Electronic Colloquium on Computational Complexity, Report No. 11 (2024)

Quantum Time-Space Tradeoffs for Matrix Problems

Paul Beame*
Computer Science & Engineering
University of Washington

Niels Kornerup†
Computer Science
University of Texas at Austin

January 24, 2024

Abstract

We consider the time and space required for quantum computers to solve a wide variety of problems involving matrices, many of which have only been analyzed classically in prior work. Our main results show that for a range of linear algebra problems—including matrix-vector product, matrix inversion, matrix multiplication and powering—existing classical time-space tradeoffs, several of which are tight for every space bound, also apply to quantum algorithms with at most a constant factor loss. For example, for almost all fixed matrices A, including the discrete Fourier transform (DFT) matrix, we prove that quantum circuits with at most T input queries and S qubits of memory require $T = \Omega(n^2/S)$ to compute matrix-vector product Ax for $x \in \{0,1\}^n$. We similarly prove that matrix multiplication for $n \times n$ binary matrices requires $T = \Omega(n^3/\sqrt{S})$. Because many of our lower bounds are matched by deterministic algorithms with the same time and space complexity, our results show that quantum computers cannot provide any asymptotic advantage for these problems with any space bound.

We obtain matching lower bounds for the stronger notion of quantum cumulative memory complexity—the sum of the space per layer of a circuit.

We also consider Boolean (i.e. AND-OR) matrix multiplication and matrix-vector products, improving the previous quantum time-space tradeoff lower bounds for $n \times n$ Boolean matrix multiplication to $T = \Omega(n^{2.5}/S^{1/3})$ from $T = \Omega(n^{2.5}/S^{1/2})$.

Our improved lower bound for Boolean matrix multiplication is based on a new coloring argument that extracts more from the strong direct product theorem that was the basis for prior work. To obtain our tight lower bounds for linear algebra problems, we require much stronger bounds than strong direct product theorems. We obtain these bounds by adding a new bucketing method to the quantum recording-query technique of Zhandry that lets us apply classical arguments to upper bound the success probability of quantum circuits.

*Research supported by NSF grant CCF-2006359
†Research supported by David Soloveichik's Sloan Fellowship

ISSN 1433-8092

1 Introduction

Matrix computations are among the most fundamental computational problems and are critically important in areas such as numerical and scientific computing, optimization, and machine learning. If quantum computers can be shown to have a significant advantage over classical computations for these types of problems then it would open up a wide range of applications for such devices.

Prior work has shown that non-standard versions of matrix problems may indeed admit exponential or large polynomial quantum advantage: For any efficiently implementable operator M, the HHL algorithm of Harrow, Hassidim, and Lloyd [HHL09] (with the improvements of [CKS15]) can efficiently ϵ-approximate the value of $x^\dagger M x$ for the solution x of a well-conditioned linear system. However, it is worth noting that this algorithm requires the input to be presented in an unconventional format.

Many extensions of the HHL algorithm have also been proposed that can be elegantly described in the quantum singular value transform (qSVT) framework first described in [LC19] and popularized by [GSLW19]. Despite initial hope of exponential speed-up, a series of papers by Tang and co-authors, and others (e.g. [Tan18, CGL$^+$20a, CGL$^+$20b, GST22, BT23, CCH$^+$22]) has shown that, by providing classical algorithms a comparable input format to the HHL algorithm, these quantum algorithms can be replaced by classical ones with only a polynomial blowup in the running time, although this polynomial is not always small.

This body of work still begs the question: What is the conventional quantum complexity of standard classical problems like explicitly computing the linear-system solutions, multiplying or inverting matrices, computing matrix-vector products, and computing the low rank approximation of a matrix?

By the polynomial method, we know that computing a single inner product (or parity) of n-bit vectors requires $\Omega(n)$ quantum queries [BBC$^+$01] but linear algebra computations generally involve $\Omega(n)$ or $\Omega(n^2)$ such computations. Sherstov [She12], generalizing results of Klauck, Špalek, and de Wolf [KŠdW07] for the OR function, gave a strong direct product lower bound for quantum query complexity proved using the polynomial method, which proves strong lower bounds for inner products involving many *disjoint* input vectors. However, the matrix problems in linear algebra are very far from direct product problems: The vectors involved are highly correlated with each other, so this prior work does not shed light on the key question of whether quantum algorithms provide any advantage for general linear algebra.

In this paper, we resolve these questions for quantum computation of a wide array of linear algebra problems. We prove lower bounds for quantum computation that are asymptotically the same as the best classical lower bounds. Since many of the problems also have deterministic algorithms whose resource usage matches the lower bounds, our results show that there is provably no asymptotic quantum advantage at all in solving these linear algebra problems!

As with the study of classical computation involving super-linear time lower bounds, we consider quantum algorithms in which we limit the number of qubits of memory and hence produce quantum time-space tradeoffs. That is, for each fixed bound on the amount of memory allowed, we derive asymptotically the same time lower bound for the quantum algorithm as one would get for the time lower bound on classical algorithms with the same number of classical bits. In many ways, quantum memory is an even more important resource than classical memory since it is a measure of the maximum number of qubits that maintain coherence at any time during the algorithm's execution. For this reason the first general-purpose fault-tolerant quantum computers will likely have very limited memory and only be able to execute low depth quantum circuits. As such, it is crucial to consider both the time and space complexity for quantum algorithms.

We prove our lower bounds for quantum computation in a query model where algorithms are able to perform arbitrary input-independent unitary transformations on their state between quantum queries to their input. This is a sufficiently general model that our lower bounds also apply to any reasonable model of quantum computation—including quantum circuits where the (classical) input is stored in quantum-readable read only memory (QROM).

The keys to proving our time-space tradeoffs are new results proving much stronger lower bounds than strong direct product theorems for matrix-vector products and matrix multiplication. While our bounds have the same form as strong direct product theorems (the success probability decays exponentially with the number of outputs), they also apply with almost completely overlapping sets of inputs, in contrast to the disjoint inputs that are necessary to apply direct product theorems.

While there is a large body of work proving strong classical time-space tradeoffs (e.g. [Tom78, BFK+79, Yes84, BC82, Abr90, Abr91, Bea91, MNT93]) and a large body of work analyzing unrestricted quantum query algorithms versus their classical randomized counterparts (e.g [DJ92, BV97, Sim97, BBC+01, Amb02, ŠS05, Špa08, She11]), there are just a few previous papers that analyze the quantum memory required to make use of these quantum queries. Klauck, Špalek, and de Wolf [KŠdW07] extended the classical method of Borodin and Cook [BC82] for proving time-space trade-offs to quantum circuits using a new strong direct product theorem for quantum query algorithms computing the OR function. They showed that algorithms making T quantum queries and using S qubits of quantum memory require $T = \Theta(n^{1.5}/S^{1/2})$ to sort lists of length n, and require $T = \Omega(n^{2.5}/S^{1/2})$ to compute $n \times n$ Boolean matrix product. Ambainis, Špalek, and de Wolf [AŠdW09] extended this direct product approach to 2-sided error algorithms computing k-threshold functions which allowed them to produce similar trade-off lower bounds for systems of linear inequalities/equalities (though these have the drawback, unlike the other results, that the hard function for space S depends on the space bound). This approach, based on an extension of the adversary method using eigenspace analysis, was very difficult to apply.

As a result, further study of quantum time-space tradeoff lower bounds languished until it was enabled by an idea of Zhandry [Zha19] who, motivated by understanding quantum algorithms interacting with random function oracles, developed an approach to understanding quantum query algorithms using a *compressed oracle* and Fourier analysis. This views computations in a *recording query* basis that allows one to keep track of a quantum query algorithm as a superposition of basis states that have a natural classical query interpretation. It has been applied to finding multi-way collisions [LZ18] and to inverting a random permutation [Ros21]. This greatly simplifies the analysis of quantum query algorithms and can be applied to many lower bound methods that use randomly chosen inputs rather than being limited to cryptographic applications.

Extending Zhandry's approach, Hamoudi and Magniez [HM21] applied an even cleaner expression of the method, using phase oracles with the recording query basis rather than Fourier analysis, and extended it using biased random inputs to derive query lower bounds in a regime of exponentially small success probability. They used this to obtain time-space tradeoff lower bounds, proving that any quantum algorithm that finds K disjoint collisions in an input of length n with T quantum queries and S qubits of memory must have $T = \Omega(KN^{1/3}/S^{1/3})$. They also re-derived the earlier sorting lower bound using this method.

Our linear algebra lower bounds and methods Time-space trade-off lower bounds for linear algebraic problems were among the first to be studied for classical computation [Yes84] after the first bounds for sorting. The strongest classical results are due to Abrahamson [Abr91] who developed a powerful general method based on matrix rigidity. This yields state-of-the-art lower bounds for computation of Fourier transforms, convolution, matrix-vector products, matrix multiplication, matrix inversion, matrix powering, and linear system solving. The lack of any analogous results for quantum computation has been a substantial gap in our understanding [1].

Our results show that all of the linear algebraic time-space tradeoff lower bounds shown by Abrahamson [Abr91] also apply to quantum computation even when the quantum circuit can adaptively decide when to produce output based on the observed input. Since many of these classical lower bounds are tight, our results directly imply that there is no hybrid classical-quantum algorithms with a polynomial advantage for these problems unlike the query bounds for search and collision finding in [HLS22]. Using the generic results in [BK23], we also prove asymptotically equivalent lower bounds on the stronger notion of quantum cumulative memory complexity for these problems. We include a table of our time-space tradeoff lower bounds in Table 1.

As discussed already, we need a much stronger lower bound method than any derivable from strong direct product theorems. We do this by the adding new ideas to the compressed oracle/recording query approach of Zhandry [Zha19] as extended and applied by Magniez and Hamoudi [HM21]. Thus far, the compressed oracle method has used a two-step pattern: First, identify a notion of unusual progress of a quantum algorithm towards a solution (i.e., the partial information so far is more determinative of the answer than one might expect) and show that the total amplitude of states where this occurs is small, Second, show that the total amplitude of the quantum states where many outputs are produced without unusual progress can be bounded; this latter part has used ideas with classical analogues that can be applied by breaking the algorithm's final state into mutually orthogonal components, each with small amplitude on the correct answers.

However, in our case with linear algebra problems, there is no form of unusual progress and also no clear way to break up the problem into mutually orthogonal basis states. Thus, neither part of the pattern seems to work. Instead, we can use the recording query framework to characterize how much a quantum circuit can know about its input. We use the triangle inequality to bucket amplitude from the algorithm's state into a small number of non-orthogonal components (or buckets) that share some set of inputs that they know nothing about. We can then apply a classical argument showing that each component must have small amplitude on the correct answers. By finding a way to divide the state into a small number of buckets that each have small amplitude on correct answers, we can obtain tight lower bounds. The properties required of this division become more subtle as we move to the problem of matrix multiplication, where in order to get small amplitude, we need to contend with a partition featuring significantly more parts.

[1]Over a field of $> n$ elements one can reduce $n \times n$ Boolean matrix multiplication to ordinary multiplication of 0-1 matrices but the lower bound is inherently too weak because in the Boolean case each output bit is a disjointness function of its inputs and hence can be computed using only $O(\sqrt{n})$ quantum queries using Grover's algorithm ([Gro96]).

Problem	Quantum Lower Bound	Source
Matrix-Vector Product $f(x) = Ax$	$T = \Omega(n^2 \log d \ / S)$	Theorem 3.5
Discrete Fourier Transform $f(x) = Wx$	$T = \Omega(n^2 \log d \ / S)$	Corollary 3.6
Convolution $f(u, v) = u * v$	$T = \Omega(n^2 \log d \ / S)$	Corollary 3.8
Binary Integer Multiplication	$T = \Omega(n^2 / (S \log^2 n))$	Corollary 3.9
Matrix Triple Product $f(A, B, C) = ABC$	$T = \Omega(n^4 \log d \ / S)$	Corollary 3.12
Matrix Cubing $f(A) = A^3$	$T = \Omega(n^4 \log d \ / S)$	Corollary 3.13
Matrix Inversion $f(A) = A^{-1}$	$T = \Omega(n^4 \log d \ / S)$	Corollary 3.14
System of Linear Equations $f(A, y) = A^{-1}y$	$T = \Omega(n^3 \log d \ / S)$	Corollary 3.15
Matrix Multiplication $f(A, B) = AB$	$T = \Omega(n^3 \sqrt{\log d \ / S})$	Theorem 4.4
Matrix Squaring $f(A) = A^2$	$T = \Omega(n^3 \sqrt{\log d \ / S})$	Corollary 4.5
Boolean Matrix Multiplication $f(A, B) = A \bullet B$	$T = \Omega(n^2 / \sqrt{S})$	[KŠdW07]
	$T = \Omega(n^2 / S^{1/3})$	Theorem 5.5
Boolean Matrix Squaring $f(A) = A \bullet A$	$T = \Omega(n^2 / S^{1/3})$	Corollary 5.15

Table 1: Our quantum time space tradeoff lower bounds. Other than Boolean matrix multiplication, where [KŠdW07] shows a quantum advantage for the problem, all of these lower bounds match the tightest known classical lower bound. For the linear algebra problems, we assume that input elements come from some finite subset D of a field and let $d = |D|$.

