Quantum Logspace Algorithm for Powering Matrices with Bounded Norm

Uma Girish ✉
Department of Computer Science, Princeton University, NJ, USA

Ran Raz ✉
Department of Computer Science, Princeton University, NJ, USA

Wei Zhan ✉
Department of Computer Science, Princeton University, NJ, USA

Abstract

We give a quantum logspace algorithm for powering contraction matrices, that is, matrices with spectral norm at most 1. The algorithm gets as an input an arbitrary $n \times n$ contraction matrix A, and a parameter $T \leq \text{poly}(n)$ and outputs the entries of A^T, up to (arbitrary) polynomially small additive error. The algorithm applies only unitary operators, without intermediate measurements. We show various implications and applications of this result:

First, we use this algorithm to show that the class of quantum logspace algorithms with only quantum memory and with intermediate measurements is equivalent to the class of quantum logspace algorithms with only quantum memory without intermediate measurements. This shows that the deferred-measurement principle, a fundamental principle of quantum computing, applies also for quantum logspace algorithms (without classical memory). More generally, we give a quantum algorithm with space $O(S + \log T)$ that takes as an input the description of a quantum algorithm with quantum space S and time T, with intermediate measurements (without classical memory), and simulates it unitarily with polynomially small error, without intermediate measurements.

Since unitary transformations are reversible (while measurements are irreversible) an interesting aspect of this result is that it shows that any quantum logspace algorithm (without classical memory) can be simulated by a reversible quantum logspace algorithm. This proves a quantum analogue of the result of Lange, McKenzie and Tapp that deterministic logspace is equal to reversible logspace [15].

Finally, we use our results to show non-trivial classical simulations of quantum logspace learning algorithms.

2012 ACM Subject Classification Theory of computation → Quantum complexity theory

Keywords and phrases BQL, Matrix Powering, Quantum Circuit, Reversible Computation

Digital Object Identifier 10.4230/LIPIcs.ICALP.2021.73

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://eccc.weizmann.ac.il/report/2020/087/`

Funding Research supported by the Simons Collaboration on Algorithms and Geometry, by a Simons Investigator Award and by the National Science Foundation grant No. CCF-1714779.

Acknowledgements We would like to thank Dieter van Melkebeek and Subhayan Roy Moulik for very helpful suggestions and comments on a previous version of this work. We also thank the anonymous reviewers for their thorough feedback.

1 Introduction

Quantum computers hold great promise, but in the near future their memory is likely to be limited to a small number of qubits. This motivates the study of quantum complexity classes with bounded space. The most important of these classes is the class of problems solvable in quantum logarithmic space and polynomial time, first studied by Watrous [28].

48th International Colloquium on Automata, Languages, and Programming (ICALP 2021).
Editors: Nikhil Bansal, Emanuela Merelli, and James Worrell; Article No. 73; pp. 73:1–73:20
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In the literature, there are several variants of this class. One variant, BQL, is the class of problems solvable in quantum logarithmic space and polynomial time when intermediate measurements are allowed. Another variant, $\mathsf{BQ_UL}$, is the class of problems solvable in quantum logarithmic space and polynomial time when only unitary operators are allowed and intermediate measurements are not allowed. We note that in most previous works, the class BQL allows a quantum algorithm to use both quantum and classical memory (see for example [17, 27, 6]).

Our first main result, Theorem 18, gives a quantum logspace algorithm for powering matrices, a fundamental problem in computational complexity, which is not known to be in classical (deterministic or probabilistic) logspace. Our algorithm uses only unitary operators, without intermediate measurements, and hence it places the problem of powering matrices in the class $\mathsf{BQ_UL}$.

The algorithm gets as an input an arbitrary $n \times n$ matrix A, a parameter $T \leq \text{poly}(n)$ and two indices $i, j \in \{1, \ldots, n\}$ and outputs the entry $(A^T)_{i,j}$, up to an additive error of $\frac{\|A\|^T}{\text{poly}(n)} + \frac{1}{\text{poly}(n)}$, where $\|A\|$ is the spectral norm of the matrix A. In particular, if A is a contraction matrix, that is, a matrix with spectral norm at most 1, the additive error is just $\frac{1}{\text{poly}(n)}$.

We note that by an easy reduction, our algorithm can also solve another fundamental problem in computational complexity, the problem of iterative matrix multiplication. In this problem, the input is T matrices $A_1, \ldots, A_T$ of size $n \times n$ each, and the algorithm outputs the entries of the product $A_1 \cdot \ldots \cdot A_T$.

Besides giving a quantum logspace algorithm for a basic computational problem, our results shed light on several fundamental issues regarding bounded-space quantum computations, and have additional applications.

$\mathsf{BQ_QL}$ is Equal to $\mathsf{BQ_UL}$

We consider the class of quantum logspace algorithms with only quantum memory and with intermediate measurements and refer to it by $\mathsf{BQ_QL}$. We use our algorithm for powering matrices to show that the two classes $\mathsf{BQ_QL}$ and $\mathsf{BQ_UL}$ are exactly equal (Theorem 12). Moreover, the way that this equality is proved is by a *simulation.* Our second main result, Theorem 16, proves that there is a quantum logspace algorithm without intermediate measurements, that is, a $\mathsf{BQ_UL}$ algorithm, that gets the description of a quantum logspace algorithm with intermediate measurements, without classical memory, that is, a $\mathsf{BQ_QL}$ algorithm, and simulates it with polynomially small error. Theorem 16 is even more general and shows how to simulate a quantum logspace algorithm with *unital* channels that are given as an input, while even the very restricted special case of simulating an arbitrary unitary operator within $\mathsf{BQ_UL}$ seems to us interesting.

The Deferred-Measurement Principle

The deferred measurement principle is a fundamental result in quantum computing which states that delaying measurements until the end of a computation doesn't affect the output. In order for the principle to hold, the qubits that were supposed to be measured cannot further participate in the computation from that point on. However, a $\mathsf{BQ_QL}$ algorithm can only store a logarithmic number of qubits, while the number of intermediate measurements is potentially polynomial, and hence excluding the qubits that are supposed to be measured from further participating in the computation is infeasible.

Nevertheless, Theorem 12 and Theorem 16 imply that intermediate measurements are not necessary even when the space used by the quantum algorithm is logarithmic, but the way to eliminate the intermediate measurements is not as straightforward.

Reversible Computation

Landauer introduced the concept of time-reversible computation and argued that any irreversible operation must be accompanied by entropy increase [14] (see also [2]). An interesting aspect of Theorem 12 and Theorem 16 is that they show that any quantum logspace algorithm (without classical memory) can be implemented using only time-reversible operations (except for the final measurement that gives the final output). This is a quantum analogue of the result of Lange, McKenzie and Tapp that deterministic logspace algorithms can be implemented using only time-reversible operations [15].

Classical Simulations of Quantum Learning with Bounded Memory

A line of recent works studied the power of (classical) algorithms for online learning, under memory constraints, where a bounded-space learner tries to learn a concept class from a stream of samples. These works showed that for a large class of online learning problems, any classical learning algorithm requires either super-linear memory size or a super-polynomial number of samples (see for example [24, 26, 22, 13, 21, 18, 1, 8] and the references therein).

Here, we study the relative power of quantum and classical algorithms for online learning, under memory constraints. More concretely, we study the task of distinguishing between two families of distributions over the possible samples. Corollary 23 proves that any quantum algorithm with time T and space S for distinguishing between arbitrary two families of distributions, can be simulated classically in time $\mathrm{poly}(2^{S^2+\log^2 T})$ and space $O(S^2+\log^2 T)$. Moreover, Theorem 24 proves that if one family is a singleton, that is, the task is to distinguish between one distribution over the samples and a family of different distributions, then any quantum learning algorithm with time T and space S can be simulated classically in time $\mathrm{poly}(2^S \cdot T)$ and space $O(S+\log T)$.

Thus, an intriguing open problem is whether any quantum algorithm with time T and space S for distinguishing between two arbitrary families of distributions, can be simulated classically in time $\mathrm{poly}(2^S \cdot T)$ and space $O(S+\log T)$. Theorem 22 proves that this holds if and only if $\mathsf{promiseBQ_UL} = \mathsf{promiseBPL}$.

1.1 Techniques

We start by proving a lemma that shows how to implement an arbitrary contraction matrix A as a subsystem of a unitary quantum circuit (Lemma 6). Since A is not necessarily unitary, rather than implementing A, the lemma implements the unitary matrix

$$U_H = \begin{pmatrix} H & \sqrt{\mathbf{I}_{2m} - H^2} \\ \sqrt{\mathbf{I}_{2m} - H^2} & -H \end{pmatrix}$$

where H is the Hermitian contraction $\begin{pmatrix} & A \\ A^\dagger & \end{pmatrix}$. That is, the lemma shows how to apply the transformation U_H on a unit vector (quantum state) that is also given as an input. The unitary matrix U_H is called a block-encoding of A in some literature [4, 10], which admits various constructions (see [9] for an exhibition). In particular, our construction in Lemma 6 is in unitary quantum logspace.

The proof of Lemma 6 is inspired by, and uses techniques from, Ta-Shma's algorithm that inverts well-conditioned matrices in quantum logspace [27], whose general framework traces back to [12]. In particular, as in [27], the proof goes according to the following lines: Given a Hermitian matrix H,

- First apply the phase estimation over the unitary e^{iH} so that it maps $|u_\lambda\rangle$ to $|u_\lambda\rangle|\lambda\rangle$, where u_λ is an eigenvector of H with eigenvalue λ.
- For each eigenvector apply the unitary transformation $|\lambda\rangle \to \lambda|0\rangle|\lambda\rangle + \sqrt{1-\lambda^2}|1\rangle|\lambda\rangle$ according to the eigenvalue λ. This is where contraction matrices come into play, as the eigenvalues of H are required to be in $[-1, 1]$.
- Uncompute the eigenvalues by reversing the phase estimation over e^{iH}.

As a special case of Lemma 6, when we take the contraction A to be unitary, we get a space-efficient unitary implementation of any unitary matrix (Corollary 7).

