin{pmatrix} A^\dagger A & -A^\dagger \\ -A & 2I \end{pmatrix} \in \widehat{\mathrm{Pos}}(2n)$. Note that $\widehat{H}^{-1} = 3\begin{pmatrix} 2(A^\dagger A)^{-1} & A^{-1} \\ (A^\dagger)^{-1} & I \end{pmatrix}$. Moreover, $\sigma_1(\widehat{H}) \le 1$ and $\sigma_{2n}(\widehat{H}) \ge \frac{1}{9}(\sigma_n(A))^2 \ge (3\kappa(n))^{-2}$. 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, $\mathsf{MATINV}(\langle A,s,t,b\rangle) = \mathsf{MATINV}(\langle H, \widehat{s}, \widehat{t}, \widehat{b}\rangle)$ and $\langle \widehat{H}, \widehat{s}, \widehat{t}, \widehat{b}\rangle \in \mathsf{MATINV}^+_{2n,(3\kappa(n))^2,(3\epsilon(n))^{-1}}$. □

Lemma 15. *poly-conditioned*-$\mathsf{ITMATPROD} \in \mathsf{BQ_UL}$.

Proof. By Lemmas 12 to 14, $\mathsf{ITMATPROD} \le^m_{\mathsf{NC}^1} \mathsf{MATINV}^+$. Recall $\mathsf{P} \le^m_{\mathsf{NC}^1} \mathsf{P}' \Rightarrow$ *poly*-conditioned-$\mathsf{P} \le^m_{\mathsf{NC}^1}$ *poly*-conditioned-P'. By [18, Theorem 13], *poly*-conditioned-$\mathsf{MATINV}^+ \in \mathsf{BQ_UL}$. □

Lemma 16. *poly-conditioned*-$\mathsf{ITMATPROD}$ *is* BQL*-hard.*

Proof. Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{BQL}$. By definition, there is a uniform family of general quantum circuits $\{\Phi_w = (\Phi_{w,1}, \dots, \Phi_{w,t_w}) : w \in \mathsf{P}\}$, 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 $\Pr[\Phi_w \text{ accepts } w] \ge \frac{2}{3}$, and if $w \in \mathsf{P}_0$, then $\Pr[\Phi_w \text{ accepts } w] \le \frac{1}{3}$. 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 each $j \in [t_w]$, we define $A(w)_j = K(\Phi_{w,t_w-j+1})$; note that, by Definition 6, $A(w)_j \in \widehat{\mathrm{Mat}}(d_w)$ 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 $1 \le j_1 \le j_2 \le t_w$, 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 [40, Theorem 1] implies the following bound on the largest singular value of any partial product of the $A(w)_j$

$$\sigma_1(A(w)_{j_1,j_2}) = \sigma_1\left(\prod_{j=j_1}^{j_2} A(w)_j\right)$$

$$= \sigma_1(K(\Phi_{w,t_w-j_2+1} \circ \cdots \circ \Phi_{w,t_w-j_1+1})) \le \sqrt{d_w} = n^{O(1)}.$$

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 7,

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

Thus, $\mathsf{ITMATPROD}(\langle A(w)_1, \dots, A(w)_{t_w}, x_w, y_w, \frac{2}{3}\rangle) = \mathsf{P}(w)$ and $\langle A(w)_1, \dots, A(w)_{t_w}, x_w, y_w, \frac{2}{3}\rangle \in$ *poly*-conditioned-$\mathsf{ITMATPROD}$. □

For the sake of completeness, in Appendix A, we also prove a version of the preceding lemma for the (equivalent) version of quantum logspace that is defined using quantum Turing machines.

Lemma 17. $\mathsf{QMAL} \subseteq \mathsf{Q_UMAL}^{\mathsf{BQ_UL}}$.

Proof. This follows by an argument similar to that of the proof of Lemma 16. Suppose $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_0) \in \mathsf{QMAL}$. There is a uniform family of general quantum circuits $\{\Phi_w = (\Phi_{w,1}, \dots, \Phi_{w,t_w}) : w \in \mathsf{P}\}$, where Φ_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 Φ_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*. If $w \in \mathsf{P}_1$, then $\exists |\psi\rangle \in \Psi_{m_w}, \Pr[\Phi_w \text{ accepts } w, |\psi\rangle] \ge \frac{2}{3}$, and if $w \in \mathsf{P}_0$, then $\forall |\psi\rangle \in \Psi_{m_w}, \Pr[\Phi_w \text{ accepts } w, |\psi\rangle] \le \frac{1}{3}$.

Note $\Pr[\Phi_w \text{ accepts } w, |\psi\rangle] = \langle \Pi_1, \Phi_w(|\psi\rangle\langle\psi| \otimes |0^{h_w}\rangle\langle 0^{h_w}|)\rangle = \langle \Phi_w^\dagger(\Pi_1), |\psi\rangle\langle\psi| \otimes |0^{h_w}\rangle\langle 0^{h_w}|\rangle$. Let $M = (I \otimes \langle 0^{h_w}|)\Phi_w^\dagger(\Pi_1)(I \otimes |0^{h_w}\rangle) \in \widehat{\mathrm{Pos}}(2^{m_w})$. Then $w \in \mathsf{P}_1 \Leftrightarrow \lambda_1(M) \ge \frac{2}{3}$ and $w \in \mathsf{P}_0 \Leftrightarrow \lambda_1(M) \le \frac{1}{3}$. Given access to M, the problem, of determining if $\lambda_1(M) \ge \frac{2}{3}$ or $\lambda_1(M) \le \frac{1}{3}$, is obviously in $\mathsf{Q_UMAL}$. By the same argument as in the proof of Lemma 16, estimating an entry of M (to $1/poly(n)$ precision) corresponds to an instance of *poly*-conditioned-$\mathsf{ITMATPROD}$, which, by Lemma 15, is in $\mathsf{BQ_UL}$. Therefore, $\mathsf{P} \in \mathsf{Q_UMAL}^{\mathsf{BQ_UL}}$. □

Lemma 18. $\mathsf{BQ_UL} = \mathsf{BQL} = \mathsf{Q_UMAL} = \mathsf{QMAL}$.