Improved bounds for Boolean matrix operations Here we improve the previous lower bound for quantum algorithms computing Boolean matrix multiplication given in [KŠdW07] from $T = \Omega(n^{2.5}/S^{1/2})$ to $T = \Omega(n^{2.5}/S^{1/3})$. We do this using a more sophisticated embedding of the k-fold direct product of OR functions into an arbitrary subset of k outputs of Boolean matrix multiplication. The embedding hinges on the number of colors needed for a certain kind of coloring of subsets E of the $n \times n$ grid.

Our lower bounds also lead to improving the classical lower bound tradeoff of $T = \Omega(n^3/S)$ for circuits shown in [KŠdW07] to $T = \Omega(n^3/S^{2/3})$. (In these bounds, T is circuit depth and S is circuit width.) This is also larger than a classical lower bound of $T = \Omega(n^3/S)$ for $S \leq n^{0.5}$ for Boolean matrix multiplication on branching programs (a more general model than circuits) due to Abrahamson [Abr90] that is tight almost surely for input matrices whose entries are 1 with probability $1/\sqrt{n}$ independently.

Finally, we make a small adjustment to convert the Boolean matrix-vector lower bounds and lower bounds for systems of inequalities given in [KŠdW07] and [AŠdW09], respectively, so that the problems that are shown hard for space S do not depend on S.

2 Preliminaries

We start with some basic facts and definitions. We define the binary entropy function $H_2 : [0, 1] \to \mathbb{R}$, by $H_2(p) = -p \log_2 p - (1 - p) \log_2(1 - p)$.

Proposition 2.1 (Shannon)**.** *The number of subsets of* $[k]$ *of size at most* αk *is at most* $2^{H_2(\alpha)\, k}$*.*

Definition 2.2. An $m \times n$ matrix is (g, h, c)*-rigid* iff every $k \times w$ submatrix where $k \leq g$ and $w \geq n - h$ has rank at least ck. We call $(g, h, 1)$-rigid matrices (g, h)-rigid.

Matrix rigidity is a robust notion of rank and is an important property for proving time-space and cumulative complexity lower bounds for linear algebra. Fortunately, Yesha gives an explicit example of such a matrix and Abrahamson proved that there are many rigid square matrices.

Proposition 2.3 (Lemma 3.2 in [Yes84])**.** *The* $n \times n$ *Discrete Fourier Transform (DFT) matrix is* $(n/4, n/4, 1/2)$ *rigid.*

Proposition 2.4 (Lemma 4.3 in [Abr91])**.** *There is a constant* $\gamma \in (0, \frac{1}{2})$ *such that at least a* $1 - d^{-1}(2/3)^{\gamma n}$ *fraction of the matrices over* $D^{n\times n}$ *are* $(\gamma n, \gamma n)$*-rigid.*

2.1 Quantum circuit models

Throughout this paper, we consider quantum circuits that seek to compute target functions $f : D^n \to R^m$. Let $d = |D|$ and assume the existence of a bijective map $\nu : D \to \{0, \ldots, d-1\}$ that gives us an ordering on the elements of D.

Unitary quantum circuits A T query quantum circuit is specified using unitaries $U_0, \ldots, U_T$ that are independent of the input to the problem. These unitaries define a sequence of quantum states $|\psi_1^X\rangle_{\mathcal{C}}, \ldots |\psi_T^X\rangle_{\mathcal{C}}$ that the algorithm enters during its execution on input X. We think of each state $|\psi_t^X\rangle_{\mathcal{C}}$ as a linear combination of basis vectors $|i, p, w\rangle$ where $i \in [\lceil \log_2 n \rceil]$, $p \in [d]$, and $w \in \{0, 1\}^*$. With this decomposition we can define a query operator for input $X = x_1, \ldots, x_n$ as a unitary $\mathcal{O}_X$ that performs the following operation:

$$\mathcal{O}_X |i, p, w\rangle = \omega_d^{p\nu(x_i)} |i, p, w\rangle$$

Where ω_d is a d-th root of unity. Thus we can think of basis state $|i, p, w\rangle$ as being composed of the index, phase, and work registers respectively. The state $|\psi_t^X\rangle_{\mathcal{C}}$ of the circuit after t queries to the input X is given by:

$$|\psi_t^X\rangle_{\mathcal{C}} = U_t \mathcal{O}_X \ldots \mathcal{O}_X U_0 |0\rangle .$$

The output of the quantum circuit on input X is determined by taking $|\psi_T^X\rangle_{\mathcal{C}}$ and measuring the work register in the standard basis and then applying some input-independent post-processing function q to interpret the result as output $\tau \in R^J$ where $J \subseteq [m]$.

Oracle State Similar to [Amb02, Zha19, HM21], instead of hard-coding the input X into oracle $\mathcal{O}_X$, we define a general oracle operator $\mathcal{O}$ that interacts with input registers that start in state $|\psi_0\rangle_{\mathcal{O}}$. Given a distribution $\mathcal{D}$ over D^n, we can make $|\psi_0\rangle_{\mathcal{O}} = \sum_{X \in D^n} \sqrt{\Pr_{X' \sim \mathcal{D}}[X = X']}\, |X\rangle$ to represent an input sampled from $\mathcal{D}$. We define our oracle operator $\mathcal{O}$ as follows:

$$\mathcal{O} |i, p, w\rangle |X\rangle = (\mathcal{O}_X |i, p, w\rangle) |X\rangle$$

We can extend the unitaries $U_0, \ldots, U_T$ from our definition of unitary quantum circuits to act as the identity on the input registers. After doing so, the joint state of the input and quantum circuit at the end of the computation is given by:

$$|\psi_t\rangle = U_t \mathcal{O} \ldots \mathcal{O} U_0 |0\rangle_{\mathcal{C}} |\psi_0\rangle_{\mathcal{O}}$$

Again, the work register of $|\psi_T\rangle$ is measured and a post-processing function q is used to determine a partial assignment τ of outputs. The correctness of these outputs is then determined by measuring the input registers in the standard basis to obtain the input X and evaluating whether τ is consistent with $f(X)$ which we denote by writing $\tau || f(X)$. In general we can define the projector Π_k such that:

$$\Pi_k = \sum_{\substack{i,p,w,x_1,\ldots,x_n \\ \text{s.t. } q(w)||f(x_1,\ldots,x_n) \\ \text{and } |q(w)| \geq k}} |i, p, w, x_1, \ldots, x_n\rangle \langle i, p, w, x_1, \ldots, x_n|$$

The probability that the circuit produces a correct partial assignment of at least k outputs is given by $\|\Pi_k |\psi_T\rangle\|^2$. For a given partial assignment $q(w)$ to some outputs, we can define $\Pi_{q(w)}$ to be the projection onto the values of $|X\rangle$ where $q(w)||f(X)$. More specifically we have that:

$$\Pi_{q(w)} = \sum_{\substack{x_1,\ldots,x_n \\ \text{s.t. } q(w)||f(x_1,\ldots,x_n)}} |x_1, \ldots, x_n\rangle \langle x_1, \ldots, x_n| \tag{1}$$

By construction when q always produces a partial assignment of at least k elements we have:

$$\Pi_k = \sum_{i,p,w} |i, p, w\rangle \langle i, p, w| \otimes \Pi_{q(w)}$$

Space Bounded Quantum Computation Without loss of generality, we think of quantum circuits as starting in the all $|0\rangle$ state and cycling between applying input queries $\mathcal{O}$, arbitrary input-independent computation U_t, and intermediate measurements as in Figure 1. Adopting the notation of [BK23], we will consider the set of consecutive $\mathcal{O}$, U_t, and measurement gates as layer L_t. The space of layer L_t is the number of qubits that are passed from layer L_t to L_{t+1} and is denoted S_t. We define the space of a circuit as the maximum space of any layer, the time as the total number of layers, and the cumulative memory as the sum over all the S_t.

Intermediate measurements enable circuits to produce parts of their output early and discard unnecessary ancillary qubits. Some prior quantum time-space tradeoff lower bounds required the quantum circuit to declare which outputs are produced at each layer (e.g. sorting, Boolean matrix multiplication, and systems of linear inequalities [KŠdW07, AŠdW09]); however the recent collision-finding bounds in [HM21, HLS22] extend the output model for quantum circuits to include indicator qubits specifying which (if any) outputs are being produced at each layer. This allows them to prove lower bounds against quantum algorithms that dynamically decide when they want to produce outputs based on their observed inputs. While our Boolean matrix bounds build on those in [KŠdW07] and thus require a fixed output order, our linear algebra bounds work with this dynamic output model.

The time-space tradeoffs we prove in this paper will follow the Borodin-Cook method, and thus rely on dividing a quantum circuit into blocks that each are unlikely to produce many correct

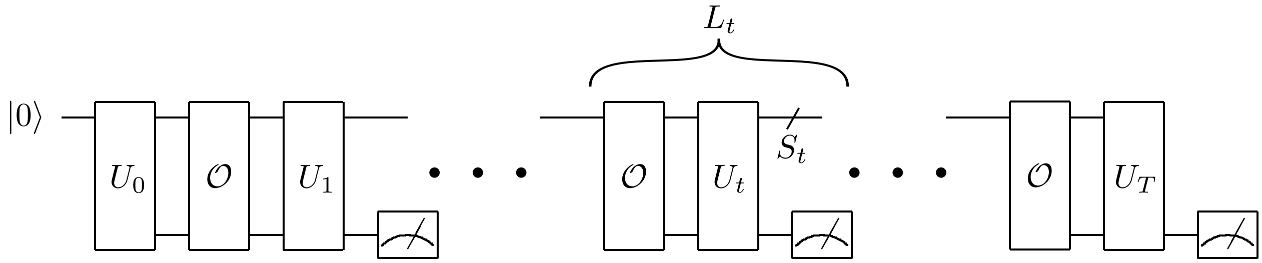

Figure 1: The general structure of a quantum circuit with T queries.

outputs. We use the unitary quantum circuits model to prove that these blocks cannot produce many outputs and then apply the results to our space bounded model using the differed measurement principle. After the first block, a quantum circuit will have some input-dependent state that can help it produce more outputs. Fortunately, a result by Aaronson lets us bound how much this initial state can amplify the success probability.

Proposition 2.5 ([Aar05]). *Let C be a quantum circuit, ρ be an S-qubit (possibly mixed) state, and π_{mix} be the S-qubit maximally mixed state. If C starting in initial state ρ produces some output z with probability p, then C starting in state π_{mix} will produce z with probability at least $p/2^{2S}$.*

We will implicitly use this proposition to limit the power of the initial quantum state in the following way: Let p be an upper bound on the success probability of a quantum circuit without any qubits of input-dependent initial state. Assume that there existed a quantum circuit with S bits of input dependent advice that could succeed with probability q. Then by Proposition 2.5 there is a quantum circuit without input-dependent initial state that succeeds with probability p that is at least $q/2^{2S}$. Thus we know that $q \le p2^{2S}$. Therefore any quantum circuit with S qubits of initial state can succeed with probability at most $p2^{2S}$.

2.2 The recording query technique and quantum lower bounds

Here we review the methods developed in [Zha19, HM21] that allow us to analyze what a quantum circuit learns about its input by making quantum queries. We will assume that the input state starts in the equal superposition state over all inputs, although [Zha19, HM21] generalize this method to other input distributions. We can exchange the general query operator $\mathcal{O}$ with a recording query operator $\mathcal{R}$ that we define as follows:

Definition 2.6 (adapted from [HM21]). Let $\mathcal{S}_1$ be the unitary operator that maps

$$\mathcal{S}_1 : \begin{cases} |\bot\rangle & \longrightarrow \frac{1}{\sqrt{d}} \sum_{y \in D} |y\rangle \\ \frac{1}{\sqrt{d}} \sum_{y \in D} |y\rangle & \longrightarrow |\bot\rangle \\ \frac{1}{\sqrt{d}} \sum_{y \in D} \omega_d^{p\,v(y)} |y\rangle & \longrightarrow \frac{1}{\sqrt{d}} \sum_{y \in D} \omega_d^{p\,v(y)} |y\rangle \;\; \forall p \in \{1, \ldots, d-1\}. \end{cases}$$

Let $\mathcal{S} = (I)_{i,p,w} \otimes (\mathcal{S}_1^{\otimes n})_{x_1,\ldots,x_n}$ and $\mathcal{O}$ be the standard oracle operator that maps the basis state

$$|i, p, w, x_1, \ldots, x_n\rangle \longrightarrow \omega_d^{p\,v(x_i)} |i, p, w, x_1, \ldots, x_n\rangle \,.$$

Then the *recording query oracle operator* $\mathcal{R}$ is defined as $\mathcal{SOS}$.