We get our algorithms for powering contraction matrices (Theorem 10 and Corollary 15) by iteratively applying the unitary matrix U_H of Lemma 6. However, since Lemma 6 implements the matrix U_H, rather than A, we need to "throw away" the unwanted dimensions introduced by U_H, by permuting them into additional dimensions.

We get our algorithm for powering arbitrary matrices (Theorem 18), by a reduction to the algorithm for powering contraction matrices, by dividing the matrix by its norm. However, the known algorithm for computing the spectral norm of a matrix, by Ta-Shma [27], only works for contraction matrices. To bypass this, we apply Ta-Shma's algorithm on the matrix A divided by its Frobenius norm (which is always larger than the spectral norm).

Finally, we get our algorithms for simulating quantum logspace algorithms with intermediate measurements, or even *unital* channels that are given as an input (Lemma 11 and Theorem 16), by reducing any unital quantum algorithm to the contraction powering problem in the m^2-dimensional space of the $m \times m$ entries of the density matrix, where $m = 2^S$ and S is the space used by the algorithm. Note that this step already doubles the space used. At the end of this step, we only get polynomially small success probability, but that success probability can be amplified to a constant using a Grover-type technique inspired by [6], resulting in Lemma 11 that simulates the computation with constant error. The error is further reduced to be polynomially small in Theorem 16. Interestingly, to reduce the error and prove Theorem 16, we use Theorem 12, so, in a way, the results are used to improve themselves.

1.2 Related Work

Independently of our work, Fefferman and Remscrim have proven closely related results to ours [7]. They took a different route from ours by proving L-reductions between several well-conditioned versions of matrix problems which turned out to be $\mathsf{BQ_UL}$-complete. In particular, they obtained a stronger version of our Theorem 12, showing that $\mathsf{BQL} = \mathsf{BQ_UL}$.

2 Preliminaries

For an integer n, let $[n] = \{0, 1, \ldots, n-1\}$. Let $\mathbb{C}$ denote the set of complex numbers, and $\mathbb{C}^{m\times n}$ denote the set of m by n complex matrices. For a matrix $A \in \mathbb{C}^{m\times n}$, let $\mathrm{vec}(A)$ be the vectorization of A, which is a vector of dimension mn formed by stacking the columns of A on top of each other, that is

$$\mathrm{vec}(A)_{i+jm} = A_{i,j}, \quad \forall i \in [m], j \in [n].$$

Let $\mathcal{U}_m$ be the set of m by m unitary matrices, and $\mathcal{D}_m$ be the set of m by m density matrices, i.e. positive semidefinite Hermitians of trace 1. The m by m identity matrix is denoted by $\mathbf{I}_m$. Let $\|A\|$ denote the spectral norm of a complex matrix A, and $\|A\|_{\mathrm{F}}$ denote the Frobenius norm.

We use ε to denote small real numbers, and $|\epsilon\rangle$ to denote vectors with small norms. When we talk about errors, approximations and ε-closeness of matrices, they are measured in spectral norms.

As we work mostly with complex numbers, we often need corresponding concentration bounds. The following is a direct corollary of the Chernoff-Hoeffding inequality:

▶ **Lemma 1.** *Let X be a random complex number with $|X| \le 1$, and $X_1, \dots, X_n$ are n independent copies of X. Then*

$$\Pr\left[\left|\frac{1}{n}(X_1 + \cdots + X_n) - \mathbf{E}[X]\right| \ge \varepsilon\right] \le 4e^{-2n\varepsilon^2}.$$

2.1 Contraction Matrices

We introduce contraction matrices and provide some useful properties, which can be found in [29, Chapter 6]:

▶ **Definition 2.** *A matrix $A \in \mathbb{C}^{m\times m}$ is a contraction if $\|A\| \le 1$. Alternatively, A is a contraction if A is in the convex hull of $\mathcal{U}_m$.*

Any eigenvalue λ of a contraction must have $|\lambda| \le 1$. If $A \in \mathbb{C}^{m\times m}$ is a contraction, then the following matrix is unitary:

$$U_A = \begin{pmatrix} A & \sqrt{\mathbf{I}_m - AA^\dagger} \\ \sqrt{\mathbf{I}_m - A^\dagger A} & -A^\dagger \end{pmatrix}$$

In particular, when $A = a$ is a real number in $[-1, 1]$, $U_a = \begin{pmatrix} a & \sqrt{1-a^2} \\ \sqrt{1-a^2} & -a \end{pmatrix}$ is a reflection. Finally, in a product of multiple contractions, individual errors will not propagate much, as we have the following lemma:

▶ **Lemma 3.** *If $A_1, \dots, A_k \in \mathbb{C}^{m\times m}$ are contractions, and $B_1, \dots, B_k \in \mathbb{C}^{m\times m}$ satisfy $\|A_i - B_i\| \le \varepsilon$ for every i, then $\|A_1 \cdots A_k - B_1 \cdots B_k\| \le (1+\varepsilon)^k - 1$. Furthermore, if $B_1, \dots, B_k$ are also contractions, then $\|A_1 \cdots A_k - B_1 \cdots B_k\| \le k\varepsilon$.*

2.2 Quantum Channels

A quantum channel (or operation), in its most general form, is a *completely-positive trace-preserving* (CPTP) map $\Phi : \mathcal{D}_m \to \mathcal{D}_n$ that maps a density matrix ρ to a density matrix $\Phi(\rho)$. We denote the set of such channels as $\mathcal{C}_{m,n}$. The *Kraus representation* of the quantum channel Φ is a set of matrices $\{E_1, \dots, E_k\}$ such that $\sum_{i=1}^k E_i^\dagger E_i = \mathbf{I}_m$, and

$$\Phi(\rho) = \sum_{i=1}^k E_i \rho E_i^\dagger.$$

The *natural representation* of Φ, denoted as $K(\Phi)$, is a matrix in $\mathbb{C}^{n^2\times m^2}$ such that $\mathrm{vec}(\Phi(\rho)) = K(\Phi)\mathrm{vec}(\rho)$ for any $\rho \in \mathcal{D}_m$. Given the Kraus representation $\{E_1, \dots, E_k\}$ of Φ, one can easily compute the natural representation $K(\Phi) = \sum_{i=1}^k \overline{E_i} \otimes E_i$.

A quantum channel Φ is *unital*, if it maps the identity to the identity of the same dimension. The Kraus representation of a unital channel is a set of square matrices $\{E_1, \ldots, E_k\}$ that additionally satisfies $\sum_{i=1}^{k} E_i E_i^\dagger = \mathbf{I}_m$. In the language of natural representation, it is known that Φ is unital if and only if $K(\Phi)$ is a contraction [20]. Notice that unitary operators and projective measurements are all unital. Our paper shows the following: one can construct in logspace a quantum circuit to simulate any arbitrary unital channel with ancillas, but without intermediate measurements.

2.3 Quantum Algorithms

A generic quantum algorithm with time T and space $S = \log m$ is specified by T quantum channels $\Phi_1, \ldots, \Phi_T \in \mathcal{C}_{m,m}$, which might depend on the inputs. We also require the channels $\Phi_1, \ldots, \Phi_T$ to be efficiently constructible, whose meaning may differ for different types of quantum algorithms, and will be specified below.

The algorithm starts from the fixed initial state $\rho_0 = |0^S\rangle\langle 0^S|$, and in the i-th step applies Φ_i on the current state, so that the state after the i-th step can be described as

$$\rho_i = \Phi_i(\rho_{i-1}) = \Phi_i \circ \Phi_{i-1} \circ \cdots \circ \Phi_1(\rho_0).$$

At the end the first qubit of the final state ρ_T is measured in the computational basis of the first qubit, where the measurement can be represented as $M_0 = |0\rangle\langle 0| \otimes \mathbf{I}_{m/2}$. The quantum algorithm outputs 0 with probability $\mathrm{Tr}[\rho_T M_0]$, and 1 with probability $1 - \mathrm{Tr}[\rho_T M_0]$. The quantum algorithm is called unitary (resp. unital), if every channel Φ_i is unitary (resp. unital).

Quantum circuit

Fix a universal quantum gate set $\mathcal{G}$, for instance Hadamard and Toffoli gates [25], and let $\mathcal{G}_S$ be the set of gates in $\mathcal{G}$ on S qubits. Let $\mathcal{M}_S$ be the set of single-qubit measurements on S qubits.

When the input of the problem is from domain X, the quantum circuit is specified by a mapping $\Phi_\mathrm{D} : X \times [T] \to \mathcal{G}_S \cup \mathcal{M}_S$ such that $\Phi_{i+1} = \Phi_\mathrm{D}(x, i)$ for every $i \in [T]$, where $x \in X$ is the input, and Φ_D can be computed deterministically in time $O(T)$ and space $O(S)$. The quantum algorithm decides a function $f : X \to \{0, 1\}$ with error ε if:

$$\forall x \in X, \quad |\mathrm{Tr}[\rho_T M_0] - f(x)| \leq \varepsilon.$$

Now, $\mathsf{BQ_QL}$ is the class of boolean-function families where $f_n : \{0,1\}^n \to \{0,1\}$ can be decided by a quantum circuit with time poly(n), space $O(\log n)$ and error $1/3$. The function is further in the class $\mathsf{BQ_UL}$ if there is no intermediate measurements, i.e. the range of Φ_D is $\mathcal{G}_S$. We define $\mathsf{promiseBQ_QL}$ and $\mathsf{promiseBQ_UL}$ similarly, but the domain of each f_n can be a subset of $\{0,1\}^n$.

Quantum learning algorithm

For a quantum online learning algorithm with Γ being the set of samples, there exists a mapping $\Phi_\mathrm{L} : \Gamma \to \mathcal{C}_{m,m}$ such that $\Phi_i = \Phi_\mathrm{L}(z_i)$ where $z_i \in \Gamma$ is the sample received in the i-th step, and each entry of $K(\Phi_\mathrm{L}(z_i))$ can be computed deterministically in time $O(T)$ and space $O(S)$.