Proof. Clearly, $\mathsf{BQ_UL} \subseteq \mathsf{BQL}$. To see that $\mathsf{BQL} \subseteq \mathsf{BQ_UL}$, note that, by Lemma 16, *poly*-conditioned-$\mathsf{ITMATPROD}$ is BQL-hard, and, by Lemma 15, *poly*-conditioned-$\mathsf{ITMATPROD} \in \mathsf{BQ_UL}$. Therefore, $\mathsf{BQL} = \mathsf{BQ_UL}$. By [18, Theorem 18] $\mathsf{Q_UMAL} = \mathsf{BQ_UL}$ (in fact, their argument shows $\mathsf{Q_UMAL}^{\mathcal{O}} = \mathsf{BQ_UL}^{\mathcal{O}}$, for any oracle $\mathcal{O}$). Clearly, $\mathsf{BQL} \subseteq \mathsf{QMAL}$, and so it suffices to show $\mathsf{QMAL} \subseteq \mathsf{BQL}$. By Lemma 17, and the (straightforward) fact that BQL is self-low, we have

$$\mathsf{QMAL} \subseteq \mathsf{Q_UMAL}^{\mathsf{BQ_UL}} = \mathsf{BQ_UL}^{\mathsf{BQ_UL}} \subseteq \mathsf{BQL}^{\mathsf{BQL}} = \mathsf{BQL}. \quad \square$$

We now sketch the proof of Theorem 1 from Section 1.1.

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

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

Proof (Sketch). Clearly, $\mathsf{BQ_USPACE}(s) \subseteq \mathsf{BQSPACE}(s)$. The reverse containment $\mathsf{BQSPACE}(s) \subseteq \mathsf{BQ_USPACE}(s)$ follows from Lemma 18 and a standard padding argument. Analogous statements hold for QMASPACE. See the full paper [19] for details. □

3.2 BQ$_{\mathrm{U}}$L Completeness

In this section, we show that the *poly*-conditioned versions of all of the standard DET^*-complete problems (Definitions 9 and 10) are $\mathsf{BQ_UL}$-complete. By [18, Theorem 13], *poly*-conditioned-MATINV is $\mathsf{BQ_UL}$-complete[2]; therefore, it suffices to exhibit a chain of reductions through the various *poly*-conditioned-P. As noted earlier, $\mathsf{P} \leq^m_{\mathsf{L}} \mathsf{P}' \Rightarrow$ *poly*-conditioned-P $\leq^m_{\mathsf{L}}$ *poly*-conditioned-P$'$

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

Proof. Consider $\langle H, b\rangle \in \mathsf{DET}^+_{n,\kappa,\epsilon^{-1}}$. By the promise, $H \in \widehat{\mathrm{Pos}}(n)$, $\lambda_1(H) = \sigma_1(H) \leq 1$, and $\lambda_n(H) = \sigma_n(H) \geq \kappa(n)^{-1}$, which implies $\sigma_1(I-H) = \lambda_1(I-H) = 1 - \lambda_n(H) \leq 1 - \kappa(n)^{-1} < 1$. 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 $m \in \mathbb{N}_{\geq 1}$, 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 $k \in [m]$, 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 $E_m = \{(d,d) : d \in [n\widehat{l} + nm]\}$. We then have

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

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

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

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 $\mathrm{tr}(R_m) \geq 0$. We then have,

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

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 $\mathsf{DET}(\langle H, b\rangle) = \mathsf{SUMITMATPROD}(\langle \widehat{A}_1, \ldots, \widehat{A}_{\widehat{m}}, \widehat{E}, \widehat{b}\rangle)$. □

Lemma 20. ITMATPROD $\leq^m_{\mathsf{AC}^0}$ $\mathsf{ITMATPROD}^{\geq 0}$.

Proof. Consider $\langle A_1, \ldots, A_m, s, t, b\rangle \in \mathsf{ITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$. For $j \in [2m+1]$, we define

$$\widehat{A}_j = \begin{cases} A_j, & j \leq m \\ |t\rangle\langle t|, & j = m+1 \\ A^{\dagger}_{2m+2-j}, & j \geq m+2 \end{cases}$$

We then have $\widehat{A}_{1,2m+1}[s,s] = A_{1,m}[s,t]\overline{A_{1,m}[s,t]} = |A_{1,m}[s,t]|^2$. Let $\widehat{s} = \widehat{t} = s$ and $\widehat{b} = b^2$. Then $\mathsf{ITMATPROD}(\langle A_1, \ldots, A_m, s, t, b\rangle) = \mathsf{ITMATPROD}(\langle \widehat{A}_1, \ldots, \widehat{A}_{2m+1}, \widehat{s}, \widehat{t}, \widehat{b}\rangle)$. 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

$$\sigma_1(\widehat{A}_{j_1,j_2}) \leq \sigma_1(A_{j_1,m})\sigma_1(|t\rangle\langle t|)\sigma_1(A^{\dagger}_{j_2,m}) \leq \kappa(n)^2.$$

If $j_2 < m+1$, then $\sigma_1(\widehat{A}_{j_1,j_2}) = \sigma_1(A_{j_1,j_2}) \leq \kappa(n)$. If $j_1 > m+1$, then $\sigma_1(\widehat{A}_{j_1,j_2}) = \sigma_1(A_{j_1-m-1,j_2-m-1}) \leq \kappa(n)$. We then conclude, $\langle \widehat{A}_1, \ldots, \widehat{A}_{2m+1}, \widehat{s}, \widehat{t}, \widehat{b}\rangle \in \mathsf{ITMATPROD}^{\geq 0}_{n,2m+1,\kappa^2,\epsilon^{-2}}$. □

Lemma 21. $\mathsf{ITMATPROD}^{\geq 0} \leq^m_{\mathsf{AC}^0}$ DET.