$\mathcal{S}_1$ introduces $\perp$ as a new value for the input registers. Intuitively, the $\perp$ symbol indicates that the algorithm does not know anything about that register of the oracle. Hence by adding and correctly manipulating the $\perp$ symbols in the oracle's registers, it is able to record what the algorithm knows about the input. Since $\mathcal{S}^2 = I$, we can exactly characterize how the states of quantum circuits with oracles $\mathcal{O}$ and $\mathcal{R}$ relate to one another.

Proposition 2.7 (Theorem 3.3 in [HM21])**.** *Let C be a quantum circuit that for each $j \le t$ applies unitary U_j after the j-th query. Let $\mathcal{S}$ be the unitary operation and $\mathcal{R}$ be the recording query oracle from Definition 2.6. Let*

$$|\psi_t\rangle = U_t \mathcal{O} U_{t-1} \dots U_1 \mathcal{O} U_0 \left(|0\rangle_{i,p,w} \otimes \frac{1}{d^{n/2}} \sum_{x_1,\dots,x_n \in D} |x_1,\dots,x_n\rangle_{x_1,\dots,x_n} \right)$$
$$|\phi_t\rangle = U_t \mathcal{R} U_{t-1} \dots U_1 \mathcal{R} U_0 \left(|0\rangle_{i,p,w} \otimes |\perp\rangle_{x_1,\dots,x_n} \right)$$

be the states of C with oracle $\mathcal{O}$ or $\mathcal{R}$ respectively. Then $|\psi_t\rangle = \mathcal{S}|\phi_t\rangle$.

In other words, it is impossible to distinguish the final state $|\psi_T\rangle$ of a circuit with standard oracle $\mathcal{O}$ from the output with recording oracle $\mathcal{R}$ if we apply $\mathcal{S}$ to the registers of $\mathcal{R}$ after the final query. Thus we can conclude that the success probability of a quantum circuit with T queries is given by $\|\Pi_{succ}|\psi_T\rangle\|^2 = \|\Pi_{succ}\mathcal{S}|\phi_T\rangle\|^2$. Note that while $|\phi_T\rangle$ may have inputs in the $\perp$ state, Proposition 2.7 tells us that $\mathcal{S}|\phi_T\rangle$ will never have an input in the $\perp$ state. This means that when considering recording query oracles, it is safe to keep our current definitions of Π_{succ} and $\Pi_{q(w)}$ which will always project out any basis state where an input is assigned to $\perp$. We will leverage the following property of $|\phi_T\rangle$ to bound the success probability of quantum circuits with at most T queries.

Proposition 2.8 (Fact 3.2 in [HM21])**.** *The state $|\phi_t\rangle$ from Proposition 2.7 is a linear combination of basis states $|i,p,w,x_1,\dots,x_n\rangle$ where at most t of the x_i are different from $\perp$.*

For the bounds in [HM21] it is essential to bound how the state of $|\phi\rangle_{\mathcal{O}}$ can change after each query. For our use of the recording query technique, this detailed analysis is not necessary. Nevertheless, we state the following proposition here for completeness.

Proposition 2.9 (Lemma 4.2 in [HM21] fixed)**.** *Let $d = |D|$. If the recording query operator $\mathcal{R}$ is applied to a basis state $|i,p,w,x_1,\dots,x_n\rangle$ where $p \neq 0$ then the register $|x_i\rangle_{\mathcal{X}}$ is mapped to*

$$\begin{cases} \sum_{y\in D} \frac{\omega_d^{py}}{\sqrt{d}} |y\rangle & \text{if } x_i = \perp \\ (1-\frac{2}{d})\omega_d^{px_i}|x_i\rangle + \frac{1}{d}|x_i\rangle + \frac{\omega_d^{px_i}}{\sqrt{d}}|\perp\rangle + \sum_{y\in D\setminus\{x_i\}} \frac{1-\omega_d^{py}-\omega_d^{px_i}}{d}|y\rangle & \text{otherwise.} \end{cases} \tag{2}$$

If $p = 0$ then none of the registers is changed.

3 Quantum matrix vector products

In this section, we consider the task of—for a fixed matrix $A \in \mathbb{F}^{m\times n}$—computing the function $f(x) = Ax$ for inputs $x \in D^m$ using a quantum circuit. We note that this is a fundamentally harder task than is considered in many quantum machine learning papers (for example [HHL09]) as we require the circuit to output a classical vector $y \in \mathbb{F}^n$ rather than either a quantum state encoding the entries of y in the amplitudes or an estimate of $y^\dagger My$. Also unlike many prior quantum time-space tradeoffs, including sorting [KŠdW07, HM21, BK23] and boolean matrix multiplication [KŠdW07] (and our Theorem 5.5), our matrix vector product and matrix multiplication lower bounds apply to circuits that can adaptively decide when to produce each output based on the observed inputs. Time-space lower bounds against such quantum circuits were first described in [HM21] for the multiple disjoint collisions problem, although they were not able to show such a result for sorting. Similar to [HM21] we are able to lower bound these circuits by identifying a single hard distribution over the inputs that applies to any set of outputs.

3.1 Success probability of small depth quantum circuits

We prove the following key lemma, which lets us bound the number of correct outputs produced by a shallow quantum circuit.

Lemma 3.1. *Let A be any (k, h, c)-rigid $m \times n$ matrix over a finite field $\mathbb{F}$ and let $f : D^n \to \mathbb{F}^m$ for $D \subseteq \mathbb{F}$ be defined by $f(x) = Ax$. Then for $\alpha > 0$ and for input x sampled uniformly from D^n and any quantum circuit $\mathcal{C}$ with at most αh queries to x, the probability that $\mathcal{C}$ produces k correct output values of $f(x)$ is at most $\lceil h/(ck) \rceil \, (2^{H_2(\alpha)}/|D|^{1-\alpha})^{ck}$.*

Note: For $\alpha \leq 0.1717$ we have $1 - \alpha - H_2(\alpha) > 1/6$ and hence the bound is at most $\lceil h/(ck) \rceil |D|^{-ck/6}$ for $d \geq 2$.

Proof. Let $d = |D|$. For simplicity we will assume that $q(w)$—the output as a function of the measured value of the work register—always produces k outputs.[2] Let A be a (k, h, c)-rigid matrix. By Proposition 2.8 after $t \leq \alpha h$ queries in the recording query oracle model, we can write the state as:

$$|\phi_t\rangle = \sum_{\substack{i,p,w \\ I \subseteq [n],\ |I| \leq t \\ y \in D^I}} \alpha_{i,p,w,I,y} |i, p, w\rangle |y\rangle_I |\perp\rangle_{[n]\setminus I} \tag{3}$$

for some $\alpha_{i,p,w,I,y}$ with $\sum_{i,p,w,I,y} |\alpha_{i,p,w,I,y}|^2 = 1$. Thus by Proposition 2.7, the final state of the algorithm in the non-recording query oracle setting is given by:

$$|\psi_t\rangle = \mathcal{S} |\phi_t\rangle = \mathcal{S} \sum_{\substack{i,p,w \\ I \subseteq [n],\ |I| \leq t \\ y \in D^I}} \alpha_{i,p,w,I,y} |i, p, w\rangle |y\rangle_I |\perp\rangle_{[n]\setminus I}$$

Since $\mathcal{S}$ behaves as the identity on $|\psi\rangle_{\mathcal{C}}$ and the $|i, p, w\rangle$ are orthogonal basis states, we can rewrite this as:

$$\sum_{i,p,w} \beta_{i,p,w} |i, p, w\rangle \otimes \left[\mathcal{S}_1^{\otimes n} \sum_{\substack{I \subseteq [n],\ |I| \leq t \\ y \in D^I}} \beta_{I,y}^{i,p,w} |y\rangle_I |\perp\rangle_{[n]\setminus I} \right]$$

[2] If in general $q(w)$ produces more than k outputs, we only consider its first k outputs.

for some $\beta_{i,p,w}$ and $\beta^{i,p,w}_{I,y}$ such that $\alpha_{i,p,w,I,y} = \beta_{i,p,w}\, \beta^{i,p,w}_{I,y}$, $\sum_{i,p,w} |\beta_{i,p,w}|^2 = 1$ and for each i, p, w, $\sum_{I,y} |\beta^{i,p,w}_{I,y}|^2 = 1$. With this decomposition, the success probability is given by:

$$\begin{aligned}
\left\| \Pi_k \mathcal{S} \left| \phi_t \right\rangle \right\|^2 &= \left\| \Pi_k \sum_{i,p,w} \beta_{i,p,w} \left| i, p, w \right\rangle \otimes \left[\mathcal{S}_1^{\otimes n} \sum_{\substack{I \subseteq [n],\ |I| \le t \\ y \in D^I}} \beta^{i,p,w}_{I,y} \left| y \right\rangle_I \left| \perp \right\rangle_{[n]\setminus I} \right] \right\|^2 \\
&= \left\| \sum_{i,p,w} \beta_{i,p,w} \left| i, p, w \right\rangle \otimes \left[\Pi_{q(w)} \mathcal{S}_1^{\otimes n} \sum_{\substack{I \subseteq [n],\ |I| \le t \\ y \in D^I}} \beta^{i,p,w}_{I,y} \left| y \right\rangle_I \left| \perp \right\rangle_{[n]\setminus I} \right] \right\|^2
\end{aligned}$$

where $\Pi_{q(w)}$ is defined as in Equation (1) and is the projection of Π_k onto fixed values of $q(w)$. Since the basis states $|i, p, w\rangle$ are orthogonal and $\sum_{i,p,w} |\beta_{i,p,w}|^2 = 1$, we have

$$\left\| \Pi_k \mathcal{S} \left| \phi_t \right\rangle \right\|^2 \le \max_{i,p,w} \left\| \Pi_{q(w)} \mathcal{S}_1^{\otimes n} \sum_{\substack{I \subseteq [n],\ |I| \le t \\ y \in D^I}} \beta^{i,p,w}_{I,y} \left| y \right\rangle_I \left| \perp \right\rangle_{[n]\setminus I} \right\|^2 \tag{4}$$

We now fix i, p, w and let $A_{q(w)}$ be the submatrix of A restricted to the rows defined by the set of the k output values U associated with $q(w)$. We can describe $\Pi_{q(w)}$ as a projection onto basis states $|x_1, \ldots, x_n\rangle$ such that:

$$A_{q(w)} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = q(w).$$

Since the basis states $\left| y \right\rangle_I \left| \perp \right\rangle_{[n]\setminus I}$ for distinct I are orthogonal in the recording query basis, they remain orthogonal in the standard basis after the $\mathcal{S}$ operator is applied. However, the subsequent application of the $\Pi_{q(w)}$ projector makes these vectors no longer orthogonal.

To handle this, we bucket the sets $I \subseteq [n]$ with $|I| \le t$ into a small number of buckets, $\mathcal{B}_1, \ldots$, so that for each bucket $\mathcal{B}_\ell$ we can bound:

$$\mu_\ell = \left\| \Pi_{q(w)} \mathcal{S}_1^{\otimes n} \sum_{I \in \mathcal{B}_\ell, y \in D^I} \beta^{i,p,w}_{I,y} \left| y \right\rangle_I \left| \perp \right\rangle_{[n]\setminus I} \right\|^2$$

and then we can use Cauchy-Schwarz to bound the success probability as a sum of the μ_ℓ.

In particular, our key observation is that if a bucket of recording query basis states completely misses querying a fixed set of input variables that could completely scramble the value of a set of r output values, then one cannot do better than randomly guess those output values. More precisely, we show that the contribution to success from that bucket of basis states has amplitude at most $\frac{1}{\sqrt{d^r}}$.