Let $\mathcal{P}(\Gamma)$ be the collection of all probability distributions over Γ. For any distribution $D \in \mathcal{P}(\Gamma)$, let D^T be T i.i.d copies of D, so that $z \sim D^T$ means that each sample z_i is independently drawn from D. Let $\mathcal{X}, \mathcal{Y}$ be two disjoint subsets of $\mathcal{P}(\Gamma)$. The quantum learning algorithm distinguishes $\mathcal{X}$ and $\mathcal{Y}$ with error ε if:

$$\forall D \in \mathcal{X}, \quad \mathop{\mathbf{E}}_{z \sim D^T}[\mathrm{Tr}[\rho_T M_0]] \geq 1 - \varepsilon$$
$$\forall D \in \mathcal{Y}, \quad \mathop{\mathbf{E}}_{z \sim D^T}[\mathrm{Tr}[\rho_T M_0]] \leq \varepsilon.$$

And for $\varepsilon = 1/3$, we simply say that the quantum learning algorithm distinguishes $\mathcal{X}$ and $\mathcal{Y}$.

Other specifications

Notice that even in the unitary algorithms where intermediate measurements are not generally allowed, a constant number of intermediate measurements are still available because of the principle of deferred measurements (see e.g. [19, Section 4.4]), which will only increase the time and space by a constant. This means the error ε in both definitions above can be safely amplified to any constant power, and the specific constant error $1/3$ can be replaced by any constant in $[0, 1/2)$.

For constructible functions $t(n) = \Omega(n)$ and $s(n) = \Omega(\log n)$, define $\mathsf{BPTISP}(t(n), s(n))$ as the class of boolean functions families that can be decided by a classical randomized algorithm with time $O(t(n))$ and space $O(s(n))$, and $\mathsf{promiseBPTISP}(t(n), s(n))$ accordingly. The classical randomized logspace class is defined as $(\mathsf{promise})\mathsf{BPL} = (\mathsf{promise})\mathsf{BPTISP}(\mathrm{poly}(n), \log(n))$.

Phase estimation

Given the dimension m and the error parameter $\varepsilon > 0$, the *phase estimation* circuit (see e.g. [19, Section 5.2]) acts on an input register of dimension m and an estimation register of dimension $2^\ell = O(1/\varepsilon)$. The circuit is with time $O(2^\ell)$ and space $O(\ell + \log m)$, and accesses 2^ℓ oracle calls to the controlled-U gates, where $U \in \mathcal{U}_m$ is an arbitrary unitary matrix. For each $j \in [2^\ell]$, define $\lambda(j) = 2j\pi/2^\ell - \pi$, and for any $\lambda \in [-\pi, \pi]$, let $J(\lambda) = \{j \in [2^\ell] \mid |\lambda(j) - \lambda| \leq \varepsilon\}$. If v is a unit eigenvector of U with eigenvalue $e^{i\lambda}$, the circuit maps $v \otimes |0^\ell\rangle$ to

$$\sum_{j=0}^{2^\ell - 1} \alpha_j v \otimes |j\rangle,$$

so that

$$\sum_{j \in J(\lambda)} |\alpha_j|^2 \geq 1 - \varepsilon^2.$$

Given a Hermitian contraction $H \in \mathbb{C}^{m \times m}$, let P_H be the above phase estimation circuit with $U = e^{iH}$, and $P_{H,\varepsilon}$ be the above phase estimation circuit where U is replaced with the Hamiltonian simulation circuit presented in [27] which differs from e^{iH} by an error of $2^{-\ell}\varepsilon$. Notice that $P_{H,\varepsilon}$ is a unitary quantum circuit with $\mathrm{poly}(m/\varepsilon)$ and space $O(\log(m/\varepsilon))$, and by Lemma 3 we have $\|P_{H,\varepsilon} - P_H\| \leq \varepsilon$.

Since H only has eigenvalues in $[-1, 1]$, we slightly modify the definition of $\lambda(j)$ so that it's truncated at ± 1, that is

$$\lambda(j) = \begin{cases} 2j\pi/2^\ell - \pi & \text{if } 2j\pi/2^\ell - \pi \in [-1, 1] \\ \mathrm{sgn}(2j\pi/2^\ell - \pi) & \text{otherwise} \end{cases}$$

which will only make $J(\lambda)$ larger for $\lambda \in [-1, 1]$.

Every unit eigenvector v of H with eigenvalue λ is also a unit eigenvector of e^{iH} with eigenvalue $e^{i\lambda}$. Therefore for any two unit eigenvectors u, v of H, we have

$$(u^\dagger \otimes \langle j|) P_H \big(v \otimes |0^\ell\rangle\big) = \begin{cases} \alpha_j & \text{if } u = v \\ 0 & \text{if } u \perp v. \end{cases}$$

In other words, since P_H is unitary,

$$(u^\dagger \otimes \langle 0^\ell|) P_H^{-1} \big(v \otimes |j\rangle\big) = \begin{cases} \overline{\alpha_j} & \text{if } u = v \\ 0 & \text{if } u \perp v. \end{cases}$$

That means the projection of $P_H^{-1}\big(v \otimes |j\rangle\big)$ onto $\mathbb{C}^m \otimes |0^\ell\rangle$ is along $v \otimes |0^\ell\rangle$ and has amplitude $\overline{\alpha_j}$. Combing the above observations we get the following lemma:

▶ **Lemma 4.** *Given a Hermitian contraction $H \in \mathbb{C}^{m\times m}$ and $\varepsilon > 0$, there is a unitary quantum circuit $P_{H,\varepsilon}$ with time $\mathrm{poly}(m/\varepsilon)$ and space $O(\log(m/\varepsilon))$ that is ε-close to a unitary operator P_H, which satisfies the following: There is a parameter $\ell = O(\log(1/\varepsilon))$, such that if v is a unit eigenvector of H with eigenvalue $\lambda \in [-1, 1]$, then*

$$P_H(v \otimes |0^\ell\rangle) = \sum_{j=0}^{2^\ell - 1} \alpha_j v \otimes |j\rangle, \ \textit{where} \sum_{j \in J(\lambda)} |\alpha_j|^2 \geq 1 - \varepsilon^2.$$

Moreover, for every $j \in [2^\ell]$,

$$P_H^{-1}(v \otimes |j\rangle) = \overline{\alpha_j} v \otimes |0^\ell\rangle + |\perp\rangle,$$

where $|\perp\rangle$ is a vector orthogonal to $\mathbb{C}^m \otimes |0^\ell\rangle$.

Pure State Preparation

Our results involve the simplest form of the *quantum state preparation* problem, which is to map the initial state $|0^S\rangle$ to a given pure state. With the efficient Solovay-Kitaev Theorem in [17], we have the following:

▶ **Lemma 5.** *Given $m = 2^S$, a unit vector $v \in \mathbb{C}^m$ and $\varepsilon > 0$, there is unitary quantum circuit Q_v on S qubits with time $O(m \cdot \mathrm{polylog}(1/\varepsilon))$ and space $O(\log(m/\varepsilon))$ such that $\|Q_v|0^S\rangle - v\|_2 \leq \varepsilon$.*

3 Quantum Implementations of Contractions

▶ **Lemma 6.** *Given a contraction $A \in \mathbb{C}^{m\times m}$ and $\varepsilon > 0$, there is a unitary quantum circuit Q_A with time $\mathrm{poly}(m/\varepsilon)$ and space $O(\log(m/\varepsilon))$, and a parameter $\ell = O(\log(1/\varepsilon))$, such that for unit vector v of dimension $4m$, $\|Q_A(v \otimes |0^\ell\rangle) - (V_A v) \otimes |0^\ell\rangle\|_2 \leq \varepsilon$, where*

$$V_A = \mathrm{diag}(U_A, U_{A^\dagger}) = \begin{pmatrix} A & \sqrt{\mathbf{I}_m - AA^\dagger} & & \\ \sqrt{\mathbf{I}_m - A^\dagger A} & -A^\dagger & & \\ & & A^\dagger & \sqrt{\mathbf{I}_m - A^\dagger A} \\ & & \sqrt{\mathbf{I}_m - AA^\dagger} & -A \end{pmatrix}$$

Proof. Let H be the Hermitian contraction $\begin{pmatrix} & A \\ A^\dagger & \end{pmatrix}$. Notice that

$$\begin{aligned} U_H &= \begin{pmatrix} H & \sqrt{\mathbf{I}_{2m} - H^2} \\ \sqrt{\mathbf{I}_{2m} - H^2} & -H \end{pmatrix} \\ &= \begin{pmatrix} & A & \sqrt{\mathbf{I}_m - AA^\dagger} & \\ A^\dagger & & & \sqrt{\mathbf{I}_m - A^\dagger A} \\ \sqrt{\mathbf{I}_m - AA^\dagger} & & & -A \\ & \sqrt{\mathbf{I}_m - A^\dagger A} & -A^\dagger & \end{pmatrix} \end{aligned}$$

which differs from V_A only by permutations:

$$V_A = \begin{pmatrix} \mathbf{I}_m & & & \\ & & & \mathbf{I}_m \\ & \mathbf{I}_m & & \\ & & \mathbf{I}_m & \end{pmatrix} \cdot U_H \cdot \begin{pmatrix} & & \mathbf{I}_m & \\ \mathbf{I}_m & & & \\ & \mathbf{I}_m & & \\ & & & \mathbf{I}_m \end{pmatrix}$$

Since the permutations are only on two qubits, it suffices to implement U_H on v up to error ε.