Proof. Consider $\langle A_1, \ldots, A_m, s, t, b\rangle \in \mathsf{ITMATPROD}^{\geq 0}_{n,m,\kappa,\epsilon^{-1}}$. 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

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

Let $C = B + |nm+t\rangle\langle s|$. By the matrix determinant lemma, and the fact that $\det(B) = 1$,

$$\det(C) = (1 + \langle s|B^{-1}|nm+t\rangle)\det(B) = 1 + A_{1,m}[s,t].$$

Next, observe that

$$\sigma_1(C) \leq \sigma_1(|nm+t\rangle\langle s|) + \sigma_1(I) + \sigma_1(Y) \leq 2 + \max_j \sigma_1(A_j) \leq 2 + \kappa(n).$$

[2]Note that Ref. [18] showed *poly*-conditioned-$\mathsf{MATINV}^+ \in \mathsf{BQ_UL}$ and that *poly*-conditioned-MATINV is $\mathsf{BQ_UL}$-hard, but the equivalence between these two problems is "obvious" (and shown explicitly in Lemma 14)

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

$$\sigma_1(B^{-1}) \le \sum_{j=0}^{m} \sigma_1(Y^j) \le 1 + \sum_{j=1}^{m}\left(\max_{k\in[m-j+1]} \sigma_1(A_{k,k+j-1})\right)$$

$$\le 1 + \sum_{j=1}^{m} \kappa(n) = 1 + m\kappa(n).$$

By the Sherman-Morrison formula, $C^{-1} = B^{-1}(I-(1+\langle s|B^{-1}|nm+t\rangle)^{-1}|nm+t\rangle\langle s|B^{-1})$. Recall that, by the promise, $\langle s|B^{-1}|nm+t\rangle = A_{1,m}[s,t] \in \mathbb{R}_{\ge 0}$. Therefore,

$$\sigma_1(C^{-1}) \le \sigma_1(B^{-1})(\sigma_1(I) + (1+\langle s|B^{-1}|nm+t\rangle)^{-1}\sigma_1(B^{-1}))$$

$$\le (1+m\kappa(n))(2+m\kappa(n)).$$

This implies $\sigma_{nm+n}(C) = \sigma_1(C^{-1})^{-1} \ge ((1+m\kappa(n))(2+m\kappa(n)))^{-1}$. Let $\widehat{l} = \lfloor 1+\ln(\lfloor 2+\kappa(n)\rfloor)\rfloor$ and let $\widehat{C} = e^{-\widehat{l}}C \in \widehat{\mathrm{Mat}}(nm+n)$. Then, for $j \in [nm+n]$, $\sigma_j(\widehat{C}) = e^{-\widehat{l}}\sigma_j(C)$; in particular, $\sigma_1(\widehat{C}) \le 1$ and $\sigma_{nm+n}(\widehat{C}) \ge (2+m\kappa(n))^{-3}$. Moreover,

$$|\det(\widehat{C})| = |e^{-\widehat{l}(nm+n)}\det(C)|$$

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

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

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

$$\ge \ln\left(1+\frac{\epsilon(n)}{1+\kappa(n)}\right) \ge \frac{\epsilon(n)}{2(1+\kappa(n))}.$$

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

Lemma 22. $\mathsf{DET} \le^m_{\mathsf{NC}^1} \mathsf{DET}^+$.

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 $\sigma_j(\widehat{H}) = \sigma_j^2(A)$, $\forall j$. Thus, $\langle \widehat{H},\widehat{b}\rangle \in \mathsf{DET}_{n,\kappa^2,2\epsilon^{-1}}$ and $\mathsf{DET}(\langle A,b\rangle) = \mathsf{DET}(\langle \widehat{H},\widehat{b}\rangle)$. ◻

Lemma 23. $\mathsf{MATINV}^+ \le^m_{\mathsf{AC}^0} \mathsf{SUMITMATPROD}$.

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 $j \in [\widehat{m}]$, let $\widehat{A}_j = I_{jn} \oplus (I_{\widehat{m}-j+1} \otimes (I-H)) \in \widehat{\mathrm{Mat}}(n\widehat{m}+n)$. For $1 \le j_1 \le j_2 \le \widehat{m}$, 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 $\widehat{E} = \{(s+jn, t+jn) : j \in [\widehat{m}]\}$. 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 $\widehat{b} = b - \frac{1}{4}\epsilon(n)$. Therefore, we then have $\mathsf{MATINV}(\langle H,s,t,b\rangle) = \mathsf{SUMITMATPROD}(\langle \widehat{A}_1,\ldots,\widehat{A}_{\widehat{m}},\widehat{E},\widehat{b}\rangle)$ and, furthermore, we have $\langle \widehat{A}_1,\ldots,\widehat{A}_{\widehat{m}},\widehat{E},\widehat{b}\rangle \in \mathsf{SUMITMATPROD}_{n\widehat{m}+n,\widehat{m},1,2\epsilon^{-1}(n)}$. ◻

Lemma 24. $\mathsf{SUMITMATPROD} \le^m_{\mathsf{AC}^0} \mathsf{ITMATPROD}$.

Proof. Fix $\langle A_1,\ldots,A_m,E,b\rangle \in \mathsf{SUMITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$. Let $T_{c,d} \in \widehat{\mathrm{Mat}}(n)$ denote the permutation matrix corresponding to interchanging elements $c,d \in [n]$ and leaving all other elements fixed. For $j \in [m]$, 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 $\widehat{A}_{0,m+1}[1,1] = \sum_{(s,t)\in E} A_{1,m}[s,t]$. Let $\widehat{s} = \widehat{t} = 1$ and $\widehat{b} = b$. We then conclude that $\mathsf{SUMITMATPROD}(\langle A_1,\ldots,A_m,E,b\rangle) = \mathsf{ITMATPROD}(\langle \widehat{A}_0,\ldots,\widehat{A}_{m+1},\widehat{s},\widehat{t},\widehat{b}\rangle)$.