Lemma 3.2. *Let $U \subseteq [m]$ be a set of output indices and $V \subseteq [n]$ be a set of input indices with $|V| = |U| = r$ such that the submatrix $A_{U,V}$ is full rank. Fix $q \in \mathbb{F}^U$ and define Π_q to be the projection map onto the span of the set of basis states $|x_1, \ldots, x_n\rangle$ with $x_1 \ldots x_n \in D^n$ such that $A_U x = q$. Then for any collection $\mathcal{B}$ of sets $I \subseteq [n] \setminus V$ and any quantum state $\sum_{I \in \mathcal{B},\ y \in D^I} \gamma_{I,y} \left| y \right\rangle_I \left| \perp \right\rangle_{[n]\setminus I}$ we have*

$$\left\| \Pi_q \mathcal{S}_1^{\otimes n} \sum_{I \in \mathcal{B},\ y \in D^I} \gamma_{I,y} \left| y \right\rangle_I \left| \perp \right\rangle_{[n]\setminus I} \right\|^2 \le \frac{1}{d^r}.$$

Proof. By definition each $I \in \mathcal{B}$ satisfies $I \cap V = \varnothing$, so

$$\begin{aligned}
&\Pi_q \mathcal{S}_1^{\otimes n} \sum_{I \in \mathcal{B},\, y \in D^I} \gamma_{I,y} \left|y\right\rangle_I \left|\perp\right\rangle_{[n]\setminus I} \\
&= \Pi_q \mathcal{S}_1^{\otimes n} (\left|\perp\right\rangle_V \otimes \sum_{I \in \mathcal{B},\, y \in D^I} \gamma_{I,y} \left|y\right\rangle_I \left|\perp\right\rangle_{[n]\setminus (I \cup V)}) \\
&= \Pi_q (\mathcal{S}_1^{\otimes j} \left|\perp\right\rangle_V \otimes \mathcal{S}_1^{\otimes (n-j)} \sum_{I \in \mathcal{B},\, y \in D^I} \gamma_{I,y} \left|y\right\rangle_I \left|\perp\right\rangle_{[n]\setminus (I \cup V)}) \\
&= \Pi_q (\sum_{y' \in D^V} \frac{1}{\sqrt{d^r}} \left|y'\right\rangle_V \otimes \mathcal{S}_1^{\otimes (n-j)} \sum_{I \in \mathcal{B},\, y \in D^I} \gamma_{I,y} \left|y\right\rangle_I \left|\perp\right\rangle_{[n]\setminus (I \cup V)})
\end{aligned}$$

since $\mathcal{S}_1(\left|\perp\right\rangle) = \sum_{y' \in D} \frac{1}{\sqrt{d}} \left|y'\right\rangle$. Now

$$\mathcal{S}_1^{\otimes (n-j)} \sum_{I \in \mathcal{B},\, y \in D^I} \gamma_{I,y} \left|y\right\rangle_I \left|\perp\right\rangle_{[n]\setminus (I \cup V)} = \sum_{z \in (D \cup \{\perp\})^{[n]\setminus V}} \delta_z \left|z\right\rangle_{n \setminus V}$$

for some δ_z amplitudes satisfying $\sum_{z \in (D \cup \{\perp\})^{[n]\setminus V}} |\delta_z|^2 = 1$.

For each value of $z \in D^{[n]\setminus V}$, since the sub-matrix $A_{U,V}$ is invertible, there is a unique value $y_z \in D^V$ such that $A_U(y_z \cup z) = q$ so we get that

$$\begin{aligned}
&\left\| \Pi_q \mathcal{S}_1^{\otimes n} \sum_{I \in \mathcal{B},\, y \in D^I} \gamma_{I,y} \left|y\right\rangle_I \left|\perp\right\rangle_{[n]\setminus I} \right\|^2 \\
&= \left\| \Pi_q \left[\sum_{y' \in D^V} \frac{1}{\sqrt{d^r}} \left|y'\right\rangle_V \otimes \sum_{z \in (D \cup \{\perp\})^{n-j}} \delta_z \left|z\right\rangle_{[n]\setminus V} \right] \right\|^2 \\
&= \left\| \frac{1}{\sqrt{d^r}} \cdot \Pi_q \left[\sum_{y' \in D^V} \left|y'\right\rangle_V \sum_{z \in D^{n-j}} \delta_z \left|z\right\rangle_{n \setminus V} \right] \right\|^2 \\
&= \left\| \frac{1}{\sqrt{d^r}} \cdot \Pi_q \sum_{z \in D^{[n]\setminus V}} \delta_z \sum_{y' \in D^V} \left|y'\right\rangle_V \left|z\right\rangle_{n \setminus V}) \right\|^2 \\
&= \left\| \frac{1}{\sqrt{d^r}} \sum_{z \in D^{[n]\setminus V}} \delta_z \left|y_z\right\rangle_V \left|z\right\rangle_{n \setminus V}) \right\|^2 \\
&\le \frac{1}{d^r}
\end{aligned}$$

since $\sum_{z \in D^{[n]\setminus V}} |\delta_z|^2 \le 1$. $\square$

Next we decompose the set of all I with $|I| \le t$ into buckets so that we can apply the above.

Lemma 3.3. *Let A be a (k, h, c)-rigid matrix and let $k' = \lceil ck \rceil$. Then for every subset U of k rows of A, there is a collection of disjoint k'-subsets of columns from $[n]$, $V_1, \ldots, V_\ell$ for $\ell = \lceil h/k' \rceil \le \lceil h/(ck) \rceil$ and corresponding sets of rows $U_1, \ldots, U_\ell \subseteq U$ such that for each $j \in [\ell]$, the $k' \times k'$ submatrix A_{U_j,V_j} is full rank. (In particular the union, W, of the sets V_j has size at least h.) If $c = 1$ then all $U_j = U$.*

Proof. Fix $U \in [m]$ with $|U| = k$. The following procedure constructs such a collection, one set at a time. We maintain a subset of W columns that is the union of the V_j constructed so far. Suppose that $|W| < h$. Then, by the (k, h, c)-rigidity of A, the submatrix $A_{U,[n]\setminus W}$ has rank at least k'. Hence there is a $k' \times k'$ submatrix A_{U_j,V_j} of $A_{U,[n]\setminus W}$ that has full rank k'. We now add V_j to the collection of k'-sets of columns, record its corresponding row set U_j, and set $W \leftarrow W \cup V_j$. This produces exactly $\lceil h/k' \rceil$ subsets. □

Fix the collection of sets $V_1, \ldots, V_\ell$ given by Lemma 3.3. Let $k'' = \lfloor \alpha k' \rfloor$. Suppose that $V_j = \{i_1, \ldots, i_{k'}\} \subseteq [n]$ with $i_1 \leq \cdots \leq i_{k'}$. For each $\lambda \in \binom{[k']}{k''}$, define the set V_j^λ to be the subset of V_j that has the k'' elements of V_j indexed by λ removed. (That is, $i_{j'} \notin V_j^\lambda$ iff $j' \in \lambda$.) Then $|V_j^\lambda| = k' - k'' \geq c(1-\alpha)k$. There are a total of $\binom{k'}{k''} \leq 2^{H_2(\alpha)\, k'}$ possible values of λ and hence $\lceil h/k' \rceil \cdot 2^{H_2(\alpha)\, k'}$ sets of the form V_j^λ. These sets have two useful properties: first any subset of $[n]$ with size at most αh must miss some V_j^λ and second if the entries of x corresponding to some V_j^λ are uniformly random, then for any set of k indices in Ax, at least $c(1-\alpha)k$ of these values are also uniformly random.

Lemma 3.4. *For $t \leq \alpha h$ and every $I \subseteq [n]$ with $|I| \leq t$, there is some $j \leq \lceil h/k' \rceil$ and $\lambda \in \binom{[k']}{k''}$ such that $I \subseteq [n] \setminus V_j^\lambda$.*

Proof. Fix such a set I with $|I| \leq t$. Since $t \leq \alpha h$, $|\bigcup_{j\in[\ell]} V_j| \geq h$, and the sets V_j are disjoint, by averaging there is some set V_j that has at most an α fraction of its elements in I. Hence V_j has at most $k'' \leq \alpha k'$ elements of I. Choose a set $\lambda \in \binom{[k']}{k''}$ that contains the indices within V_j of all of the elements of $V_j \cap I$. Then by construction $I \cap V_j^\lambda = \varnothing$. □

By applying Lemma 3.4 we can associate each $I \subseteq [n]$ with $|I| \leq t$ with a pair (j, λ) such that $I \in [n] \setminus V_j^\lambda$ and define bucket $\mathcal{B}_j^\lambda$ to consist of all such sets I associated with pair (j, λ).[3] Further, define a set $U_j^\lambda \subseteq U_j \subseteq [m]$ of the rows of $A_{q(w)}$ with $|U_j^\lambda| = k' - k''$ such that the submatrix $A_{U_j^\lambda, V_j^\lambda}$ is full rank. Such a subset of rows must exist since A_{U_j, V_j^λ} is a full rank matrix. Then let $q_j^\lambda = q(w)|_{U_j^\lambda}$ be the portion of the assignment $q(w)$ on the rows of U_j^λ.

We are now ready to provide an upper bound on the success probability from Equation (4).

$$
\begin{aligned}
&\left\| \Pi_{q(w)} \mathcal{S}_1^{\otimes n} \sum_{\substack{I \subseteq [n],\, |I| \leq t \\ y \in D^I}} \beta_{I,y}^{i,p,w} \, |y\rangle_I \, |\perp\rangle_{[n]\setminus I} \right\|^2 \\
&= \left\| \Pi_{q(w)} \mathcal{S}_1^{\otimes n} \sum_{j\in[\ell]} \sum_{\lambda \in \binom{[k']}{k''}} \sum_{I \in \mathcal{B}_j^\lambda,\, y \in D^I} \beta_{I,y}^{i,p,w} \, |y\rangle_I \, |\perp\rangle_{[n]\setminus I} \right\|^2 \\
&\leq \left\| \sum_{j\in[\ell]} \sum_{\lambda \in \binom{[k']}{k''}} \Pi_{q_j^\lambda} \, \mathcal{S}_1^{\otimes n} \sum_{I \in \mathcal{B}_j^\lambda,\, y \in D^I} \beta_{I,y}^{i,p,w} \, |y\rangle_I \, |\perp\rangle_{[n]\setminus I} \right\|^2 .
\end{aligned} \tag{5}
$$

[3] Note that while some sets I could be associated with multiple pairs (j, λ), we will pick only one such pair for each I.

Applying Lemma 3.2 with $r = k' - k''$, $q = q_j^\lambda$, $U = U_j^\lambda$, $V = V_j^\lambda$, and $\mathcal{B} = \mathcal{B}_j^\lambda$, we have that

$$\left\| \Pi_{q_j^\lambda}\, \mathcal{S}_1^{\otimes n} \sum_{I \in \mathcal{B}_j^\lambda,\, y \in D^I} \beta_{I,y}^{i,p,w} \left|y\right\rangle_I \left|\perp\right\rangle_{[n]\setminus I} \right\|^2 \leq 1/d^{k'-k''} \leq 1/d^{(1-\alpha)\,k'}.$$

and hence using Equation (5) we obtain that

$$\left\|\Pi_k \mathcal{S} \left|\phi_t\right\rangle\right\|^2 \leq \ell \binom{k'}{k''} / d^{(1-\alpha)\,k'} \leq \lceil h/k' \rceil\, 2^{H_2(\alpha)\,k'} / d^{(1-\alpha)\,k'} = \lceil h/k' \rceil\, (2^{H_2(\alpha)}/d^{(1-\alpha)})^{k'}.$$

Without loss of generality in our desired bound we can assume that $2^{H_2(\alpha)}/d^{(1-\alpha)} < 1$. Therefore the bound still applies when we replace k' by the potentially smaller ck which is what we needed to show. □

3.2 Matrix-vector product time-space tradeoffs and related lower bounds

Theorem 3.5. *Let m be $n^{O(1)}$. Let A be an $m \times n$ matrix over a field $\mathbb{F}$ that is $(g(m), h(n), c)$-rigid for $c \in (0, 1/2]$. Then any quantum circuit using time T and space S that computes a function $f : D^n \to \mathbb{F}^m$ for $D \subseteq \mathbb{F}$ with $d = |D|$ given by $f(x) = Ax$ with success probability larger than 2^{-S} requires that T is $\Omega(g(m)\, h(n) \log(d)\ /S)$; more precisely, T must be $\Omega(\min\{g(m)\, n \log d,\ m\, h(n) \log d\}/S)$.*

Proof. First observe that since $S \geq \log_2 n$ and $T \geq n$ we know that $T \cdot S$ is $\Omega(n \log n)$ which is $\Omega(g(m)\, n \log |D|)$ if $g(m) < (12/c) \log_d n$. Therefore we can assume without loss of generality that $g(m) \geq (12/c) \log_d n$.

Let $\mathcal{C}$ be a quantum circuit with T queries and space S, write $h = h(n)$, $g = g(m)$, and let $\alpha = 0.1717$. We partition $\mathcal{C}$ into $\lceil T/(\alpha h) \rceil$ sub-circuits that each have at most αh queries. By combining Proposition 2.5 and Lemma 3.1, we know that each sub-circuit can produce $k \leq g$ correct outputs with probability at most $2^{2S} \lceil h/(ck) \rceil\, d^{-ck/6} \leq h\, 2^{2S} d^{-ck/6}$.