Let $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$ where both v_1 and v_2 are of dimension $2m$. Suppose H has the eigen decomposition $H = \sum\limits_{k=1}^{2m} \lambda_k u_k^\dagger u_k$, and v_1, v_2 are decomposed into this eigenbasis as

$$v_1 = \sum_{k=1}^{2m} \omega_k^{(0)} u_k, \quad v_2 = \sum_{k=1}^{2m} \omega_k^{(1)} u_k, \quad \text{where } \sum_{k=1}^{2m} \left|\omega_k^{(0)}\right|^2 + \sum_{k=1}^{2m} \left|\omega_k^{(1)}\right|^2 = 1.$$

Since v can be written as $|0\rangle \otimes v_1 + |1\rangle \otimes v_2$, applying the phase estimation circuit P_{H,ε_1} in Lemma 4 on $v \otimes |0^\ell\rangle$ results in:

$$\begin{aligned} &\sum_{k=1}^{2m} \sum_{j=0}^{2^\ell - 1} \omega_k^{(0)} \alpha_{j,k} |0\rangle \otimes u_k \otimes |j\rangle + \sum_{k=1}^{2m} \sum_{j=0}^{2^\ell - 1} \omega_k^{(1)} \alpha_{j,k} |1\rangle \otimes u_k \otimes |j\rangle + |\epsilon_1\rangle \\ =& \sum_{k=1}^{2m} \sum_{j \in J(\lambda_k)} \omega_k^{(0)} \alpha_{j,k} |0\rangle \otimes u_k \otimes |j\rangle + \sum_{k=1}^{2m} \sum_{j \in J(\lambda_k)} \omega_k^{(1)} \alpha_{j,k} |1\rangle \otimes u_k \otimes |j\rangle + |\epsilon_2\rangle. \end{aligned}$$

where for each k it holds $\sum_{j \in J(\lambda_k)} |\alpha_{j,k}|^2 \geq 1 - \varepsilon_1^2$. Here ε_1 is an error parameter to be determined later, and $\ell = O(\log(1/\varepsilon_1))$. The error vector $|\epsilon_1\rangle$ is introduced due to the difference between P_{H,ε_1} and P_H, and thus $\||\epsilon_1\rangle\|_2 \leq \|P_{H,\varepsilon_1} - P_H\| \leq \varepsilon_1$. The error vector $|\epsilon_2\rangle - |\epsilon_1\rangle$ is a weighted sum of $4m$ orthogonal error vectors, with lengths at most ε_1 and weights $\omega_k^{(0)}, \omega_k^{(1)}$, and thus has length at most ε_1. Therefore $\||\epsilon_2\rangle\|_2 \leq 2\varepsilon_1$.

Now apply the following unitary transformation on the first qubit and last ℓ qubits:

$$\begin{aligned} &|0\rangle|j\rangle \to \lambda(j)|0\rangle|j\rangle + \sqrt{1 - \lambda(j)^2}|1\rangle|j\rangle \\ &|1\rangle|j\rangle \to \sqrt{1 - \lambda(j)^2}|0\rangle|j\rangle - \lambda(j)|1\rangle|j\rangle \end{aligned}$$

which gives

$$\begin{aligned} &\sum_{k=1}^{2m} \sum_{j \in J(\lambda_k)} \omega_k^{(0)} \alpha_{j,k} \left[\lambda(j)|0\rangle + \sqrt{1 - \lambda(j)^2}|1\rangle\right] \otimes u_k \otimes |j\rangle \\ +& \sum_{k=1}^{2m} \sum_{j \in J(\lambda_k)} \omega_k^{(1)} \alpha_{j,k} \left[\sqrt{1 - \lambda(j)^2}|0\rangle - \lambda(j)|1\rangle\right] \otimes u_k \otimes |j\rangle + |\epsilon_3\rangle \end{aligned}$$

This unitary transformation can be implemented as a serial combination of 2^ℓ single-qubit unitaries $U_{\lambda(j)}$ controlled by the last ℓ qubits representing j. Each one of them can be constructed up to error $2^{-\ell}\varepsilon_1$ in time $\mathrm{polylog}(1/\varepsilon_1)$ and space $O(\log(1/\varepsilon_1))$ by [17, Theorem 7]. Therefore by Lemma 3 we have $\||\epsilon_3\rangle\|_2 \leq \||\epsilon_2\rangle\|_2 + \varepsilon_1 \leq 3\varepsilon_1$.

Finally applying the reverse phase estimation P^{-1}_{H,ε_1} gives the following state, where $|\perp\rangle$ is orthogonal to $\mathbb{C}^2 \otimes \mathbb{C}^{2m} \otimes |0^\ell\rangle$:

$$
\begin{aligned}
&\sum_{k=1}^{2m} \sum_{j\in J(\lambda_k)} |\alpha_{j,k}|^2 \omega_k^{(0)} \left[\lambda(j)|0\rangle + \sqrt{1-\lambda(j)^2}|1\rangle\right] \otimes u_k \otimes |0^\ell\rangle \\
&\quad + \sum_{k=1}^{2m} \sum_{j\in J(\lambda_k)} |\alpha_{j,k}|^2 \omega_k^{(1)} \left[\sqrt{1-\lambda(j)^2}|0\rangle - \lambda(j)|1\rangle\right] \otimes u_k \otimes |0^\ell\rangle + |\epsilon_4\rangle + |\perp\rangle \\
=&\sum_{k=1}^{2m} \sum_{j\in J(\lambda_k)} |\alpha_{j,k}|^2 \omega_k^{(0)} \left[\lambda_k|0\rangle + \sqrt{1-\lambda_k^2}|1\rangle\right] \otimes u_k \otimes |0^\ell\rangle \\
&\quad + \sum_{k=1}^{2m} \sum_{j\in J(\lambda_k)} |\alpha_{j,k}|^2 \omega_k^{(1)} \left[\sqrt{1-\lambda_k^2}|0\rangle - \lambda_k|1\rangle\right] \otimes u_k \otimes |0^\ell\rangle + |\epsilon_5\rangle + |\perp\rangle \\
=&\sum_{k=1}^{2m} \omega_k^{(0)} \left[\lambda_k|0\rangle + \sqrt{1-\lambda_k^2}|1\rangle\right] \otimes u_k \otimes |0^\ell\rangle \\
&\quad + \sum_{k=1}^{2m} \omega_k^{(1)} \left[\sqrt{1-\lambda_k^2}|0\rangle - \lambda_k|1\rangle\right] \otimes u_k \otimes |0^\ell\rangle + |\epsilon_6\rangle + |\perp\rangle \\
=&\sum_{k=1}^{2m} \omega_k^{(0)} \left[U_H(|0\rangle \otimes u_k)\right] \otimes |0^\ell\rangle + \sum_{k=1}^{2m} \omega_k^{(1)} \left[U_H(|1\rangle \otimes u_k)\right] \otimes |0^\ell\rangle + |\epsilon_6\rangle + |\perp\rangle \\
=&\left[U_H(|0\rangle \otimes v_1)\right] \otimes |0^\ell\rangle + \left[U_H(|1\rangle \otimes v_2)\right] \otimes |0^\ell\rangle + |\epsilon_6\rangle + |\perp\rangle \\
=&(U_H v) \otimes |0^\ell\rangle + |\epsilon_6\rangle + |\perp\rangle.
\end{aligned}
$$

Here $\||\epsilon_4\rangle\|_2 \leq \||\epsilon_3\rangle\|_2 + \|P^{-1}_{H,\varepsilon_1} - P^{-1}_H\| \leq 4\varepsilon_1$. Also, similar to the reasoning for $|\epsilon_2\rangle - |\epsilon_1\rangle$, since for every k, $1 - \varepsilon_1^2 \leq \sum_{j\in J(\lambda_k)} |\alpha_{j,k}|^2 \leq 1$, and for every $j \in J(\lambda_k)$, $\|U_{\lambda(j)} - U_{\lambda_k}\|_2 \leq |\lambda(j) - \lambda_k| \leq \varepsilon_1$, we have

$$\||\epsilon_6\rangle\|_2 \leq \||\epsilon_5\rangle\|_2 + \varepsilon_1^2 \leq \||\epsilon_4\rangle\|_2 + \varepsilon_1 + \varepsilon_1^2 \leq 6\varepsilon_1.$$

Finally, notice that both $(U_H v) \otimes |0^\ell\rangle$ and $(U_H v) \otimes |0^\ell\rangle + |\epsilon_6\rangle + |\perp\rangle$ are unit vectors, while $|\perp\rangle$ is orthogonal to $(U_H v) \otimes |0^\ell\rangle$, so we have

$$\left|\left((U_H v)^\dagger \otimes \langle 0^\ell|\right)\left((U_H v) \otimes |0^\ell\rangle + |\epsilon_6\rangle + |\perp\rangle\right)\right| = \left|1 + \left((U_H v)^\dagger \otimes \langle 0^\ell|\right)|\epsilon_6\rangle\right| \geq 1 - \||\epsilon_6\rangle\|_2,$$

which implies that $\||\epsilon_6\rangle + |\perp\rangle\|_2 \leq \sqrt{2\||\epsilon_6\rangle\|_2}$. Therefore it suffices to take $\varepsilon_1 \leq \varepsilon^2/12$, and the theorem follows. ◀

As a by product, when we take the contraction A in Lemma 6 to be unitary, we get the unitary implementation of any unitary matrix, with the number of ancillas only depending on the error:

▶ **Corollary 7.** *Given a unitary matrix $U \in \mathcal{U}_m$ and $\varepsilon > 0$, there is a unitary quantum circuit Q_U with time $\mathrm{poly}(m/\varepsilon)$ and space $O(\log(m/\varepsilon))$, and a parameter $\ell = O(\log(1/\varepsilon))$, such that for any unit vector v of dimension m, $\|Q_U(v \otimes |0^\ell\rangle) - (Uv) \otimes |0^\ell\rangle\|_2 \leq \varepsilon$.*

Proof. Use the exact same circuit in Lemma 6 by adding two ancilla qubits to v initialized at $|00\rangle$. Notice that $V_U = \text{diag}(U, -U^\dagger, U^\dagger, -U)$, and thus the output state is ε close to $[V_U(|00\rangle \otimes v)] \otimes |0^\ell\rangle = |00\rangle \otimes (Uv) \otimes |0^\ell\rangle$. Rearranging the order of qubits and the claim follows. ◀

Finally, for permutation matrices, we present a simple unitary implementation without any ancillas by decomposing it into transpositions.