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 that, for $0 \le j_1 \le j_2 \le m+1$,

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

Thus, $\langle \widehat{A}_0,\ldots,\widehat{A}_{m+1},\widehat{s},\widehat{t},\widehat{b}\rangle \in \mathsf{ITMATPROD}_{n^3,m+2,2n^2\kappa,\epsilon^{-1}}$. ◻

We now prove Theorem 2 from Section 1.2.

Theorem 2. *All of the poly-conditioned versions of the "standard" DET^*-complete problems, given in Definitions 9 and 10 are $\mathsf{BQ_UL}$-complete.*

Proof. Recall that *poly*-conditioned-MATINV is already known to be $\mathsf{BQ_UL}$-complete [18, Theorem 13]. By Lemmas 12 to 14 and 19 to 24, we have

$$\mathsf{MATINV}^+ \le^m_{\mathsf{AC}^0} \mathsf{SUMITMATPROD} \le^m_{\mathsf{AC}^0} \mathsf{ITMATPROD}$$

$$\le^m_{\mathsf{AC}^0} \mathsf{MATPOW} \le^m_{\mathsf{AC}^0} \mathsf{MATINV} \le^m_{\mathsf{NC}^1} \mathsf{MATINV}^+$$

and

$$\mathsf{DET}^+ \le^m_{\mathsf{AC}^0} \mathsf{SUMITMATPROD} \le^m_{\mathsf{AC}^0} \mathsf{ITMATPROD}$$

$$\le^m_{\mathsf{AC}^0} \mathsf{ITMATPROD}^{\ge 0} \le^m_{\mathsf{AC}^0} \mathsf{DET} \le^m_{\mathsf{NC}^1} \mathsf{DET}^+.$$

Recall that $\le^m_{\mathsf{AC}^0}$ or $\le^m_{\mathsf{NC}^1}$ reducibility implies $\le^m_{\mathsf{L}}$ reducibility, and that $\mathsf{P} \le^m_{\mathsf{L}} \mathsf{P}' \Rightarrow$ *poly*-conditioned-$\mathsf{P} \le^m_{\mathsf{L}}$ *poly*-conditioned-P'. Therefore, each such *poly*-conditioned-P is $\mathsf{BQ_UL}$-complete. ◻

4 FULLY LOGARITHMIC APPROXIMATION SCHEMES

We next 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 25. 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 9 to 11, let

$$\mathcal{D}(poly\text{-}matinv) = \bigcup_n \{\langle A, s, t\rangle : A \in \widehat{\mathrm{Mat}}(n, n^{-O(1)}, 1), s, t \in [n]\}.$$

In other words, $\mathcal{D}(poly\text{-}matinv)$ consists of encodings of instances of a variant of *poly*-conditioned-MATINV where we only have a promise involving the singular values (i.e., there is no restriction on $A^{-1}[s,t]$ involving b). We then consider the *poly*-conditioned matrix inversion function $|poly\text{-}matinv(\cdot)| : \mathcal{D}(poly\text{-}matinv) \to \mathbb{R}_{\geq 0}$, which is given by $|poly\text{-}matinv(\langle A, s, t\rangle)| = |A^{-1}[s,t]|$.

Similarly, $\mathcal{D}(poly\text{-}itmatprod)$ consists of all $\langle A_1, \ldots, A_m, s, t\rangle$, where $m = poly(n)$, $A_1, \ldots, A_m \in \widehat{\mathrm{Mat}}(n)$, $\sigma_1(A_{j_1,j_2}) \leq poly(n)$ for $1 \leq j_1 \leq j_2 \leq m$, and $s, t \in [n]$. We then define the function $|poly\text{-}itmatprod(\cdot)| : \mathcal{D}(poly\text{-}itmatprod) \to \mathbb{R}_{\geq 0}$ such that $|poly\text{-}itmatprod(\langle A_1, \ldots, A_m, s, t\rangle)| = |A_{1,m}[s,t]|$. Lastly, we define $\mathcal{D}(poly\text{-}det) = \bigcup_n \{\langle A\rangle : A \in \widehat{\mathrm{Mat}}(n, n^{-O(1)}, 1)\}$. Note that the promise problem DET, given in Definition 9, corresponds to approximating the function $\ln(|poly\text{-}det(\cdot)|) : \mathcal{D}(poly\text{-}det) \to \mathbb{R}_{\leq 0}$, defined in the obvious way.

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, [23]) that multiplicative approximations are defined using $e^{\pm\epsilon}$, rather than the more standard (and essentially equivalent) $(1 \pm \epsilon)$. Note that $\ln(|poly\text{-}det(\cdot)|)$ has an (additive) $\mathsf{FLQ_UAS}$ (resp. FLQAS, FLRAS, FLAS) if and only if $|poly\text{-}det(\cdot)|$ has a multiplicative $\mathsf{FLQ_UAS}$ (resp. FLQAS, FLRAS, FLAS); this follows from the fact that $|\ln(|\det(A)|) - y| \geq \epsilon$ if and only if $e^y \notin [e^{-\epsilon}|\det(A)|, e^{\epsilon}|\det(A)|]$.

Lemma 26. *Each of* $|poly\text{-}matinv(\cdot)|$, $|poly\text{-}itmatprod(\cdot)|$, *and* $\ln(|poly\text{-}det(\cdot)|)$ *have a* $\mathsf{FLQ_UAS}$. *Moreover,* $|poly\text{-}det(\cdot)|$ *has a multiplicative* $\mathsf{FLQ_UAS}$.

Proof. By [18, Theorem 14] (and the discussion following it), $|poly\text{-}matinv(\cdot)|$ has a $\mathsf{FLQ_UAS}$; this improved upon the earlier result of Ta-Shma [46], which showed that this function has a FLQAS [16]. By Lemmas 12 and 13, $|poly\text{-}itmatprod(\cdot)|$ has a $\mathsf{FLQ_UAS}$. Finally, by Lemmas 19, 22 and 24, $\ln(|poly\text{-}det(\cdot)|)$ has a $\mathsf{FLQ_UAS}$; this implies $|poly\text{-}det(\cdot)|$ has a multiplicative $\mathsf{FLQ_UAS}$. □

Doron and Ta-Shma [16, Theorem 6] showed that, if BQL = 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 2) and our result that $\mathsf{BQ_UL} = \mathsf{BQL}$ (Lemma 18), the following proposition is immediate; we note that a partial (weaker) version of this proposition was implicit in [15].