Now suppose that $h\, 2^{2S} d^{-cg/6} > 2^{-S}/T$. Then $T 2^{3S} > d^{cg/6}/h \geq d^{cg/6}/n \geq d^{cg/12}$ by the assumption on g. Since $S \geq \log_2 n$ and T is at most polynomial in n (or the bound applies already), $T 2^{3S}$ is at most $2^{c'S}$ for some constant $c' > 0$. This implies that S is $\Omega(g(m) \log d)$ and since $T \geq n$, we get that $T \cdot S$ is $\Omega(g(m)\, n \log |D|)$ as claimed.

Otherwise set $k \leq g$ to be the smallest integer such that $h\, 2^{2S} d^{-ck/6} \leq 2^{-S}/T$. Then the probability that a sub-circuit produces k correct outputs is at most $2^{-S}/T$. This gives $k = \lceil [6 \log_2(hT) + 18S]/(c \log_2 d) \rceil$, which is at most $c^* S/\log_2 d$ for some constant $c^* > 0$ since S is $\Omega(\log n)$ which is $\Omega(\log(hT))$.

Taking a union bound over the sub-circuits, the probability that any of them produces k correct outputs is at most 2^{-S}. Since f has m outputs, this means that

$$\lceil T/(\alpha h) \rceil\, (k-1) \geq m$$

Since $T \geq n \geq \alpha h$, we have

$$2Tk \geq \alpha m h.$$

Plugging in our upper bound on k we have that

$$2c^*TS/\log_2 d \geq \alpha mh$$

and hence $T \cdot S$ is $\Omega(mh \log d)$ which is $\Omega(m\, h(n) \log |D|)$ as claimed. □

Following the same arguments as for classical computation [Abr91], we obtain a collection of time-space lower bounds for problems that are closely related to matrix vector products. Our proofs are identical to their classical counterparts proven in[Abr91, Sections 5-6] and are duplicated here for completeness.

Corollary 3.6. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ such that $d = |D|$. Any quantum circuit that computes the discrete Fourier transform (DFT) of vectors in D^n in time T and space S with probability at least 2^{-S} requires T to be $\Omega(n^2 \log(d)\ /S)$.*

Proof. Applying Theorem 3.5 with the rigidity of the DFT from Proposition 2.3 directly gives us the lower bound. □

Proposition 3.7 ([Abr91])**.** *There is a constant $\gamma \in (0, 1/2)$ such that at least a $1 - |D|^{-1}(2/3)^{\gamma n}$ fraction of the Toeplitz (diagonal constant) matrices over $D^{n\times n}$ are $(\gamma n, \gamma n)$-rigid.*

Corollary 3.8. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ such that $d = |D|$. Computing the convolution of two vectors in D^n in time T and space S with probability at least 2^{-S} requires T to be $\Omega(n^2 \log(d)\ /S)$*

Proof. For simplicity assume that n is even. Let

$$U = \begin{bmatrix} u_n & u_{n-1} & \dots & u_2 & u_1 \\ u_1 & u_n & \dots & u_3 & u_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ u_{n-2} & u_{n-3} & \dots & u_n & u_{n-1} \\ u_{n-1} & u_{n-2} & \dots & u_1 & u_n \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

Where A, B, C and D are $n/2 \times n/2$ submatrices. Then Uv is the convolution between vectors u and v. Observe that U is a Toeplitz matrix and by picking u to be a uniform vector over D, Proposition 3.7 tells us that for sufficiently large n, there is a constant $\gamma \in (0, 1/2)$ such that both A and B are $(\gamma n, \gamma n/2)$-rigid with probability at least $1/2$. This lets us restrict our input to such choices for u and observe that the matrix $U' = \begin{bmatrix} A & B \end{bmatrix}$ is $(\gamma n, \gamma n/2)$-rigid, so Theorem 3.5 gives us that computing U' requires T that is $\Omega(n^2 \log(d)\ /S)$. Since U' is a subfunction of U, convolution also requires T that is $\Omega(n^2 \log(d)\ /S)$. □

Corollary 3.9. *A quantum circuit that multiplies two n bit binary numbers in time T and space S with probability at least 2^{-S} requires T to be $\Omega(n^2/(S \log^2 n))$.*

Proof. Let u, v be arbitrary vectors over $\mathbb{F}_2$. Define the binary number

$$u' = 0^{\lceil \log_2 n \rceil - 1} u_n \dots 0^{\lceil \log_2 n \rceil - 1} u_1 0^{\lceil \log_2 n \rceil - 1} u_n \dots 0^{\lceil \log_2 n \rceil - 1} u_1$$

and similarly define v'. Then observe that the product $u' \cdot v'$ contains all entries of the convolution between u and v encoded in blocks of $\lceil \log_2 n \rceil$ bits each. By Corollary 3.8 this requires T to be $\Omega(n^2/(S \log^2 n))$. □

Proposition 3.10 ([Abr91])**.** *Let $A, B, C \in D^{n\times n}$ and $\mathcal{Y}$ (and $\mathcal{Y}$) be the vectors in D^{n^2} formed by stacking the transposes of the rows of B (and Y) into a column vector. If D is a commutative ring, then the following conditions are equivalent:*

$$Y = ABC$$
$$\mathcal{Y} = (A \otimes C^T)\mathcal{B}$$

Where $\otimes$ is the standard tensor (Kronecker) product.

Proposition 3.11 ([Abr91])**.** *Let $\gamma \in (0, 1/2)$. If A and B are $(\gamma n, \gamma n)$-rigid, then $A \otimes B$ is $(\gamma^2 n^2, \gamma^2 n^2, \gamma^2)$-rigid.*

Corollary 3.12. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ such that $d = |D|$. Any quantum circuit that computes the product ABC on inputs $A, B, C \in D^{n\times n}$ in time T and space S with probability at least 2^{-S} requires T that is $\Omega(n^4 \log(d) \,/S)$.*

Proof. We use Proposition 3.10 to view this as a matrix-vector product problem where $\mathcal{B}$ is the input and $\mathcal{Y}$ is the output. By Proposition 2.4 there is a constant $\gamma \in (0, 1/2)$ such that both A and C are γ rigid with constant probability, so we can assume such without increasing the expected cost by more than a constant factor. Then Proposition 3.11 gives us that $A \otimes C$ is $(\gamma^2 n^2, \gamma^2 n^2, \gamma^2)$-rigid and we can apply Theorem 3.5 to get that T must be $\Omega(n^4 \log(d) \,/S)$ as desired. □

Corollary 3.13. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ such that $d = |D|$. Any quantum circuit that computes A^3 on inputs in $D^{n\times n}$ in time T and space S with probability at least 2^{-S} requires T that is $\Omega(n^4 \log(d) \,/S)$.*

Proof. Let $A, B, C \in D^{n\times n}$. Then construct the $4n \times 4n$ matrix:

$$M = \begin{bmatrix} 0 & A & 0 & 0 \\ 0 & 0 & B & 0 \\ 0 & 0 & 0 & C \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Observe that the top right $n \times n$ sub-matrix of M^3 is equal to the product ABC. Thus we get a reduction to matrix-matrix-matrix product and can apply Corollary 3.12 to get our lower bound. □

Corollary 3.14. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ such that $d = |D|$. Any quantum circuit that computes A^{-1} on inputs in $D^{n\times n}$ in time T and space S with probability at least 2^{-S} requires T that is $\Omega(n^4 \log(d)/S)$.*

Proof. Let $A, B, C \in D^{n\times n}$. Then construct the $4n \times 4n$ matrix:

$$M = \begin{bmatrix} I & -A & 0 & 0 \\ 0 & I & -B & 0 \\ 0 & 0 & I & -C \\ 0 & 0 & 0 & I \end{bmatrix}$$

Where I is the $n \times n$ identity submatrix. Then observe that M^{-1} has the product ABC as its top right $n \times n$ submatrix. We can again use Theorem 3.5 to get our lower bound. □

Corollary 3.15. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ such that $d = |D|$. Any quantum circuit that solves any $n \times n$ system of linear equations over D in time T and space S with probability at least 2^{-S} requires T that is $\Omega(n^3 \log(d)\ /S)$*

Proof. It is possible to invert a matrix by solving n systems of n linear equations. By a reduction Corollary 3.14 gives us that solving these equations requires T that is $\Omega(n^4 \log(d)\ /S)$. Thus least one of these equations must require T that is $\Omega(n^3 \log(d)\ /S)$ to solve. □

In [BK23] the authors showed that the kinds of quantum time-space product lower bounds we proved in this section can be extended to asymptotically equivalent lower bounds on the stronger notion of cumulative memory complexity. We restate a simplified version of their main theorem for quantum circuits here.

Proposition 3.16 ([BK23]). *Let $f : D^n \to R^m$ be a function such that there exists constant C, functions $m'(n) \in \omega(\log n), h(k,n) = k^\Delta h_1(n), K(n)$, and a distribution μ over D^n where when $x \sim \mu$ the probability that - for any $k \leq m'(n)$ - any quantum circuit with at most $h(k,n)$ queries to x produces k correct outputs of $f(x)$ with probability at most $C \cdot K(n)^{-k}$. Then for any constant $c > 0$, any quantum circuit that computes f with T queries and error $\epsilon \leq (1 - 1/(2T^c))$ must have cumulative memory that is:*

$$\Omega\left(\min\left([(mh_1(n))^{1/(1-\Delta)} \log K(n)]/T^{\Delta/(1-\Delta)}, m'(n)^{1+\Delta} h_1(n) \log K(n)\right)\right) \tag{6}$$

Using the above result, we can extend the quantum time-space product lower bound for matrix vector products to a matching quantum cumulative memory lower bound.

Theorem 3.17. *Let $\gamma > 0$ and $c \in (0, 1/2]$ be fixed. If A is a $(\gamma n, \gamma n, c)$-rigid $n \times n$ matrix over a field $\mathbb{F}$ then any quantum circuit using time T and space S that computes the function $f : D^n \to \mathbb{F}^n$ for $D \subseteq \mathbb{F}$ with $d = |D|$ given by $f(x) = Ax$ with success probability larger than $1/T$ requires cumulative memory that is $\Omega(n^2 \log d)$.*

Proof. By Lemma 3.1 we can apply Proposition 3.16 where $C = \lceil 1/c \rceil$, $m'(n) = \gamma n$, $\Delta = 0$, $h_1(n) = \alpha n$, $K(n) = d^{1/6}$, and μ is the uniform distribution. This give us that any quantum circuit computing f with T queries and error at most $1 - 1/(2T)$ requires cumulative memory $\Omega(n^2 \log d)$ as desired. □

Directly applying this in place of Theorem 5.5 gives us matching cumulative (CM) memory lower bounds for Corollary 3.6 through Corollary 3.15.

Corollary 3.18. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ such that $d = |D|$. Any quantum circuit with inputs over D that computes the DFT or vector convolution requires CM that is $\Omega(n^2 \log d)$. Any quantum circuit that computes the product of three matrices, matrix cubing, or matrix inversion requires CM that is $\Omega(n^4 \log d)$. Any quantum circuit that solves $n \times n$ systems of linear equations requires CM that is $\Omega(n^3 \log d)$. Additionally any quantum circuit that multiplies two n bit binary numbers requires CM that is $\Omega(n^2 / \log^2 n)$.*

4 Quantum matrix multiplication

While many of the applications so far, including the matrix triple product lower bound discussed in the previous section, are derived from the matrix-vector product lower bound, our matrix multiplication lower bound requires a separate argument using ideas from the classical lower bound for the problem in [Abr91]. Implementing this requires a much more subtle way of applying our bucketing method for states that allows us to concentrate on just a subset of the buckets containing most of the total amplitude and ignore the others. As in Section 3, our lower bounds in this section apply to a more general model of quantum circuits that can decide which outputs they want to produce in a given layer based on the inputs that they have queried.

4.1 The success probability of small depth quantum circuits

Lemma 4.1. *Let $\gamma \in (0, 1/2)$ and $f : D^{n^2} \times D^{n^2} \to \mathbb{F}^{n^2}$ for $D \subseteq \mathbb{F}$ with $|D| = d$ be defined by $f(A, B) = AB$. Then for any constant $\beta > 0$ and quantum circuit $\mathcal{C}$ with at most $h = \beta\gamma n\sqrt{k/2}$ queries to input matrices A, B sampled uniformly from D^{n^2}, the probability that A and B are $(\gamma n, \gamma n)$-rigid and $\mathcal{C}$ produces k correct output values of $f(A, B)$ is at most $16 \min(k, n)^{\sqrt{k/2}} (2^{H_2(4\beta)}/d^{1-4\beta})^{k/4}$*

Note that for $\beta \le 0.0429$ we have $1 - 4\beta - H_2(4\beta) > 1/6$ so the bound is at most $16 \min(k, n)^{\sqrt{k/2}} d^{-k/24}$.