▶ **Lemma 8.** *Given a permutation $\sigma \in S_m$ and $\varepsilon > 0$, there is a unitary quantum circuit U with time $\text{poly}(m/\varepsilon)$ and space $O(\log(m/\varepsilon))$, such that $\|U - P_\sigma\| \leq \varepsilon$, where $P_\sigma \in \{0,1\}^{m\times m}$ is the matrix representation of σ.*

4 Contraction Powering in Quantum Logspace

▶ **Definition 9** (Contraction Powering). *Given $m = 2^S$, a contraction $A \in \mathbb{C}^{m\times m}$, a positive integer T in unary, and two vectors $v, w \in \mathbb{C}^m$ with $\|v\|_2 = \|w\|_2 = 1$ as the input, it is promised that $|w^\dagger A^T v|^2$ is either in $[0, 1/3]$ or $[2/3, 1]$, and the goal of the* CONTRACTION-POWERING *problem is to distinguish between the two cases.*

▶ **Theorem 10.** CONTRACTIONPOWERING $\in$ promiseBQ$_\mathsf{U}$L. *Moreover, given the same input (m, A, T, v, w) but without the promise on $|w^\dagger A^T v|^2$, while also given an error parameter $\varepsilon > 0$, there is a unitary quantum circuit W with time $\text{poly}(mT/\varepsilon)$ and space $S' = O(\log(mT/\varepsilon))$ such that $|\langle 0^{S'}|W|0^{S'}\rangle|^2$ is ε-close to $|w^\dagger A^T v|^2$.*

Proof. First, let Q_v and Q_w be the circuits preparing states v and w with error $\varepsilon/8$ in Lemma 5 respectively. Since

$$\left| |\langle 0^S|Q_w^\dagger A^T Q_v|0^S\rangle|^2 - |w^\dagger A^T v|^2 \right| \leq 4\|Q_v|0^S\rangle - v\|_2 + 4\|Q_w|0^S\rangle - w\|_2 \leq \varepsilon/2,$$

in the rest of the proof we can safely assume that $Q_v|0^S\rangle = v$ and $Q_w|0^S\rangle = w$ while halving ε.

Let $\ell = O(\log(T/\varepsilon))$ be the one in Lemma 6 with error parameter $(2T)^{-1}\varepsilon$. The circuit works on three parts of qubits: the counter register C of dimension $2T$, the vector register of dimension m, and ℓ ancilla qubits. The circuit starts by preparing $|0\rangle_C \otimes v \otimes |0^\ell\rangle$ by applying Q_v. Then repeat the following two steps for T times:

1. Apply V_A on the last two qubits of the timer register and the entire vector register by Lemma 6;
2. Apply the permutation $|0\rangle \to |0\rangle$, $|2T-2\rangle \to |1\rangle$, $|2T-1\rangle \to |2\rangle$, $|i\rangle \to |i+2\rangle$, $\forall i = 1, \ldots, 2T-3$. on the counter register by Lemma 8.

Finally, apply $Q_w^\dagger$ on the vector register and measure with the projection onto $|0\rangle_C \otimes |0^S\rangle \otimes |0^\ell\rangle$.

To prove the correctness of the algorithm, we first assume that all the implementations in Lemma 6 and Lemma 8 are errorless, i.e. the evolution is completely within the subspace $\mathbb{C}^{2T} \otimes \mathbb{C}^m \otimes |0^\ell\rangle$. Then it suffices to notice that V_A is block-diagonal, so that step 1 acts locally on the T subspaces spanned by $|2i\rangle_C$ and $|2i+1\rangle_C$. Therefore after the i-th application of V_A, the projection of the current state onto $|j\rangle_C$ is always 0 for $j \geq 2i$, and thus before each application of V_A, the projection onto $|1\rangle_C$ is always 0. So the state after the i-th repetition is $|0\rangle_C \otimes (A^i v) + |\perp\rangle$, where $|\perp\rangle$ is orthogonal to $|0\rangle_C$. The output probability is then

$$\left| \left(\langle 0|_C \otimes \langle 0^S|\right) \left(\mathbf{I}_{2T} \otimes U_w^\dagger\right) \left(|0\rangle_C \otimes (A^T v) + |\perp\rangle\right) \right|^2 = |w^\dagger A^T v|^2.$$

Now each step in the repetition introduces an error of $(2T)^{-1}\varepsilon$. Therefore, by Lemma 3, the total error of the unitary quantum circuit W, compared to the above ideal case, is at most ε. ◀

5 Equivalence of Unital and Unitary Quantum Logspace

5.1 Simulating Unital Quantum Logspace with Constant Error

▶ **Lemma 11.** *Given a unital quantum algorithm with time T and space $S = \log m$ specified by the natural representations $K(\Phi_1), \ldots, K(\Phi_T) \in \mathbb{C}^{m^2 \times m^2}$, and an error parameter $\varepsilon > 0$, there is a unitary quantum circuit with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$, such that if the original unital circuit outputs 0 with probability p, then the unitary circuit outputs 0 with probability $\sin^2(p + \alpha)$, where $|\alpha| \le \varepsilon$.*

Proof. We can always assume that S is an odd number and $m \ge \max(4\varepsilon^{-1}, 8)$ by adding dummy dimensions. As each $K(\Phi_i)$ is a contraction, the following matrix A of dimension m^2T is also a contraction:

$$A = \begin{pmatrix} & & & & K(\Phi_T) \\ K(\Phi_1) & & & & \\ & K(\Phi_2) & & & \\ & & \ddots & & \\ & & & K(\Phi_{T-1}) & \end{pmatrix}$$

Since the final state of the unital quantum algorithm is $\rho_T = \Phi_T \circ \cdots \circ \Phi_1(\rho_0)$, we can rewrite the output probability of the unital quantum algorithm as

$$\begin{aligned} p = \mathrm{Tr}[\rho_T M_0] &= \mathrm{vec}(M_0)^\dagger \mathrm{vec}(\rho_T) = \mathrm{vec}(M_0)^\dagger K(\Phi_T) \cdots K(\Phi_1) \mathrm{vec}(\rho_0) \\ &= \left(\mathrm{vec}(M_0)^\dagger \otimes \langle 0|\right) A^T \left(\mathrm{vec}(\rho_0) \otimes |0\rangle\right) \end{aligned}$$

Let $v = \mathrm{vec}(\rho_0) \otimes |0\rangle$ which is already a unit vector. Since $\|\mathrm{vec}(M_0)\|_2 = \sqrt{m/2}$, let $w = \sqrt{\frac{2}{m}}\mathrm{vec}(M_0) \otimes |0\rangle$. Let ε_1 be the error parameter to be determined later. Theorem 10 constructs a unitary quantum algorithm W with time $\mathrm{poly}(mT/\varepsilon_1)$ and space $S' = O(\log(mT/\varepsilon_1))$, such that $|\langle 0^{S'}|W|0^{S'}\rangle|^2$ is ε_1-close to $2p^2/m$. Therefore $|\langle 0^{S'}|W|0^{S'}\rangle|$ is $\sqrt{\varepsilon_1}$-close to $\sqrt{\frac{2}{m}}p$.

Let

$$R = \left(\mathbf{I}_{S'} - 2W|0^{S'}\rangle\langle 0^{S'}|W^\dagger\right)\left(\mathbf{I}_{S'} - 2|0^{S'}\rangle\langle 0^{S'}|\right)$$

be the rotation on the subspace spanned by $|0^{S'}\rangle$ and $W|0^{S'}\rangle$, of degree $2\cos^{-1}|\langle 0^{S'}|W|0^{S'}\rangle|$. By the estimation

$$\frac{\pi}{2} - x - \frac{1}{4}x^3 \le \cos^{-1} x \le \frac{\pi}{2} - x, \forall x \in [0,1],$$

and since $(x+y)^3 \le 4(x^3 + y^3)$ for non-negative x, y, it can be calculated that the degree of the rotation R is in the range

$$\left[\pi - 2\sqrt{\frac{2}{m}}p - 4\sqrt{\varepsilon_1} - \frac{4}{m}\sqrt{\frac{2}{m}},\ \pi - 2\sqrt{\frac{2}{m}}p + 2\sqrt{\varepsilon_1}\right].$$

Since S is an odd number, $k = \sqrt{m/8}$ is an integer. Applying R for k times will rotate $|0^{S'}\rangle$ by a degree of $k\pi - p - \alpha$, where $|\alpha| \leq \sqrt{2m\varepsilon_1} + 2m^{-1}$. Therefore the projective measurement of the state $R^k|0^{S'}\rangle$ onto the subspace orthogonal to $|0^{S'}\rangle$ outputs 0 with probability $\sin^2(p+\alpha)$. Let $\varepsilon_1 = (8m)^{-1}\varepsilon^2$, and notice that $2m^{-1} \leq \varepsilon/2$, so that we have $|\alpha| < \varepsilon$, and the circuit R^k is unitary with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$. ◀

▶ **Theorem 12.** $\mathsf{BQ_QL} = \mathsf{BQ_UL}$, *and* $\mathsf{promiseBQ_QL} = \mathsf{promiseBQ_UL}$.

Proof. Clearly $\mathsf{BQ_QL} \supseteq \mathsf{BQ_UL}$, and $\mathsf{promiseBQ_QL} \supseteq \mathsf{promiseBQ_UL}$. To prove the other direction, notice that quantum circuits are unital, therefore by Lemma 11 with $\varepsilon = 0.01$ they can be simulated by unitary quantum circuits with polynomial time and logarithmic space. Since the original output probability p is promised to be in $[0, 1/3]$ or $[2/3, 1]$, the value of $\sin^2(p+\alpha)$ is in $[0, 0.12]$ or $[0.37, 1]$ respectively, and thus it suffices to perform a constant rounds of amplification in order to bring the error down to less than $1/3$. ◀

▶ Remark 13. Though we proved Theorem 12 via the contraction powering algorithm, the unitary quantum circuit that simulates a given quantum circuit with intermediate measurements can be more simply constructed without using Lemma 6. In details, given a channel Φ in the quantum circuit, we can directly write out the natural representations $K(\Phi)$, and apply the matrix on the vectorized density matrix $\mathrm{vec}(\rho)$:

- If Φ is a unitary quantum gate U, then $K(\Phi) = \overline{U} \otimes U$ which can be implemented by applying U and then $\overline{U}$;
- If Φ is a single-qubit measurement, then $K(\Phi)$ is a diagonal matrix with diagonal entries in $\{0, 1\}$. It can be implemented using a similar "permute and throw away" technique as in Theorem 10, which after applied T times increases the dimension (instead of the space!) by a factor of T.