Proposition 27. *The following statements are equivalent.*

(i) BQL = 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) $|poly\text{-}matinv(\cdot)|$ *has a* FLRAS.
(v) $|poly\text{-}itmatprod(\cdot)|$ *has a* FLRAS.
(vi) $\ln(|poly\text{-}det(\cdot)|)$ *has a* FLRAS.
(vii) $|poly\text{-}det(\cdot)|$ *has a multiplicative* FLRAS.

Remark. In particular, the preceding proposition suggests that $|poly\text{-}det(\cdot)|$ *does not* have a multiplicative FLRAS (as this would imply the *seemingly* unlikely statement BQL = BPL). It is natural to compare this statement with the result of Jerrum, Sinclair, and Vigoda [23] which shows the existence of a (multiplicative) FPRAS (fully *polynomial* randomized approximation scheme) for the *permanent* of a nonnegative integer matrix.

5 WELL-CONDITIONED SINGULAR

The class $\mathsf{C_=L}$ has a collection of natural complete problems, given by the "verification" versions of the standard DET^*-complete problems [43]. In this section, we study the well-conditioned versions of these problems.

Definition 28. Consider functions $m : \mathbb{N} \to \mathbb{N}$, $\kappa : \mathbb{N} \to \mathbb{R}_{\geq 1}$, and $\epsilon : \mathbb{N} \to \mathbb{R}_{>0}$.

$\mathsf{SINGULAR}_{n,\epsilon^{-1}}$
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

$\mathsf{vMATINV}_{n,\kappa,\epsilon^{-1}}$
Input: $A \in \widehat{\mathrm{Mat}}(n)$, $s, t \in [n]$, $b \in \mathbb{Q}[i]_n$
Promise: $A \in \widehat{\mathrm{Mat}}(n, 1/\kappa(n), 1)$, $|A^{-1}[s,t] - b| \in \{0\} \cup [\epsilon(n), \infty)$

Output: 1 if $A^{-1}[s,t]=b$, 0 otherwise
$\mathsf{vMATPOW}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A\in\widehat{\mathrm{Mat}}(n)$, $s,t\in[n]$, $b\in\mathbb{Q}[i]_n$
Promise: $\sigma_1(A^j)\le\kappa(n)\,\forall j\in[m]$, $|A^m[s,t]-b|\in\{0\}\cup[\epsilon(n),\infty)$
Output: 1 if $A^m[s,t]=b$, 0 otherwise
$\mathsf{vITMATPROD}_{n,m,\kappa,\epsilon^{-1}}$
Input: $A_1,\dots,A_m\in\widehat{\mathrm{Mat}}(n)$, $s,t\in[n]$, $b\in\mathbb{Q}[i]_n$
Promise: $\sigma_1(A_{j_1,j_2})\le\kappa(n)\,\forall 1\le j_1\le j_2\le m$,
$\left|A_{1,m}[s,t]-b\right|\in\{0\}\cup[\epsilon(n),\infty)$
Output: 1 if $A_{1,m}[s,t]=b$, 0 otherwise

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 29. $\mathsf{vMATINV}\le^m_{\mathsf{AC}^0}\mathsf{SINGULAR}$.

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

$$\det(\widehat{C})=(1-u\widehat{B}^{-1}v)\det(\widehat{B})$$

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

If $A^{-1}[s,t]=b$, then $\det(\widehat{C})=0$, which implies $\sigma_{n+1}(\widehat{C})=0$. If, instead, $|A^{-1}[s,t]-b|\ge\epsilon(n)$, then $|\det(\widehat{C})|\ge\frac{\epsilon(n)}{2\lceil\kappa(n)\rceil}|\det(\widehat{B})|$. By the Weyl inequalities, $\sigma_1(\widehat{C})\le\sigma_1(\widehat{B})+\sigma_1(-vu^\dagger)=\sigma_1(\widehat{B})+1$ and, for $j\in[2,\dots,n+1]$, we have $\sigma_j(\widehat{C})\le\sigma_{j-1}(\widehat{B})+\sigma_2(-vu^\dagger)=\sigma_{j-1}(\widehat{B})$. Moreover, $\sigma_1(\widehat{B})=2\lceil\kappa(n)\rceil\sigma_1(A)\le2\lceil\kappa(n)\rceil$, $\sigma_n(\widehat{B})=2\lceil\kappa(n)\rceil\sigma_n(A)\ge2$, and $\sigma_{n+1}(\widehat{B})=\left|1-\frac{b}{2\lceil\kappa(n)\rceil}\right|^{-1}\ge\frac{2}{\sqrt{5}}$. Therefore,

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

$$=\frac{\epsilon(n)\sigma_n(\widehat{B})\sigma_{n+1}(\widehat{B})}{2\lceil\kappa(n)\rceil(\sigma_1(\widehat{B})+1)}\ge\frac{2\epsilon(n)}{\sqrt{5}\lceil\kappa(n)\rceil(2\lceil\kappa(n)\rceil+1)}.$$

Let $\widehat{d}=\frac{1}{2\lceil\kappa(n)\rceil+1}$ and $\widehat{H}=\widehat{d}\begin{pmatrix}0_{n+1}&\widehat{C}\\ \widehat{C}^\dagger&0_{n+1}\end{pmatrix}\in\widehat{\mathrm{Herm}}(2n+2)$. Notice that $\widehat{H}$ has eigenvalues $\{\pm\widehat{d}\sigma_1(\widehat{C}),\dots,\pm\widehat{d}\sigma_{n+1}(\widehat{C})\}$. This implies $\sigma_1(\widehat{H})=\widehat{d}\sigma_1(\widehat{C})\le1$ 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 $\mathsf{vMATINV}(\langle A,s,t,b\rangle)=\mathsf{SINGULAR}(\langle\widehat{H}\rangle)$. □

The following pair of lemmas have proofs precisely analogous to that of Lemma 12 and Lemma 13, respectively.