Proof. Let $C = AB$, $\Pi_{\mathrm{rigid}(A)}$ ($\Pi_{\mathrm{rigid}(B)}$) be the projection onto inputs where A (B) is a $(\gamma n, \gamma n)$-rigid matrix, and define $\Pi_{\mathrm{rigid}} = \Pi_{\mathrm{rigid}\ A}\Pi_{\mathrm{rigid}\ B}$. Assume that $q(w)$—the output as a function of the measured value of the work register—produces exactly k outputs; we ignore anything it produces after the first k. We will use $[A]$ to denote the set of indices of elements in A and likewise for $[B]$ and $[C]$. By Proposition 2.8, after $t \le h$ queries in the recording query basis, our state can be written as:

$$|\phi_t\rangle = \sum_{\substack{i,p,w \\ E \subseteq [A], F \subseteq [B] \\ |E|+|F| \le t \\ x \in D^E, y \in D^F}} \alpha_{i,p,w,E,F,x,y} |i, p, w\rangle |x\rangle_E |\bot\rangle_{[A]\setminus E} |y\rangle_F |\bot\rangle_{[B]\setminus F}$$

for some $\alpha_{i,p,w,E,F,x,y}$ with $\sum_{i,p,w,E,F,x,y} |\alpha_{i,p,w,E,F,x,y}|^2 = 1$. We first apply analogous series observations and decompositions to those that allowed us to derive (4) from (3) in the case of matrix-vector product. By Proposition 2.7, we note that the final state of the algorithm in the standard oracle setting is given by:

$$|\psi_t\rangle = \mathcal{S} |\phi_t\rangle = \sum_{\substack{i,p,w \\ E \subseteq [A], F \subseteq [B] \\ |E|+|F| \le t \\ x \in D^E, y \in D^F}} \alpha_{i,p,w,E,F,x,y} |i, p, w\rangle |x\rangle_E |\bot\rangle_{[A]\setminus E} |y\rangle_F |\bot\rangle_{[B]\setminus F}$$

Because $\mathcal{S}$ behaves as the identity on $|\psi\rangle_C$ and each distinct choice of $|i, p, w\rangle$ gives an orthogonal basis state, this equals:

$$\sum_{i,p,w} \beta_{i,p,w} |i, p, w\rangle \otimes \left[S_1^{\otimes 2n^2} \sum_{\substack{E \subseteq [A], F \subseteq [B] \\ |E|+|F| \le t \\ x \in D^E, y \in D^F}} \beta^{i,p,w}_{E,F,x,y} |x\rangle_E |\bot\rangle_{[A]\setminus E} |y\rangle_F |\bot\rangle_{[B]\setminus F} \right]$$

for some $\beta_{i,p,w}$ and $\beta^{i,p,w}_{E,F,x,y}$ such that $\sum_{i,p,w} |\beta_{i,p,w}|^2 = 1$ and $\sum_{E,F,x,y} |\beta^{i,p,w}_{E,F,x,y}|^2 = 1$ for each i, p, w. Now the probability over the choices of the input matrices and the result of the quantum algorithm making t queries that the matrices A and B are both $(\gamma n, \gamma n)$-rigid and the algorithm produces k correct output values from $C = AB$ is at most:

$$
\begin{aligned}
&\left\|\Pi_k \Pi_{\text{rigid}} \mathcal{S} \left|\phi_t\right\rangle\right\|^2 \\
&= \left\| \Pi_k \Pi_{\text{rigid}} \sum_{i,p,w} \beta_{i,p,w} \left|i,p,w\right\rangle \otimes \left[S_1^{\otimes 2n^2} \sum_{\substack{E\subseteq[A],F\subseteq[B]\\ |E|+|F|\leq t\\ x\in D^E, y\in D^F}} \beta^{i,p,w}_{E,F,x,y} \left|x\right\rangle_E \left|\perp\right\rangle_{[A]\setminus E} \left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F} \right] \right\|^2 \\
&= \left\| \sum_{i,p,w} \beta_{i,p,w} \left|i,p,w\right\rangle \otimes \left[\Pi_{q(w)} \Pi_{\text{rigid}} S_1^{\otimes 2n^2} \sum_{\substack{E\subseteq[A],F\subseteq[B]\\ |E|+|F|\leq t\\ x\in D^E, y\in D^F}} \beta^{i,p,w}_{E,F,x,y} \left|x\right\rangle_E \left|\perp\right\rangle_{[A]\setminus E} \left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F} \right] \right\|^2 \\
&= \sum_{i,p,w} |\beta_{i,p,w}|^2 \left\| \left[\Pi_{q(w)} \Pi_{\text{rigid}} S_1^{\otimes 2n^2} \sum_{\substack{E\subseteq[A],F\subseteq[B]\\ |E|+|F|\leq t\\ x\in D^E, y\in D^F}} \beta^{i,p,w}_{E,F,x,y} \left|x\right\rangle_E \left|\perp\right\rangle_{[A]\setminus E} \left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F} \right] \right\|^2 \\
&\leq \max_{i,p,w} \left\| \Pi_{q(w)} \Pi_{\text{rigid}} \mathcal{S}_1^{\otimes 2n^2} \sum_{\substack{E\subseteq[A],F\subseteq[B]\\ |E|+|F|\leq t\\ x\in D^E, y\in D^F}} \beta^{i,p,w}_{E,F,x,y} \left|x\right\rangle_E \left|\perp\right\rangle_{[A]\setminus E} \left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F} \right\|^2. \qquad (7)
\end{aligned}
$$

For the rest of the proof we fix an i, p, w to achieve the maximum value in Equation (7) and prove a upper bound on the resulting probability. This fixes the output values $q(w)$; we write $G \subseteq [C]$ with $|G| = k$ for the set of indices of the outputs given by $q(w)$. To keep notations simpler in the remainder of the proof we observe that Equation (7) is upper bounded by the maximum of

$$
\left\| \Pi_{q(G)} \Pi_{\text{rigid}} \mathcal{S}_1^{\otimes 2n^2} \sum_{\substack{E\subseteq[A],F\subseteq[B]\\ |E|,|F|\leq t\\ x\in D^E, y\in D^F}} \beta_{E,F,x,y} \left|x\right\rangle_E \left|\perp\right\rangle_{[A]\setminus E} \left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F} \right\|^2. \qquad (8)
$$

over all $\beta_{E,F,x,y}$ with $\sum_{E,F,x,y} |\beta_{E,F,x,y}|^2 = 1$, all sets $G \subseteq [C]$ with $|G| = k$ and all assignments $q(G)$ to G.

We will split the sum in Equation (8) over the different sets E and F of queried input indices depending on how they relate to the set of output indices given by G. Let $r(G)$ be the set of rows containing elements of G and $c(G)$ be the set of columns containing elements of G.

We define a *light row of E* to be an element of $r(G)$ that contains at most $\beta\gamma n$ elements of E and define a *light column of F* to be an element of $c(G)$ that contains at most $\beta\gamma n$ elements of F. Since $|E|, |F| \leq t \leq \beta\gamma n\sqrt{k/2}$ we have $\leq \sqrt{k/2}$ rows of E in $r(G)$ and $\leq \sqrt{k/2}$ columns of F in $c(G)$ that are not light. We define $\mathcal{L}(E) \subseteq r(G)$, to be any set of $|r(G)| - \lfloor\sqrt{k/2}\rfloor$ light rows of E and $\mathcal{L}'(F) \subseteq c(G)$ to be any set of $c(G) - \lfloor\sqrt{k/2}\rfloor$ light columns of F. Therefore $|\{(i',j') \in G \mid i' \notin \mathcal{L}(E),\ j' \notin \mathcal{L}'(F)\}| \leq k/2$ so at least $k/2$ elements of G are in light rows of E or in light columns of F. Therefore for every pair (E, F) at least one of the sets of outputs $G^r_{\mathcal{L}(E)} = \{(i', j') \in G \mid i' \in \mathcal{L}(E)\}$ or $G^c_{\mathcal{L}(F)} = \{(i', j') \in G \mid j' \in \mathcal{L}'(F)\}$ has size $\geq k/4$.

Let $\mathcal{E}$ be the set of all $E \subseteq [A]$ with $|E| \leq t$ such that G has many outputs in light rows, $|G^r_{\mathcal{L}(E)}| \geq k/4$, and $\mathcal{F}$ be the set of all $F \subseteq [B]$ with $|F| \leq t$ such that G has many outputs in light columns, $|G^c_{\mathcal{L}'(F)}| \geq k/4$. We separately bound the contribution to Equation (8) from pairs (E, F) with $E \in \mathcal{E}$ and $F \in \mathcal{F}$. The analyses of the two cases are completely symmetric up to matrix transposition. It will be convenient to focus on the case $F \in \mathcal{F}$ that there are many outputs of G in light columns and compute an upper bound on

$$\left\| \Pi_{q(G)} \Pi_{\text{rigid}} \mathcal{S}_1^{\otimes 2n^2} \sum_{\substack{E \subseteq [A] \\ |E| \leq t \\ x \in D^E}} \sum_{\substack{F \in \mathcal{F} \\ y \in D^F}} \beta_{E,F,x,y} \left| x \right\rangle_E \left| \perp \right\rangle_{[A] \setminus E} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\|^2. \tag{9}$$

The case that $E \in \mathcal{E}$ has exactly the same upper bound as Equation (9) by applying the argument to the transposed product $B^T A^T$ and corresponding transposed sets F^T, E^T, and G^T. Hence, the quantity in Equation (8) is at most 4 times that of Equation (9).

To upper bound Equation (9), we first remove the projection operator $\Pi_{\text{rigid } B}$ from $\Pi_{q(G)} \Pi_{\text{rigid}} = \Pi_{q(G)} \Pi_{\text{rigid } A} \Pi_{\text{rigid } B}$ to get $\Pi_{q(G)} \Pi_{\text{rigid } A}$. We then rewrite this combined projection operator as $\Pi_{q(G)} \Pi_{\text{rigid } A} = \sum_{A\ (\gamma n, \gamma n)\textbf{-rigid}} \Pi_A \otimes \Pi^A_{q(G)}$ where Π_A is the projection onto the specific matrix A and for each A, $\Pi^A_{q(G)}$ is the projection onto the choices for matrix B such that $C = AB$ agrees with $q(w)$. We therefore obtain that Equation (9) is at most

$$\begin{aligned}
&\left\| \sum_{A\ (\gamma n, \gamma n)\text{-rigid}} (\Pi_A \otimes \Pi^A_{q(G)}) \mathcal{S}_1^{\otimes 2n^2} \sum_{\substack{E \subseteq [A] \\ |E| \leq t \\ x \in D^E}} \sum_{\substack{F \in \mathcal{F} \\ y \in D^F}} \beta_{E,F,x,y} \left| x \right\rangle_E \left| \perp \right\rangle_{[A] \setminus E} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\|^2 \\
&= \left\| \sum_{A\ (\gamma n, \gamma n)\text{-rigid}} (\Pi_A \otimes \Pi^A_{q(G)} \mathcal{S}_1^{\otimes n^2}) \sum_{A' \in (D \cup \{\perp\})^{[A]}} \sum_{\substack{F \in \mathcal{F} \\ y \in D^F}} \beta_{A'} \beta^{A'}_{F,y} \left| A' \right\rangle_{[A]} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\|^2 \\
&= \left\| \sum_{A\ (\gamma n, \gamma n)\text{-rigid}} \beta_A \left| A \right\rangle_{[A]} \otimes [\Pi^A_{q(G)} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{F} \\ |F| \leq t \\ y \in D^F}} \beta^A_{F,y} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F}] \right\|^2
\end{aligned} \tag{10}$$

for some β_A and $\beta^A_{F,y}$ such that $\sum_{A \in (D \cup \{\perp\})^{n^2}} |\beta_A|^2 = 1$ and $\sum_{F \in \mathcal{F}, y \in D^{F0}} |\beta^A_{F,y}|^2 = 1$ for each A. Since $\Pi_{q(G)^A}$ only projects onto the $[B]$ input registers, each distinct choice of $\left| A \right\rangle_{[A]}$ gives orthogonal states so Equation (10) equals

$$\begin{aligned}
&\sum_{A\ (\gamma n, \gamma n)\text{-rigid}} |\beta_A|^2 \left\| \Pi^A_{q(G)} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{F} \\ |F| \leq t \\ y \in D^F}} \beta^A_{F,y} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\|^2 \\
&\leq \max_{A\ (\gamma n, \gamma n)\text{-rigid}} \left\| \Pi^A_{q(G)} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{F} \\ y \in D^F}} \beta^A_{F,y} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\|^2
\end{aligned} \tag{11}$$

We fix a $(\gamma n, \gamma n)$-rigid matrix A that maximizes (11). We now partition the set $\mathcal{F}$ based on the set $\mathcal{L}'(F)$ which contains all but precisely $\lfloor\sqrt{k/2}\rfloor$ columns in $c(G)$. Therefore we can rewrite (11) as

$$\left\| \sum_{H \in \binom{c(G)}{\lfloor\sqrt{k/2}\rfloor}} \Pi^A_{q(G)} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{F} \\ \mathcal{L}'(F) = c(G) \setminus H \\ y \in D^F}} \beta^A_{F,y} |y\rangle_F |\perp\rangle_{[B] \setminus F} \right\|^2. \tag{12}$$