And the resulting circuit can be amplified in the same way as in Lemma 11.

5.2 Simulating Unital Quantum Logspace with Small Error

Now we can improve the result in Lemma 11 to arbitrarily small error (namely the probability of outputting 0 is $(p + \alpha)$ instead of $\sin^2(p + \alpha)$). Interestingly, the improvement relies on a stronger version of Theorem 10, which in turn relies on Theorem 12. In a way, we use these results to improve themselves!

We start with the stronger version of Theorem 10, which outputs the numerical value of $|w^\dagger A^T v|^2$ instead of outputting 0 with such probability. Here the quantum circuit outputs a number by a final measurement over the computational basis.

▶ **Lemma 14.** *Given $m = 2^S$, a contraction $A \in \mathbb{C}^{m\times m}$, a positive integer T, two unit vectors $v, w \in \mathbb{C}^m$ and an error parameter $\varepsilon > 0$, there is a unitary quantum circuit with time* $\mathrm{poly}(mT/\varepsilon)$ *and space* $O(\log(mT/\varepsilon))$ *such that with probability* $1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$*, it outputs* $|w^\dagger A^T v|^2$ *with additive error* ε.

Proof. Theorem 10 provides a unitary quantum circuit W with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$ which outputs 0 with probability p such that $\left|p - |w^\dagger A^T v|^2\right| \leq \varepsilon/2$. By Marriott-Watrous amplification [16, Theorem 3.3], there is a quantum circuit W' with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$ *with* intermediate measurements, that uses W and W^{-1} as sub-circuits, and with probability $1 - \delta = 1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$ outputs a value $\tilde{p}$ such that $|\tilde{p} - p| \leq \varepsilon/4$.

Since the resulting circuit W' is not unitary, we would like to use Theorem 12 to compute unitarily each bit in the output value $\tilde{p}$ of W'. Furthermore, using the result in [5] that $\mathsf{BQ_UL} = \mathsf{Q_UL}(1-2^{-\mathrm{poly}(n)}, 2^{-\mathrm{poly}(n)})$ (which stands for unitary quantum logspace with exponentially small error) the total error probability can be reduced down to $2^{-\mathrm{poly}(mT/\varepsilon)}$. Assuming that every bit in $\tilde{p}$ is 0 with probability either in $[0, 1/3]$ or $[2/3, 1]$, then for $1 \le i \le \lceil \log(1/\varepsilon) \rceil + 2$, we let W_i be the unitary quantum circuit that computes the i-th bit of $\tilde{p}$ with exponentially small error. Ideally, the outputs of W_i combined together would ε-approximate $|w^\dagger A^T v|^2$.

However, the value $\tilde{p}$ outputted by the Marriott-Watrous amplification might be different in each W_i, so the final approximation assembled can be totally wrong (for instance, when $p = 0.5$, the outputs $\tilde{p} = \overline{0.1000\dots}$ and $\tilde{p} = \overline{0.0111\dots}$ might be assembled to $\overline{0.1111\dots}$). Moreover, the error reduction in [5] may have unpredictable results, as the promises on the distributions of the bits in $\tilde{p}$ are not guaranteed (again when $p = 0.5$, the most significant bit of $\tilde{p}$ is equally distributed on 0 and 1).

Fortunately, we can solves both problems by computing from the most significant bit to the least significant bit. We maintain a value $q \in [0, 1]$ which is initialized to 0. For each $i = 1$ to $\lceil \log(1/\varepsilon) \rceil + 2$ do the following: Run the modified circuit W_i which outputs the i-th bit of $(\tilde{p} - q)$ instead of $\tilde{p}$. To deal with case when $\tilde{p} - q$ is outside of $[0, 2^{-i+1})$, if $\tilde{p} - q < 0$ it outputs 0, and if $\tilde{p} - q \ge 2^{-i+1}$ it outputs 1. Let the output bit be b_i and update q to $q + b_i \cdot 2^{-i}$.

We claim that with probability $1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$, $|q - p| \le \varepsilon/2$. First notice that, if every bit in $\tilde{p}$ is 0 with probability in $[0, 2\delta] \cup [1 - 2\delta, 1]$, then the error reduction will work as intended, while with probability $1 - O(\delta \log(1/\varepsilon)) = 1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$ the value $\tilde{p}$ is the same in each circuit W_i, so that q is also the same as $\tilde{p}$.

Now let i be the first index such that the i-th bit of $\tilde{p}$ is 0 with probability in $[2\delta, 1 - 2\delta]$. As the Marriott-Watrous amplification outputs incorrectly with probability at most δ, it means that there are two valid outputs $\tilde{p}_1$ and $\tilde{p}_2$, both are $\varepsilon/4$-close to p, and they coincide in the first $i - 1$ bits but differs at the i-th bit. Let q_i be the value of q at that step, which consists of the first $i - 1$ bits of $\tilde{p}_1$ and $\tilde{p}_2$, then $|q_i + 2^{-i} - p| \le \varepsilon/4$. Therefore the remaining bits of q could only be $\overline{011\dots 11}$, $\overline{100\dots 00}$ or $\overline{100\dots 01}$, which means $|q_i + 2^{-i} - q| \le \varepsilon/4$ and thus $|q - p| \le \varepsilon/2$. Notice that on the i-th (and the last bit when $b_i = 1$) the error reduction may fail and arbitrarily output 0 or 1, but it does not matter as both 0 and 1 are viable in these cases.

As a conclusion, the value q is an ε-approximation of $|w^\dagger A^T v|^2$ with probability $1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$. The above circuit that outputs q is clearly with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$ as we use constructions in Theorem 12 and [5]. Finally, the the circuit is unitary since the $O(\log(1/\varepsilon))$ measurements that output b_i's can be deferred, and each W_i can be uncomputed by implementing the circuit in reverse. ◀

▶ **Corollary 15.** *Given $m = 2^S$, a contraction $A \in \mathbb{C}^{m\times m}$, a positive integer T, two unit vectors $v, w \in \mathbb{C}^m$ and an error parameter $\varepsilon > 0$, there is a unitary quantum circuit with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$ such that with probability $1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$, it outputs $w^\dagger A^T v$ with additive error ε.*

Proof. Let $A_1 = \begin{pmatrix} A & \\ & 1 \end{pmatrix}$, $v_1 = \begin{pmatrix} v/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}$, $v_1' = \begin{pmatrix} v/\sqrt{2} \\ \mathrm{i}/\sqrt{2} \end{pmatrix}$ and $w_1 = \begin{pmatrix} w/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}$. Since we have

$$w^\dagger A^T v = \frac{1}{2}\left(4|w_1^\dagger A_1^T v_1|^2 - |w^\dagger A^T v|^2 - 1\right) + \frac{\mathrm{i}}{2}\left(4|w_1^\dagger A_1^T v_1'|^2 - |w^\dagger A^T v|^2 - 1\right),$$

computing $|w^\dagger A^T v|^2$, $|w_1^\dagger A_1^T v_1|^2$ and $|w_1^\dagger A_1^T v_1'|^2$ each up to error $\varepsilon/2$ gives $w^\dagger A^T v$ with error ε. ◀

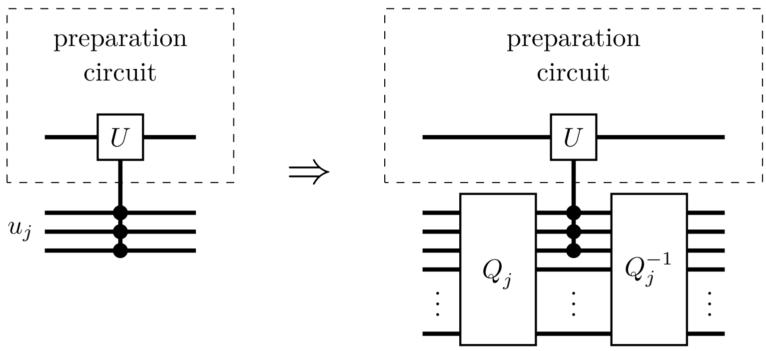

Figure 1 The quantum operator U in the preparation circuit controlled by an entry u_j of u, in binary representation with classical bits. We replace the classical control by first implementing the circuit Q_j, applying the controlled-U operator, and implementing Q_j in reverse.

Notice that one can instead achieve $1/\mathrm{poly}(mT/\varepsilon)$ error probability without using the exponential error reduction in [5], by simply repeating the decision circuit in $\mathsf{BQ_UL}$ for $O(\log(mT/\varepsilon))$ rounds. Nevertheless, it is enough for proving the following theorem, which states that unitary quantum circuits can simulate any unital quantum algorithm by computing its output distribution with arbitrarily small error.

▶ **Theorem 16.** *Given a unital quantum algorithm with time T and space $S = \log m$ specified by the natural representations $K(\Phi_1), \ldots, K(\Phi_T) \in \mathbb{C}^{m^2 \times m^2}$, where $\rho_T = \Phi_T \circ \Phi_{T-1} \circ \cdots \circ \Phi_1(|0^S\rangle\langle 0^S|)$ is its final state, a multi-outcome measurement $\{M_0, \ldots, M_{r-1}\}$ over the computational basis, and an error parameter $\varepsilon > 0$, there is a unitary quantum circuit W with time $\mathrm{poly}(mT/\varepsilon)$ and space $S' = O(\log(mT/\varepsilon))$ such that if $w \in \mathbb{C}^{2^{S'}}$ is the vector representation of $W|0^{S'}\rangle$ in computational basis, for every $j \in [r]$ it holds that $\big||w_j|^2 - \mathrm{Tr}[\rho_T M_j]\big| \le \varepsilon$.*

Proof. For every $j \in [r]$, let m_j be the dimension of the subspace that M_j projects onto. In other words, $m_j = \|\mathrm{vec}(M_j)\|_2^2$. As in the proof of Lemma 11, we can construct a contraction $A \in \mathbb{C}^{m^2T \times m^2T}$ and unit vectors $v, w \in \mathbb{C}^{m^2T}$ such that $w^\dagger A^T v = \mathrm{Tr}[\rho_T M_j]/\sqrt{m_j}$. By Corollary 15, for every $j \in [r]$ there is a unitary quantum circuit Q_j with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$ such that with probability $1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$, it gives an $(2m)^{-3}\varepsilon^2$-approximation of $\mathrm{Tr}[\rho_T M_j]/\sqrt{m_j}$, which implies an $(2m)^{-1}\varepsilon$-approximation of $\sqrt{\mathrm{Tr}[\rho_T M_j]}$.