Lemma 30. $\mathsf{vITMATPROD}\le^m_{\mathsf{AC}^0}\mathsf{vMATPOW}$.

Lemma 31. $\mathsf{vMATPOW}\le^m_{\mathsf{AC}^0}\mathsf{vMATINV}$.

5.1 RQSPACE vs. RQ$_\mathrm{U}$SPACE vs. RQ$_\mathrm{U}$MASPACE

In this section, we will prove the following relationships between the various one-sided bounded-error space-bounded quantum complexity classes.

Theorem 32. *For any space-constructible function $s:\mathbb{N}\to\mathbb{N}$, where $s(n)=\Omega(\log n)$, we have*

$$\mathsf{RQ_UMASPACE}(s)=\mathsf{RQ_USPACE}(s)\subseteq\mathsf{RQSPACE}(s)$$
$$\subseteq\mathsf{coQ_UMASPACE_1}(s).$$

This theorem, which is very much the one-sided error analogue of Theorem 1, has a proof which follows the same general structure of Theorem 1. For those promise problems P given in Definition 28, we define *poly*-conditioned-P as in Definition 11.

Lemma 33. *poly-conditioned-*$\mathsf{vITMATPROD}$ *is* coRQL*-hard.*

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

Lemma 34. $\mathsf{RQ_UMAL}\subseteq\mathsf{RQ_UL}$

Proof (sketch). Apply the well-known technique of replacing Merlin's proof with the totally mixed state [32], which preserves perfect soundness [27]; then use space-efficient probability amplification for one-sided bounded-error (unitary) quantum logspace [51]. See the full paper [19] for details. □

Lemma 35. *poly-conditioned-*$\mathsf{SINGULAR}$ *is* $\mathsf{Q_UMAL_1}$*-complete.*

Proof. We first establish $\mathsf{Q_UMAL_1}$-hardness. Consider some $\mathsf{P}=(\mathsf{P}_1,\mathsf{P}_0)\in\mathsf{Q_UMAL_1}$. By definition, there is a uniform family of (unitary) quantum circuits $\{V_w=(V_{w,1},\dots,V_{w,t_w}):w\in\mathsf{P}\}$, where V_w acts on $m_w+h_w=O(\log|w|)$ qubits and has $t_w=poly(|w|)$ gates, such that $w\in\mathsf{P}_1\Rightarrow\exists|\psi\rangle\in\Psi_{m_w},\Pr[V_w\text{ accepts }w,|\psi\rangle]\ge c=1$, and $w\in\mathsf{P}_0\Rightarrow\forall|\psi\rangle\in\Psi_{m_w},\Pr[V_w\text{ accepts }w,|\psi\rangle]\le k=\frac{1}{2}$, where $\Pr[V_w\text{ accepts }w,|\psi\rangle]=\|\Pi_1V_w(|\psi\rangle\otimes|0^{h_w}\rangle)\|^2$.

We make use of the Kitaev clock Hamiltonian construction [26, 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 $\Pi_b=I_{2^{b-1}}\otimes|1\rangle\langle1|\otimes I_{2^{m_w+h_w-b}}\in\widehat{\mathrm{Proj}}(2^{m_w+h_w})$ 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:

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

$$H_w^{in}=\sum_{b=m_w+1}^{m_w+h_w}(\Pi_b\otimes|0\rangle\langle0|),\quad H_w^{out}=\Pi_1\otimes|t_w\rangle\langle t_w|,$$

$$H_w=H_w^{in}+H_w^{prop}+H_w^{out}.$$

By [26, Section 14.4], $\exists r_0,r_1\in\mathbb{R}_{>0}$, such that $\forall w\in\mathsf{P}$ the following conditions hold: (1) $\sigma_1(H_w)\le r_0$, (2) $w\in\mathsf{P}_1\Rightarrow\sigma_{d_w}(H_w)\le\frac{1-c}{t_w+1}=0$, and (3) $w\in\mathsf{P}_0\Rightarrow\sigma_{d_w}(H_w)\ge r_1\frac{1-\sqrt{k}}{t_w+1}=1/poly(d_w)$.

By the above, $\langle H_w \rangle \in poly$-conditioned-SINGULAR and $\mathsf{P}(w) = \mathsf{SINGULAR}(\langle H_w \rangle)$, which implies *poly*-conditioned-SINGULAR is $\mathsf{Q_UMAL_1}$-hard.

The fact that *poly*-conditioned-SINGULAR $\in \mathsf{Q_UMAL_1}$ 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 36. $\mathsf{RQ_UMAL} = \mathsf{RQ_UL} \subseteq \mathsf{RQL} \subseteq \mathsf{coQ_UMAL_1}$.

Proof. Clearly, $\mathsf{RQ_UL} \subseteq \mathsf{RQ_UMAL}$. By Lemma 34, $\mathsf{RQ_UMAL} \subseteq \mathsf{RQ_UL}$, which implies $\mathsf{RQ_UMAL} = \mathsf{RQ_UL}$. Trivially, $\mathsf{RQ_UL} \subseteq \mathsf{RQL}$ By Lemma 33, *poly*-conditioned-vITMATPROD is coRQL-hard, and so, by Lemmas 29 to 31, *poly*-conditioned-SINGULAR is coRQL-hard. By Lemma 35, *poly*-conditioned-SINGULAR $\in \mathsf{Q_UMAL_1}$, which implies $\mathsf{coRQL} \subseteq \mathsf{Q_UMAL_1}$; thus, $\mathsf{RQL} \subseteq \mathsf{coQ_UMAL_1}$. □

The main theorem stated at the beginning of this section now follows immediately.

Proof of Theorem 32. Follows from Lemma 36 and a padding argument analogous to that of Theorem 1. □

5.2 NQSPACE vs. $\mathsf{NQ_U}$SPACE vs. $\mathsf{NQ_U}$SPACE

By considering variants of the problems of Definition 28, in which $\epsilon(n) = 0$, $\forall n \in \mathbb{N}$, we establish the following result. See the full paper [19] for details.