Since the different choices of F, and hence different choices of H, correspond to orthogonal basis states, we can upper bound (12) by

$$\binom{|c(G)|}{\lfloor\sqrt{k/2}\rfloor} \cdot \max_{H \in \binom{c(G)}{\lfloor\sqrt{k/2}\rfloor}} \left\| \Pi^A_{q(G)} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{F} \\ \mathcal{L}'(F) = c(G) \setminus H \\ y \in D^F}} \beta^A_{F,y} |y\rangle_F |\perp\rangle_{[B] \setminus F} \right\|^2. \tag{13}$$

We fix the set H achieving the maximum value in Equation (13) which fixes the value of $\mathcal{L}'(F) = c(G) \setminus H$. This fixes the set $G^c_{\mathcal{L}'(F)}$ of elements in G that are in light columns of F (equivalently, not in H) which, since $F \in \mathcal{F}$, contains at least $k/4$ elements of G. Let G' be a fixed subset of $k/4$ of the elements of $G^c_{\mathcal{L}'(F)}$. By construction we have $c(G') \subseteq \mathcal{L}'(F)$. By only requiring that the outputs in G' are correct and using the fact that $|c(G)| \leq \min(k, n)$, we therefore can upper bound $\left\|\Pi_k \Pi_{\text{rigid}} \mathcal{S} |\phi_t\rangle\right\|^2$ by the maximum value of

$$4 \cdot \min(k, n)^{\sqrt{k/2}} \left\| \Pi^A_{q(G')} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \subseteq [B] \\ c(G') \subseteq \mathcal{L}'(F) \\ y \in D^F}} \beta'_{F,y} |y\rangle_F |\perp\rangle_{[B] \setminus F} \right\|^2. \tag{14}$$

over all $G' \subseteq [C]$ with $|G'| = k/4$ and $\beta'_{F,y}$ with $\sum_{F,y} |\beta'_{F,y}|^2 = 1$.

For each $j \in c(G')$, let k_j be the number of elements of G' in column j. Our overall strategy is to consider the $j \in c(G')$ one by one, and show that the total amplitude on states where these k_j outputs are correct conditioned on the success for previous values of j is of the form $d^{-\delta k_j}$ for some fixed constant $\delta > 0$. These are k_j outputs of the matrix-vector product Ay^j where y^j is the j-th column of B and that fact that $c(G') \subseteq \mathcal{L}'(F)$ implies that F has made at most $\beta\gamma n$ queries to $y^{(j)}$. This is very similar to the situation with the matrix-vector problem from Lemma 3.1. In analogy with the Lemma 3.1, we define U^j to be the set of k_j rows containing outputs of G' in column j.

Applying Lemma 3.3 with $c = 1$, for each $j \in q(G')$ there is a collection $V^j_1, \ldots, V^j_{\ell_j}$ of $\ell_j = \lceil \gamma n / k_j \rceil$ k_j-subsets of $[n]$ such that the $k_j \times k_j$ sub-matrix $A_{U^j V^j_i}$ has full rank.

Using the ideas of Lemma 3.1 we could bucket the possible quantum states into one bucket for each tuple $(V^j_i)_{j \in q(G')}$ using Lemmas 3.2 and 3.3 and bound each bucket separately. However, unlike Lemma 3.1, the value of many of the k_j can be very small, as low as 1, in which case the upper bounds using Lemmas 3.2 and 3.3 would yield a probability bound larger than 1.

Instead, we need a stronger argument to show that, except for an exponentially small amount in k, all of the amplitude can be allocated to a very small number of buckets. The following lemma gives the inductive step that allows us to define those buckets. Rather than thinking about each column $j \in c(G')$ as separate matrix-vector problems, it works by considering all of the answers in G' at once.

Lemma 4.2. *Let $G' \subseteq [C]$ with $|G'| = k/4$ and $\mathcal{F}'$ be a set of $F \subseteq [B]$ such that $c(G') \subseteq \mathcal{L}'(F)$. Suppose further that $\sum_{F\in\mathcal{F}',y\in D^F} |\delta_{F,y}|^2 = 1$ for some $\delta_{F,y}$. Let $C' \geq 2$ be a constant and define $\alpha = C'\beta$. Then there is a $\mathcal{F}'' \subseteq \mathcal{F}'$ and $\delta'_{F,y}$ such that $\sum_{F\in\mathcal{F}'',y\in D^F} |\delta'_{F,y}|^2 = 1$ and*

$$\Big\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}'\\ y\in D^F}} \delta_{F,y}\,|y\rangle_F\,|\bot\rangle_{[B]\setminus F}\Big\|^2 \leq \frac{2^{1+H_2(\alpha)\,k/4}}{d^{(1-\alpha)\,k/4}} + \frac{2}{C'}\Big\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}''\\ y\in D^F}} \delta'_{F,y}\,|y\rangle_F\,|\bot\rangle_{[B]\setminus F}\Big\|^2.$$

Proof. We first recall the definitions in our discussion preceding the lemma statement. For each $j \in c(G')$, define U^j to be the set of row indices of G' in column j and let $k_j = |U_j|$; let $\ell_j = \lceil \gamma n/k_j \rceil$ and $V_1^j, \ldots, V_{\ell_j}^j$ be the collection of disjoint subsets of $[n]$ of size k_j given by Lemma 3.3 such that each $k_j \times k_j$ sub-matrix $A_{U^j V_i^j}$ has full rank.

For each $F \in \mathcal{F}'$ and $i \in c(G')$, define F^j to be the set of row indices of elements of F in column j; since $c(G') \subseteq \mathcal{L}'(F)$, we have $|F^j| \leq \beta\gamma n$. For each $i \in [\ell_j]$ define

$$m_i^j = \sum_{F\in\mathcal{F}',\ y\in D^F} |\delta_{F,y}|^2 \cdot |F^j \cap V_i^j|.$$

Since $\sum_{F,y} |\delta_{F,y}|^2 = 1$, m_i^j can be viewed as the expected size of the overlap between the recorded queries in the j-th column of the matrix B and each V_i^j. Since for each j, the sets V_i^j are disjoint and $|F^j| \leq \beta\gamma n$ we have $\sum_{i\in[\ell_j]} m_i^j \leq \beta\gamma n$. Therefore, for each j, we have some index $i_j \in [\ell_j]$ such that $m_{i_j}^j \leq \beta\gamma n/\ell_j \leq \beta k_j$.

Since $\sum_{j\in c(G')} k_j = |G'| = k/4$, the expected total overlap between the recorded queries in the columns of G and the chosen sets $V_{i_j}^j$ for those columns is $\sum_j m_{i_j}^j \leq \sum_j \beta k_j = \beta k/4$. Define $\mathcal{F}''$ to be the set of $F \in \mathcal{F}'$ such that $\sum_j |F^j \cap V_{i_j}^j| \geq \alpha k/4 = C'\beta k/4$. By Markov's inequality we have

$$\sum_{F\in\mathcal{F}'',\ y\in D^F} |\delta_{F,y}|^2 \leq \frac{\sum_j m_{i_j}^j}{C'\beta k/4} \leq 1/C'.$$

We split our analysis for $\mathcal{F}'$ into two parts due to sets F in $\mathcal{F}''$ and $\mathcal{F}' \setminus \mathcal{F}''$, respectively.

We begin with $F \in \mathcal{F}''$. Write $\kappa = \sum_{F\in\mathcal{F}'',\ y\in D^F} |\delta_{F,y}|^2 \leq 1/C'$. For $F \in \mathcal{F}''$, define $\delta'_{F,y} = \frac{1}{\sqrt{\kappa}}\delta_{F,y}$. Then $\sum_{F\in\mathcal{F}'',y\in D^F} |\delta'_{F,y}|^2 = 1$ and

$$\begin{aligned}\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}''\\ y\in D^F}} \delta_{F,y}\,|y\rangle_F\,|\bot\rangle_{[B]\setminus F}\right\|^2 &= \kappa\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}''\\ y\in D^F}} \delta'_{F,y}\,|y\rangle_F\,|\bot\rangle_{[B]\setminus F}\right\|^2 \\ &\leq \frac{1}{C'}\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}''\\ y\in D^F}} \delta'_{F,y}\,|y\rangle_F\,|\bot\rangle_{[B]\setminus F}\right\|^2. \end{aligned} \tag{15}$$

We now consider $\mathcal{F}' \setminus \mathcal{F}''$. By definition, for $F \in \mathcal{F}' \setminus \mathcal{F}''$, we have $\sum_j |F^j \cap V^j_{i_j}| < \alpha k/4$. By definition we have $\sum_j |V^j_{i_j}| = \sum_j k_j = k/4$ so F must miss more than $(1-\alpha)k/4$ elements of the set $V = \bigcup_j (V^j_{i_j} \times \{j\})$ of size $k/4$. For each subset V' of V of size $k/4 - \lfloor \alpha k/4 \rfloor$ we define a bucket $\mathcal{B}_{V'}$ that contains sets F that must the elements of V' and assign each $F \in \mathcal{F}' \setminus \mathcal{F}''$ to a unique bucket in an arbitrary fixed way. There are at most $2^{H_2(\alpha)k/4}$ such buckets. Then

$$\begin{aligned}
&\left\| \Pi^A_{q(G')} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{F}' \setminus \mathcal{F}'' \\ y \in D^F}} \delta_{F,y} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\|^2 \\
&\leq \left(\sum_{\substack{V' \subseteq V \\ |V'| = k/4 - \lfloor \alpha k/4 \rfloor}} \left\| \Pi^A_{q(G')} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{B}_{V'} \\ y \in D^F}} \delta_{F,y} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\| \right)^2 \\
&\leq 2^{H_2(\alpha)\, k/4} \cdot \sum_{\substack{V' \subseteq V \\ |V'| = k/4 - \lfloor \alpha k/4 \rfloor}} \left\| \Pi^A_{q(G')} \mathcal{S}_1^{\otimes n^2} \sum_{\substack{F \in \mathcal{B}_{V'} \\ y \in D^F}} \delta_{F,y} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus F} \right\|^2 \\
&= 2^{H_2(\alpha)\, k/4} \cdot \sum_{\substack{V' \subseteq V \\ |V'| = k/4 - \lfloor \alpha k/4 \rfloor}} \left\| \Pi^A_{q(G')} \mathcal{S}_1^{\otimes n^2} \left| \perp \right\rangle_{V'} \sum_{\substack{F \in \mathcal{B}_{V'} \\ y \in D^F}} \delta_{F,y} \left| y \right\rangle_F \left| \perp \right\rangle_{[B] \setminus (F \cup V')} \right\|^2 \qquad (16)
\end{aligned}$$

where we first used the triangle inequality followed by Jensen's inequality.

Now, applying the $\mathcal{S}_1^{\otimes n^2}$ operator in (16) will convert the $|\perp\rangle_{V'}$ to a uniform superposition of all $|y'\rangle_{V'}$ for all $y' \in D^{V'}$ and convert $\sum_{\substack{F \in \mathcal{B}_{V'} \\ y \in D^F}} \delta_{F,y} |y\rangle_F |\perp\rangle_{[B] \setminus (F \cup V')}$ to some superposition of $|y''\rangle \in D^{[B] \setminus V'}$ with amplitudes some $\delta_{V',y''}$ such that $\sum_{y''} |\delta_{V',y''}|^2 = \sum_{F \in \mathcal{B}_{V'}, y \in D^F} |\delta_{F,y}|^2$. Therefore, we can rewrite (16) as

$$2^{H_2(\alpha)\, k/4} \cdot \sum_{\substack{V' \subseteq V \\ |V'| = k/4 - \lfloor \alpha k/4 \rfloor}} \left\| \Pi^A_{q(G')} \left[\sum_{y' \in D^{V'}} \frac{1}{\sqrt{d^{|V'|}}} \left| y' \right\rangle_{V'} \right] \otimes \sum_{y'' \in D^{[n] \setminus V'}} \delta_{V',y} \left| y \right\rangle_{[B] \setminus V'} \right\|^2. \qquad (17)$$

We now consider the application of $\Pi^A_{q(G')}$. Let $V'_j \subseteq V^j_{i_j}$ be the set of row indices in column j of $V' \subseteq [B]$ and consider the corresponding set of columns in A. Since $A_{U^j V^j_{i_j}}$ has full rank, there is a subset $U^j_0 \subseteq U^j$ with $|U^j_0| = |V'_j|$ so that $A_{U^j_0 V'_j}$ also has full rank. Now define $G'_0 \subseteq G'$ to be $\bigcup_{j \in c(G)} (U_j \times \{j\})$ which has size $|V'|$.