Consider the preparation circuit constructed in Lemma 5 which prepares the unit vector

$$u = \left(\sqrt{\mathrm{Tr}[\rho_T M_0]}, \sqrt{\mathrm{Tr}[\rho_T M_1]}, \ldots, \sqrt{\mathrm{Tr}[\rho_T M_{r-1}]}\right).$$

with error $\varepsilon/2$. By construction, the preparation circuit can be viewed as a composition of $r-1$ unitary operators, each controlled by a different entry in u. Since u is not explicitly given, we instead control these unitary operators with the output qubits of Q_j, but without measurements. Each circuit Q_j is applied in reverse after the control, so that the space can be reused.

It is clear that the entire circuit is with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$. The error introduced by replacing each of th $r-1$ unitary operators is at most $(2m)^{-1}\varepsilon + 2^{-\mathrm{poly}(mT/\varepsilon)}$, therefore the total error is at most $\varepsilon/2 + (r-1)((2m)^{-1}\varepsilon + 2^{-\mathrm{poly}(mT/\varepsilon)}) < \varepsilon$. See Figure 1 for an illustration. ◀

The measurement $\{M_0, \dots, M_{r-1}\}$ in Theorem 16 could be over any subset of the qubits. In particular, when it is a two-outcome measurement over one qubit, we have the following direct corollary which improves Lemma 11:

▶ **Corollary 17.** *Given a unital quantum algorithm with time T and space $S = \log m$ specified by the natural representations $K(\Phi_1), \dots, K(\Phi_T) \in \mathbb{C}^{m^2 \times m^2}$, and an error parameter $\varepsilon > 0$, there is a unitary quantum circuit with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$, such that if the original unital circuit outputs 0 with probability p, then the unitary circuit outputs 0 with probability $p + \alpha$, where $|\alpha| \le \varepsilon$.*

6 Powering of Non-Contraction Matrices

In this section we extend the result of Corollary 15 to matrices that may not necessarily be contractions. We state the result for general square matrices, while the additive error can be exponentially large with respect to the spectral norm:

▶ **Theorem 18.** *Given $m = 2^S$, an arbitrary matrix $A \in \mathbb{C}^{m \times m}$, a positive integer T, two unit vectors $v, w \in \mathbb{C}^m$ and an error parameter $\varepsilon > 0$, there is a unitary quantum circuit W with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$ such that with probability $1 - 2^{-\mathrm{poly}(mT/\varepsilon)}$, it outputs $w^\dagger A^T v$ with additive error $\varepsilon \cdot \max(1, \|A\|^T)$.*

Proof. Ideally, we would like to apply the contraction powering algorithm on $A/\|A\|$ and multiply the result by $\|A\|^T$. However, the current best quantum algorithm for computing the spectral norm is [27, Theorem 5.2] which approximates $\|A\|$ with additive error ε_1 within time $\mathrm{poly}(m/\varepsilon_1)$ and space $O(\log(m/\varepsilon_1))$ and only works for contractions A. We use this algorithm to approximate $\|A\|$ for arbitrary A with multiplicative error as follows[1]: First compute $\|A\|_{\mathrm{F}}$ in $O(\log m + \log \|A\|_F) = O(\log(m\|A\|_2))$ space. Notice that $A/\|A\|_{\mathrm{F}}$ is a contraction since $\|A\|_{\mathrm{F}} \ge \|A\|$. Therefore, let σ be the approximation of $\|A/\|A\|_{\mathrm{F}}\|$ with additive error ε_1 by [27], then $\sigma\|A\|_F$ approximates $\|A\|$ since

$$\Big|\sigma\|A\|_F - \|A\|\Big| = \|A\|_F \cdot \Big|\sigma - \big\|A/\|A\|_F\big\|\Big| \le \sqrt{m}\varepsilon_1\|A\|.$$

Let $\varepsilon_1 = (3T\sqrt{m})^{-1}$, and let $\alpha = (1 - \sqrt{m}\varepsilon_1)^{-1}\sigma\|A\|_F$. Then

$$\|A\| \le \alpha \le \frac{1 + \sqrt{m}\varepsilon_1}{1 - \sqrt{m}\varepsilon_1}\|A\| \le (1 + T^{-1})\|A\|.$$

Now let $\widetilde{A} = \alpha^{-1}A$ so that $\widetilde{A}$ is always a contraction. Applying the contraction powering algorithm in Corollary 15 on $\widetilde{A}$ with error $\varepsilon/3$ results in a unitary quantum circuit with time $\mathrm{poly}(mT/\varepsilon)$ and space $O(\log(mT/\varepsilon))$ which outputs $w^\dagger \widetilde{A}^T v$ with additive error $\varepsilon/3$. Multiplying it by α^T gives the desired result, while the error is at most $\alpha^T\varepsilon/3 \le \varepsilon \cdot \|A\|^T$. ◀

7 Classical Simulation of Quantum Learning

7.1 Equivalence of Classical Simulation in Decision and Learning

▶ **Theorem 19.** *If there are functions $t(\cdot,\cdot)$ and $s(\cdot,\cdot)$, such that every unitary quantum learning algorithm with time T and space S can be simulated classically with time $t(T, S)$ and space $s(T, S)$, then*

$$\mathsf{promiseBQ_UL} \subseteq \mathsf{promiseBPTISP}(t(\mathrm{poly}(n), O(\log n)), s(\mathrm{poly}(n), O(\log n))).$$

[1] During the analysis we assume without loss of generality that $\|A\| \ge 1$, since otherwise it can always be relaxed to 1 whenever necessary.

Specifically, if every unitary quantum learning algorithm in time T and space S can be simulated classically with time $\mathrm{poly}(2^S T)$ *and space* $O(S + \log T)$*, then* $\mathsf{promiseBQ_UL} = \mathsf{promiseBPL}$*.*

Proof. Suppose that we have a unitary quantum circuit with time $T(n) = \mathrm{poly}(n)$ and space $S(n) = O(\log n)$ that decides a partial function $f : X \to \{0,1\}$, where $X \subseteq \{0,1\}^n$. Let $\Phi_{\mathrm{D}}(x,i)$ be the unitary gate at the i-th step of the decision algorithm with input x, which can be constructed in time $\mathrm{poly}(n)$ and space $O(\log n)$.

We can convert the quantum circuit to a learning algorithm as follows. Use X directly as the sample space, while the samples are always constant x for some fixed $x \in X$. The learning task is to distinguish between $x \in f^{-1}(0)$ or $x \in f^{-1}(1)$. Upon receiving the sample x, the learning algorithm simply applies the following unitary operator on $\mathbb{C}^{2^{S(n)}} \otimes \mathbb{C}^{T(n)}$:

$$|\psi\rangle|i\rangle \to \big(\Phi_{\mathrm{D}}(x,i)|\psi\rangle\big)|(i+1) \bmod T(n)\rangle$$

so that after $T(n)$ steps it computes in the first register the same state as in the quantum circuit. Therefore it computes $f(x)$ and distinguishes between the two cases. Using the premises, we have a classical learning algorithm with time $t(\mathrm{poly}(n), O(\log n))$ and space $s(\mathrm{poly}(n), O(\log n))$ that accomplishes the same task. The classical learning algorithm can be viewed as a randomized decision algorithm that computes $f(x)$ by self-constructing the stochastic matrices in the same time and space. ◀

▶ **Theorem 20.** *If* CONTRACTIONPOWERING $\in \mathsf{promiseBPTISP}(t(n), s(n))$*, where* $t(n) \geq \Omega(n)$ *and* $s(n) \geq \Omega(\log n)$*, then every unital quantum learning algorithm with time T and space S can be simulated classically with time* $t(\mathrm{poly}(2^S T))$ *and space* $s(\mathrm{poly}(2^S T))$*.*

Proof. Suppose that we have a unital quantum learning algorithm with time T and space $S = \log m$ that distinguishes between two distribution families $\mathcal{X}$ and $\mathcal{Y}$. Let $\Phi_{\mathrm{L}}(z)$ be the unital channel applied when receiving the sample z. With the sample distribution D, let $A = \mathop{\mathbf{E}}_{z\sim D}[K(\Phi_{\mathrm{L}}(z))]$. We note that A is a contraction matrix of dimension $m^2 \times m^2$ as every $K(\Phi_{\mathrm{L}}(z))$ is a contraction. Similar to proof of Lemma 11, the probability of the learning algorithm outputting 0 is

$$\mathop{\mathbf{E}}_{z\sim D^T}\left[\mathrm{vec}(M_0)^\dagger K(\Phi_T)\cdots K(\Phi_1)\mathrm{vec}(\rho_0)\right] = \mathrm{vec}(M_0)^\dagger A^T \mathrm{vec}(\rho_0).$$

What's different from Lemma 11 is that here A is not explicitly given. Instead, by Lemma 1, each time an entry of A is requested, it takes $\mathrm{poly}(mT)$ samples z to approximate the entry to at most $O((m^{2.5}T)^{-1})$ error, so that the approximated matrix $\widetilde{A}$ differs from the actual matrix A by at most $\|\widetilde{A} - A\| \leq O((\sqrt{m}T)^{-1})$. By Lemma 3 it means that $\|\widetilde{A}^T - A^T\| \leq O(m^{-1/2})$. Therefore applying the contraction powering algorithm on $\widetilde{A}$ gives a classical learning algorithm that distinguishes $\mathcal{X}$ and $\mathcal{Y}$ in time $t(\mathrm{poly}(mT))$ and space $s(\mathrm{poly}(mT))$.