Theorem 37. *For any space-constructible function* $s : \mathbb{N} \to \mathbb{N}$, *where* $s(n) = \Omega(\log n)$, *we have*

$$\mathsf{NQ_UMASPACE}(s) = \mathsf{NQ_USPACE}(s) = \mathsf{NQSPACE}(s)$$
$$= \mathsf{coPreciseQ_UMA_1SPACE}(s) = \mathsf{coC_{=}SPACE}(s).$$

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 37. However, in the case of one-sided bounded-error, we only have the partial results of Theorem 32. In particular, specializing to the case of logspace, we have $\mathsf{BQL} = \mathsf{BQ_UL} = \mathsf{Q_UMAL}$ in the two-sided bounded-error case (Lemma 18), and we have $\mathsf{RQ_UMAL} = \mathsf{RQ_UL} \subseteq \mathsf{RQL} \subseteq \mathsf{coQ_UMAL_1}$ in the one-sided bounded-error case (Lemma 36). 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 $\mathsf{RQL} = \mathsf{coQ_UMAL_1}$?

By the well-known result of Zachos and Fürer [56], $\mathsf{MA} = \mathsf{MA_1}$; 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, 24] 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 [56] 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, $\mathsf{QCMA} = \mathsf{QCMA_1}$ [24]. 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, $\mathsf{QMAL} = \mathsf{QMAL_1} \Leftrightarrow \mathsf{QCMAL} = \mathsf{QMAL_1} \Leftarrow \mathsf{QCMAL} = \mathsf{QCMAL_1}$.

Recall that $\mathsf{BQL} \subseteq \mathsf{DSPACE}(\log^2 n)$ [52] is the current best upper bound of this type. As discussed in Section 1.2, Boix-Adserà, Eldar, and Mehraban [8] have recently shown that κ-conditioned-DET $\in \mathsf{DSPACE}(\log(n)\log(\kappa(n))poly(\log\log n))$. Furthermore, we have shown that *poly*-conditioned-DET is BQL-complete. Therefore, if $\mathsf{BQL} \not\subseteq \mathsf{DSPACE}(\log^{2-\epsilon} n)$, $\forall \epsilon > 0$, then both our result and their result are essentially optimal, in terms of the relationship between condition number and needed space. It is then natural to ask if either result can be improved.

Open Problem 3. Does κ-conditioned-DET remain BQL-hard for some $\kappa(n) = n^{o(1)}$? Is κ-conditioned-DET $\in \mathsf{DSPACE}(s)$ for some $s(n) = o(\log(n)\log(\kappa(n))poly(\log\log n))$?

We conclude with a general question.

Open Problem 4. What further relationships can be established between BQL and other natural logspace complexity classes (e.g., #L, GapL, L/*poly*, etc.)?

ACKNOWLEDGMENTS

B.F. and Z.R acknowledge support from AFOSR (YIP number FA9550-18-1-0148 and FA9550-21-1-0008). B.F additionally acknowledges support from the National Science Foundation under Grant CCF-2044923 (CAREER). We would like to thank Dieter van Melkebeek, as well as the anonymous reviewers, for many helpful comments on a preliminary draft.

A A TM-BASED PROOF OF $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$=BQL

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 [50, 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 2, *poly*-conditioned-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 $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$.

Proposition 38. $\mathsf{BPL} \subseteq \mathsf{BQ_UL}$.

Proof. Suppose $\mathcal{P} = (\mathcal{P}_1, \mathcal{P}_0) \in \mathsf{BPL}$. Then there is some probabilistic TM M that recognizes $\mathcal{P}$ 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 $w \in \mathcal{P}$ 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 $w \in \mathcal{P}$, one may construct, in deterministic space $O(\log(|w|))$ a stochastic matrix $A_w \in \widehat{\mathrm{Mat}}(c(n))$ and values $x_w, y_w \in [c(n)]$ such that $A_w^t[x_w, y_w]$ is precisely the probability that M accepts w within t steps [15, 36]; this implies $\mathsf{MATPOW}(\langle A_w, x_w, y_w, \frac{2}{3}\rangle) = \mathcal{P}(w)$. Note that, as A_w is stochastic, so is A_w^t, $\forall t \in \mathbb{N}$; this implies $\sigma_1(A_w^t) \leq \sqrt{c(n)} = n^{O(1)}$, which then implies $\langle A_w, x_w, y_w, \frac{2}{3}\rangle \in \mathsf{MATPOW}_{c(n),t(n),\sqrt{c(n)},3}$. By Theorem 2, $\mathsf{MATPOW}_{c(n),t(n),\sqrt{c(n)},3} \in \mathsf{BQ_UL}$, which implies $\mathcal{P} \in \mathsf{BQ_UL}$. □

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 $\mathsf{BQL} = \mathsf{BQ_UL}$). Here, BQL is defined in terms of a logspace *quantum Turing machine* (QTM), as was the case in, for instance [25, 33, 37, 46, 50–53], rather than the equivalent model of a uniform family of *general quantum circuits* used in this paper.

For concreteness, we use the *classically controlled* logspace (general) QTM defined by Watrous [52] (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 [52] for a complete definition.

Proposition 39. $\mathsf{BQL} = \mathsf{BQ_UL}$.