The above scheme has two problems. First, a fixed matrix $\widetilde{A}$ cannot be directly stored, and if every time the same entry is requested, the entry is approximated as the average of a different batch of samples, it may result in different requested values for the same entry (even though the difference is small with high probability), similar to the problem in Lemma 14. However, unlike the case in Lemma 14, here the classical contraction powering algorithm is not explicitly given, and may not be robust against changing inputs.

The solution to this problem is the *shift and truncate* method by Saks and Zhou[23], which has found numerous applications in space-bounded algorithms [27] and derandomizations [3, 11]. Concretely, let $P = t(\mathrm{poly}(mT))$ be the largest number of possible requests to entries of A in the contraction powering algorithm, and take a uniform random number $\zeta \in [8L]$. For simplicity let $L = 12\sqrt{2m}T$ and $N = 24m^{2.5}T$. When the entry A_{jk} is requested, the algorithm takes $t(\mathrm{poly}(mT))$ samples z_i and calculate the average value a of the (j,k)-entries of $K(\Phi_{\mathrm{L}}(z_i))$, so that $|a - A_{jk}| < \frac{1}{8NP}$ with probability at least $1 - 2^{-P}$. The value fed back for the request is

$$\widetilde{A}_{jk} = \frac{1}{N}\left\lfloor N \cdot \mathrm{Re}(a) + \frac{\zeta}{8P}\right\rfloor + \frac{\mathrm{i}}{N}\left\lfloor N \cdot \mathrm{Im}(a) + \frac{\zeta}{8P}\right\rfloor.$$

We claim that with high probability, this value coincides with the fixed value

$$\frac{1}{N}\left\lfloor N \cdot \mathrm{Re}(A_{jk}) + \frac{\zeta}{8P}\right\rfloor + \frac{\mathrm{i}}{N}\left\lfloor N \cdot \mathrm{Im}(A_{jk}) + \frac{\zeta}{8P}\right\rfloor.$$

For the real part, as $|N \cdot \mathrm{Re}(a) - N \cdot \mathrm{Re}(A_{jk})| < \frac{1}{8P}$, there is at most one possibility for ζ such that $\left\lfloor N \cdot \mathrm{Re}(a) + \frac{\zeta}{8P}\right\rfloor \neq \left\lfloor N \cdot \mathrm{Re}(A_{jk}) + \frac{\zeta}{8P}\right\rfloor$, which is of probability $\frac{1}{8P}$, and the same holds for the imaginary part. By the union bound on the bad events during all L requests, with probability

$$1 - \left(2^{-P} + \frac{1}{4P}\right)P \geq \frac{2}{3}$$

for every (j,k) the value $\widetilde{A}_{jk}$ are always the same, and $|\widetilde{A}_{jk} - A_{jk}| \leq \frac{\sqrt{2}}{N} = \frac{1}{m^2 L}$, so $\|\widetilde{A} - A\| \leq L^{-1}$.

The second problem is that because of the approximation error, $\widetilde{A}$ might not be a contraction matrix. This is easily fixed by using the matrix $\widetilde{A}' = \frac{L}{L+1} \cdot \widetilde{A}$ as the input. Since $\|\widetilde{A} - A\| \leq L^{-1}$ with probability $2/3$, it is implied that

$$\|\widetilde{A}'\| = \frac{L}{L+1} \cdot \|\widetilde{A}\| \leq \frac{L}{L+1} \cdot \left(1 + L^{-1}\right) = 1,$$

$$\|\widetilde{A}' - A\| \leq \|\widetilde{A} - A\| + \frac{1}{L+1}\|\widetilde{A}\| \leq \frac{2}{L}.$$

Since $\|\mathrm{vec}(M_0)\|_2 = \sqrt{m/2}$, $\|\mathrm{vec}(\rho_0)\|_2 = 1$, in this case we have (by Lemma 3)

$$\left|\mathrm{vec}(M_0)^\dagger(\widetilde{A}'^T - A^T)\mathrm{vec}(\rho_0)\right| \leq \frac{\sqrt{2m}T}{L} = \frac{1}{12}.$$

Since the error of the original quantum learning algorithm can be amplified to $1/4$ so that $\mathrm{vec}(M_0)^\dagger A^T \mathrm{vec}(\rho_0)$ is in $[0, 1/4]$ or $[3/4, 1]$, we conclude that with probability $5/6$,

$$\mathrm{vec}(M_0)^\dagger \widetilde{A}'^T \mathrm{vec}(\rho_0) \in [0, 1/3] \text{ or } [2/3, 1]$$

Therefore the two cases can be distinguished by the classical contraction powering algorithms on $\widetilde{A}'$, and it can be repeated for constant rounds so that the total error rate is brought down to $1/3$. ◀

▶ **Corollary 21.** *If* CONTRACTIONPOWERING $\in$ promiseBPL, *then every unital quantum learning algorithm with time T and space S can be simulated classically with time* $\mathrm{poly}(2^S T)$ *and space $O(S + \log T)$.*

Since by Theorem 10 we already know CONTRACTIONPOWERING $\in$ promiseBQ$_{\mathrm{U}}$L, combined with Theorem 19, we get the equivalence between efficient simulations of decision problems and learning problems:

▶ **Theorem 22.** *Every (unital) quantum learning algorithm with time T and space S can be simulated classically with time* $\mathrm{poly}(2^S T)$ *and space* $O(S + \log T)$*, if and only if* $\mathsf{promiseBQ_UL} = \mathsf{promiseBPL}$*.*

Also, as we already know $\mathsf{promiseBQ_UL} \subseteq \mathsf{promiseL}^2$ [28], we have the following unconditional result:

▶ **Corollary 23.** *Every unital quantum learning algorithm with time T and space S can be simulated classically with time* $2^{O(S^2+\log^2 T)}$ *and space* $O(S^2 + \log^2 T)$*.*

7.2 Classical Simulation when One Family is Singleton

▶ **Theorem 24.** *If $\mathcal{Y} = \{Y\}$, then any quantum learning algorithm that distinguishes between $\mathcal{X}$ and $\mathcal{Y}$ within time T and space S can be simulated classically in time* $\mathrm{poly}(2^S T)$ *and space* $O(S + \log T)$*.*

References

1 Paul Beame, Shayan Oveis Gharan, and Xin Yang. Time-space tradeoffs for learning finite functions from random evaluations, with applications to polynomials. In *Conference On Learning Theory (COLT 2018)*, pages 843–856, 2018.

2 Charles Bennett. Notes on landauer's principle, reversible computation, and maxwell's demon. *Studies In History and Philosophy of Science Part B: Studies In History and Philosophy of Modern Physics*, 34:501–510, September 2003. `doi:10.1016/S1355-2198(03)00039-X`.

3 Jin-Yi Cai, Venkatesan T Chakaravarthy, and Dieter van Melkebeek. Time-space tradeoff in derandomizing probabilistic logspace. *Theory of Computing Systems*, 39(1):189–208, 2006.

4 Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. The power of block-encoded matrix powers: Improved regression techniques via faster hamiltonian simulation. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.

5 Bill Fefferman, Hirotada Kobayashi, Cedric Yen-Yu Lin, Tomoyuki Morimae, and Harumichi Nishimura. Space-efficient error reduction for unitary quantum computations. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*, volume 55. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016. `doi:10.4230/LIPIcs.ICALP.2016.14`.

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

7 Bill Fefferman and Zachary Remscrim. Eliminating intermediate measurements in space-bounded quantum computation, 2020. `arXiv:2006.03530`.

8 Sumegha Garg, Ran Raz, and Avishay Tal. Extractor-based time-space lower bounds for learning. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC 2018)*, pages 990–1002, 2018.

9 András Gilyén. Quantum singular value transformation & its algorithmic applications. *ILLC Dissertation Series*, 2019.

10 András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 193–204, 2019.

11 Ofer Grossman and Yang P Liu. Reproducibility and pseudo-determinism in log-space. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 606–620. SIAM, 2019.

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

13 Gillat Kol, Ran Raz, and Avishay Tal. Time-space hardness of learning sparse parities. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC 2017)*, pages 1067–1080, 2017.

14 R. Landauer. Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5(3):183–191, 1961.

15 Klaus-Jörn Lange, Pierre McKenzie, and Alain Tapp. Reversible space equals deterministic space. *Journal of Computer and System Sciences*, 60(2):354–367, 2000.

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

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

18 Dana Moshkovitz and Michal Moshkovitz. Entropy samplers and strong generic lower bounds for space bounded learning. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

19 Michael A Nielsen and Isaac L Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2010.

20 David Perez-Garcia, Michael M Wolf, Denes Petz, and Mary Beth Ruskai. Contractivity of positive and trace-preserving maps under l p norms. *Journal of Mathematical Physics*, 47(8):083506, 2006.

21 Ran Raz. A time-space lower bound for a large class of learning problems. In *58th IEEE Annual Symposium on Foundations of Computer Science (FOCS 2017)*, pages 732–742, 2017.

22 Ran Raz. Fast learning requires good memory: A time-space lower bound for parity learning. *Journal of the ACM (JACM)*, 66(1):1–18, 2018.

23 Michael Saks and Shiyu Zhou. $\mathsf{BPHSPACE}(s) \subseteq \mathsf{DSPACE}(s^{3/2})$. *Journal of computer and system sciences*, 58(2):376–403, 1999.

24 Ohad Shamir. Fundamental limits of online and distributed algorithms for statistical learning and estimation. In *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, pages 163–171, 2014.

25 Yaoyun Shi. Both toffoli and controlled-not need little help to do universal quantum computing. *Quantum Info. Comput.*, 3(1):84–92, 2003.

26 Jacob Steinhardt, Gregory Valiant, and Stefan Wager. Memory, communication, and statistical queries. In *Conference on Learning Theory (COLT 2016)*, pages 1490–1516, 2016.

27 Amnon Ta-Shma. Inverting well conditioned matrices in quantum logspace. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing (STOC 2013)*, pages 881–890, 2013.

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

29 Fuzhen Zhang. *Matrix theory: basic results and techniques*. Springer Science & Business Media, 2011.