Proof. Trivially, $\mathsf{BQL} \supseteq \mathsf{BQ_UL}$. We next show $\mathsf{BQL} \subseteq \mathsf{BQ_UL}$. Suppose $\mathcal{P} = (\mathcal{P}_1, \mathcal{P}_0) \in \mathsf{BQL}$. By definition, there is some QTM M such that the following conditions are satisfied: (1) on any input $w \in \mathcal{P}$ 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 $w \in \mathcal{P}_1$, then $\Pr[M \text{ accepts } w] \geq \frac{2}{3}$, and (3) if $w \in \mathcal{P}_0$, 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 $w \in \mathcal{P}$. 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 $\Phi_{M,w}^t \in \mathrm{Chan}(\mathcal{H}_{M,n})$, which implies $\sigma_1((K(\Phi_{M,w}))^t) = \sigma_1(K(\Phi_{M,w}^t)) \leq \sqrt{d(n)} = n^{O(1)}$ [40, Theorem 1]. Let $|\psi_{start}^n\rangle = |c_{start}^n\rangle|q_{start}^n\rangle \in \mathcal{H}_{M,n}$ denote the starting configuration of M on an input of length n, where $c_{start}^n \in C_n$ is the classical part of the starting configuration, and $|q_{start}^n\rangle = |0^{s(n)+b_q}\rangle \in \mathbb{C}^{Q_n}$ 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 $|\psi_{accept}^n\rangle = |c_{accept}^n\rangle|q_{start}^n\rangle \in \mathcal{H}_{M,n}$ on any input of length n. Let $A_w =$

$K(\Phi_{M,w}) \in \widehat{\mathrm{Mat}}(d^2(n))$, $x_w = \mathrm{vec}(|\psi^n_{accept}\rangle\langle\psi^n_{accept}|) \in [d^2(n)]$, and $y_w = \mathrm{vec}(|\psi^n_{start}\rangle\langle\psi^n_{start}|) \in [d^2(n)]$. Then $A^t_w[x_w, y_w]$ is precisely the probability that M accepts w within t steps. We then have $\mathsf{MATPOW}(\langle A_w, x_w, y_w, \frac{2}{3}\rangle) = \mathcal{P}(w)$ and $\langle A_w, x_w, y_w, \frac{2}{3}\rangle \in \mathsf{MATPOW}_{d^2(n),t(n),\sqrt{d(n)},3}$. To complete the proof, note that, by Theorem 2, $\mathsf{MATPOW}_{d^2(n),t(n),\sqrt{d(n)},3} \in \mathsf{BQ_UL}$, which implies $\mathcal{P} \in \mathsf{BQ_UL}$. □

REFERENCES

[1] Scott Aaronson. 2009. ON PERFECT COMPLETENESS FOR QMA. *Quantum Information and Computation* 9, 1 (2009), 0081–0089.

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

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

[4] Eric Allender and Mitsunori Ogihara. 1996. Relationships Among PL, #L, and the Determinant. *RAIRO-Theoretical Informatics and Applications* 30, 1 (1996), 1–21.

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

[6] Stuart J Berkowitz. 1984. On computing the determinant in small parallel time using a small number of processors. *Information processing letters* 18, 3 (1984), 147–150.

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

[8] Enric Boix-Adserà, Lior Eldar, and Saeed Mehraban. 2019. Approximating the Determinant of Well-Conditioned Matrices by Shallow Circuits. *arXiv preprint arXiv:1912.03824* (2019).

[9] Sergey Bravyi. 2011. Efficient algorithm for a quantum analogue of 2-SAT. *Contemp. Math.* 536 (2011), 33–48.

[10] Harry Buhrman, John Tromp, and Paul Vitányi. 2001. Time and space bounds for reversible simulation. In *International Colloquium on Automata, Languages, and Programming*. Springer, 1017–1027.

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

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

[13] C Damm. 1991. DET=L(#L). *Fachbereich Informatik der Humboldt-Universitat zu, Berlin* (1991).

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

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

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

[17] Bill Fefferman, Hirotada Kobayashi, Cedric Yen-Yu Lin, Tomoyuki Morimae, and Harumichi Nishimura. 2016. 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.

[18] Bill Fefferman and Cedric Yen-Yu Lin. 2018. A Complete Characterization of Unitary Quantum Space. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[19] Bill Fefferman and Zachary Remscrim. 2020. Eliminating intermediate measurements in space-bounded quantum computation. *arXiv preprint arXiv:2006.03530* (2020).

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

[21] Uma Girish, Ran Raz, and Wei Zhan. 2020. Quantum logspace algorithm for powering matrices with bounded norm. *arXiv preprint arXiv:2006.04880* (2020).

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

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

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

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

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

[27] Hirotada Kobayashi, Keiji Matsumoto, and Tomoyuki Yamakami. 2003. Quantum Merlin-Arthur proof systems: Are multiple Merlins more helpful to Arthur?. In *International Symposium on Algorithms and Computation*. Springer, 189–198.

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

[29] Klaus-Jörn Lange, Pierre McKenzie, and Alain Tapp. 2000. Reversible space equals deterministic space. *J. Comput. System Sci.* 60, 2 (2000), 354–367.

[30] Seth Lloyd. 1996. Universal quantum simulators. *Science* (1996), 1073–1078.

[31] Meena Mahajan and V Vinay. 1997. Determinant: Combinatorics, Algorithms, and Complexity. *Chicago Journal of Theoretical Computer Science* (1997).

[32] Chris Marriott and John Watrous. 2005. Quantum Arthur–Merlin games. *Computational Complexity* 14, 2 (2005), 122–152.

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

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

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

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

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

[38] Zachary Remscrim. 2020. The Power of a Single Qubit: Two-Way Quantum Finite Automata and the Word Problem. In *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 168)*. 139:1–139:18.

[39] Zachary Remscrim. 2021. Lower Bounds on the Running Time of Two-Way Quantum Finite Automata and Sublogarithmic-Space Quantum Turing Machines. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*.

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

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

[42] Michael Saks and Shiyu Zhou. 1999. BPhSPACE(s) in DSPACE(sˆ3/2). *Journal of computer and system sciences* 58, 2 (1999), 376–403.

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

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

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

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

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

[48] Leslie G Valiant. 1992. Why is Boolean complexity theory difficult. *Boolean Function Complexity* 169 (1992), 84–94.

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

[50] John Watrous. 1999. Space-bounded quantum complexity. *J. Comput. System Sci.* 59, 2 (1999), 281–326.

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

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

[53] John Watrous. 2009. Encyclopedia of Complexity and System Science, chapter Quantum computational complexity.

[54] John Watrous. 2009. Zero-knowledge against quantum attacks. *SIAM J. Comput.* 39, 1 (2009), 25–58.

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

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