For each j, the outputs in $U_j \times \{j\} \subset [C]$ can be expressed as the matrix-vector product $A_{U^j_0 V'_j} y^j_{V'_j} + M$ for some $|V'_j| \times |V'_j|$ matrix M defined by the product of the $U^j_0 \times ([n] \setminus V'_j)$ submatrix of the fixed matrix A and $y^j_{[n] \setminus V'_j}$. Since $A_{U^j_0 V'_j}$ is full rank, for each value of M given by $y^j_{[n] \setminus V'_j}$, there is precisely one value of $y^j_{V'_j}$ that will yield the output values $q(U_j \times \{j\})$. Therefore, putting the properties for the columns of $c(G')$ together, there is precisely one value $y' \in D^{V'}$ that will yield the output values $q(G'_0)$. Therefore, (17) is at most

$$
\begin{aligned}
&2^{H_2(\alpha)\,k/4}\cdot \sum_{\substack{V'\subseteq V\\ |V'|=k/4-\lfloor \alpha k/4\rfloor}} \left\| \Pi^A_{q(G'_0)} \left[\sum_{y'\in D^{V'}} \frac{1}{\sqrt{d^{|V'|}}} \left|y'\right\rangle_{V'} \right] \otimes \sum_{y''\in D^{[n]\setminus V'}} \delta_{V',y} \left|y\right\rangle_{[B]\setminus V'} \right\|^2 \\
&= 2^{H_2(\alpha)\,k/4}\cdot \sum_{\substack{V'\subseteq V\\ |V'|=k/4-\lfloor \alpha k/4\rfloor}} \left\| \frac{1}{\sqrt{d^{|V'|}}} \sum_{y''\in D^{[n]\setminus V'}} \delta_{V',y} \left|y\right\rangle_{[B]\setminus V'} \right\|^2 \\
&= 2^{H_2(\alpha)\,k/4}\cdot \sum_{\substack{V'\subseteq V\\ |V'|=k/4-\lfloor \alpha k/4\rfloor}} \frac{1}{d^{|V'|}} \sum_{y''\in D^{[n]\setminus V'}} |\delta_{V',y}|^2 \\
&= 2^{H_2(\alpha)\,k/4}\cdot \sum_{\substack{V'\subseteq V\\ |V'|=k/4-\lfloor \alpha k/4\rfloor}} \frac{1}{d^{|V'|}} \sum_{F\in\mathcal{B}_{V'},y\in D^F} |\delta_{F,y}|^2 \\
&= 2^{H_2(\alpha)\,k/4}\cdot \frac{1}{d^{|V'|}} \sum_{F\in\mathcal{F}'\setminus\mathcal{F}'',y\in D^F} |\delta_{F,y}|^2 \\
&\leq 2^{H_2(\alpha)\,k/4}/d^{(1-\alpha)\,k/4} \qquad (18)
\end{aligned}
$$

where the last equality follows since the buckets $\mathcal{B}_{V'}$ partition $\mathcal{F}'\setminus\mathcal{F}''$.

We now combine the contributions from $\mathcal{F}''$ and $\mathcal{F}'\setminus\mathcal{F}''$. Applying Jensen's inequality together with the bounds in (15) and (18) we obtain that

$$
\begin{aligned}
&\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}'\\ y\in D^F}} \delta_{F,y}\left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F}\right\|^2 \\
&\leq 2\left[\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}'\setminus\mathcal{F}''\\ y\in D^F}} \delta_{F,y}\left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F}\right\|^2 + \left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}''\\ y\in D^F}} \delta_{F,y}\left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F}\right\|^2\right] \\
&\leq \frac{2^{1+H_2(\alpha)\,k/4}}{d^{(1-\alpha)\,k/4}} + \frac{2}{C'}\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}''\\ y\in D^F}} \delta'_{F,y}\left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F}\right\|^2
\end{aligned}
$$

as required. ∎

Corollary 4.3. *Let $G'\subseteq[C]$ with $|G'|=k/4$, $\mathcal{F}'$ be a set of $F\subseteq[B]$ such that $c(G')\subseteq\mathcal{L}'(F)$, and $\sum_{F\in\mathcal{F}',y\in D^F}|\delta_{F,y}|^2=1$ for some $\delta_{F,y}$. Then*

$$
\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}'\\ y\in D^F}} \delta_{F,y}\left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F}\right\|^2 \leq 2^{2+H_2(4\beta)\,k/4}/d^{(1-4\beta)\,k/4}.
$$

Proof. Let M be the maximum value of $\left\|\Pi^A_{q(G')}\mathcal{S}_1^{\otimes n^2} \sum_{\substack{F\in\mathcal{F}'\\ y\in D^F}} \delta_{F,y}\left|y\right\rangle_F \left|\perp\right\rangle_{[B]\setminus F}\right\|^2$ over all choices of $\mathcal{F}'$ and $\delta_{F,y}$ with the required properties. This corollary follows from Lemma 4.2 with $C'=4$ by observing that the term multiplied by $2/C'$ is also upper bounded by M and hence $M\leq 2^{1+H_2(4\beta)\,k/4}/d^{(1-4\beta)\,k/4}+M/2$. ∎

Finally, plugging the bound from Corollary 4.3 into (14), we obtain that the probability that A and B are both $(\gamma n, \gamma n)$-rigid and $\mathcal{C}$ produces k correct output values for $C = AB$, $\left\|\Pi_k \Pi_{\text{rigid}} \mathcal{S} \left|\phi_t\right\rangle\right\|^2$, is at most

$$16 \ \min(k,n)^{\sqrt{k/2}} \left(\frac{2^{H_2(4\beta)}}{d^{(1-4\beta)}}\right)^{k/4}$$

as desired. □

4.2 Matrix multiplication time-space tradeoff lower bounds

Here we consider the matrix multiplication problem $f(A,B) = AB$ where both A and B are considered input. If we could fix a choice of A, we would be able to make our proof somewhat simpler. However, as Abrahamson pointed out in [Abr91], there is a classical algorithm that can compute the function $f(B) = AB$ for any fixed matrix A in $O(n^2)$ time and $O(n \log d)$ space. Thus our lower bound requires both A and B to be inputs to the function.

Theorem 4.4. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ with $d = |D|$. Then any quantum circuit $\mathcal{C}$ that uses time T and space S and computes the function $f : D^{2n^2} \to \mathbb{F}^{n^2}$ given by $f(A,B) = AB$ with success probability larger than $1/T$ must have T that is $\Omega(n^3 \sqrt{\log d \ /S})$.*

Proof. Let $\gamma \in (0, 1/2)$ be the constant given by Proposition 2.4. By that proposition, the probability that either of two matrices A and B chosen uniformly randomly from D^{n^2} is not $(\gamma n, \gamma n)$-rigid is at most $2d^{-1}(2/3)^{\gamma n}$. Let $\mathcal{C}$ be a quantum circuit with T queries and space S. Let $\beta = 0.0429$, $d = |D|$, and set $k = \lceil 48(5S+5)/\log_2 d \rceil$. We partition $\mathcal{C}$ into $\lceil T/(\beta\gamma n\sqrt{k/2}) \rceil$ sub-circuits that each have at most $\beta\gamma n\sqrt{k/2}$ queries. Without loss of generalities there are at most n^2 such sub-circuits. By combining Proposition 2.5 with Lemma 4.1, we know that for a uniformly random input, the probability that A and B are $(\gamma n, \gamma n)$-rigid matrices and a fixed sub-circuit can produce k outputs is at most $16k^{\sqrt{k/2}} 2^{2S} d^{-k/24}$. Therefore the probability that A and B are $(\gamma n, \gamma n)$-rigid matrices and one of the sub-circuits produces k correct outputs is at most $16k^{\sqrt{k/2}} 2^{2S} d^{-k/24} n^2$. Combining this with the probability that one of A or B is not $(\gamma n, \gamma n)$-rigid, the probability that there is a sub-circuit that produces k correct outputs is at most

$$16k^{\sqrt{k/2}} 2^{2S} d^{-k/24} n^2 + 2d^{-1}(2/3)^{2\gamma n}.$$

Since we can assume without loss of generality that $T \leq n^3$, for sufficiently large n, $2d^{-1}(2/3)^{2\gamma n} \leq 1/(2T)$ and $k^{\sqrt{k/2}} \leq 2^{k/48} \leq d^{k/48}$. Plugging in our value of k and the fact that $S \geq \log_2 n$ without loss of generality gives a probability of at most

$$\begin{aligned} 16k^{\sqrt{k/2}} 2^{2S} d^{-k/24} n^2 + 2d^{-1}(2/3)^{2\gamma n} &\leq 16 2^{2S} d^{-k/48} n^2 + 1/(2T) \\ &\leq 1/(2T) + 1/(2T) = 1/T. \end{aligned}$$

Since $\mathcal{C}$ must be correct with probability larger than $1/T$, this implies that

$$(k-1) \left\lceil T/(\beta\gamma n\sqrt{k/2}) \right\rceil \geq n^2.$$

Plugging in our value of k gives us that

$$T \text{ is } \Omega(n^3\sqrt{\log d}/\sqrt{S+\log T}).$$

Since $S \geq \log_2 n$ and our bound trivially holds when T is $\omega(n^3\sqrt{\log d})$ there is a constant $c > 0$ such that $cS \geq \log_2 T$. This implies that T is $\Omega(n^3\sqrt{\log d/S})$ as desired. $\square$

Corollary 4.5. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ with $d = |D|$. If $\mathcal{C}$ is a quantum circuit that computes the function $f : D^{n^2} \to \mathbb{F}^{n^2}$ where $f(A) = A^2$ on all upper triangular inputs in time T and space S with success probability at least $1/T$, then T must be $\Omega(n^3\sqrt{\log d\ /S})$.*

Proof. Let $A, B \in D^{n^2}$ and construct the $3n \times 3n$ matrix:

$$M = \begin{bmatrix} 0 & A & 0 \\ 0 & 0 & B \\ 0 & 0 & 0 \end{bmatrix}$$

Since the top right $n \times n$ sub-matrix of M^2 is equal to the product AB, we get a reduction from matrix multiplication and can apply Theorem 4.4 to derive the lower bound. $\square$

Using Proposition 3.16 we can also bound the cumulative memory complexity for these problems.

Corollary 4.6. *Let $\mathbb{F}$ be a field and $D \subseteq \mathbb{F}$ with $d = |D|$. If $\mathcal{C}$ is a quantum circuit that computes the function $f : D^{2n^2} \to \mathbb{F}^{n^2}$ given by $f(A, B) = AB$ or the function $g : D^{n^2} \to \mathbb{F}^{n^2}$ given by $f(A) = A^2$, then $\mathcal{C}$ must have cumulative memory complexity $\Omega(n^3\sqrt{\log d\ /S})$.*

Proof. For f, we apply Proposition 3.16 with Lemma 4.1 where m' is $\Theta(n^2)$, Δ is $1/2$, $h_1(n)$ is $\Theta(n)$, $K(n) = d^{-1/48}$, $C = 16$. This gives us that the cumulative memory complexity is $\Omega(n^6\log(d)\ /T)$. Using the same reduction as in Corollary 4.5, this same lower bound applies to computing g. $\square$

5 Quantum Tradeoffs for Boolean Matrix Operations

In this section we focus on Boolean matrix operations, which use $(\vee, \wedge)$ inner product of vectors rather than the usual $(+, \times)$ inner product. We denote this Boolean inner product of vectors u and v by $u \bullet v$ and extend this notation to Boolean matrix-vector product and Boolean matrix multiplication. For $u, v \in \{0, 1\}^n$, $u \bullet v = 1$ if and only if the subsets of $[n]$ encoded by u and v intersect, so the problems of computing Boolean matrix multiplication and Boolean matrix-vector product can be seen as computing many correlated copies of the set disjointness problem.

5.1 Tradeoffs for Boolean matrix multiplication

Unlike what we have shown for algebraic problems, quantum algorithms for Boolean matrix multiplication have better time-space tradeoff properties than their classical counterparts.

Proposition 5.1. *For any $c > 0$, there are quantum circuits computing $n \times n$ Boolean matrix multiplication $A \bullet B$ with error at most n^{-c} using space $O(\log n)$ and a number of queries T that is $O(n^{2.5}\log n)$.*

Proof. Fix $c > 0$. Each of the n^2 entries in the product is a disjointness function of length n that can be computed with error at most n^{-c-2} and space $O(\log n)$ using Grover's algorithm in time $O(\sqrt{n}\log n)$ for error at most n^{-c} overall. □

This is in contrast to the following result of Abrahamson which shows that classical algorithms as fast as this quantum algorithm require space $\tilde{\Omega}(n^{0.5})$ rather than $O(\log n)$.

Proposition 5.2 ([Abr90])**.** *There is a probability distribution on input matrices and constants $0 < c_1 < c_2$ under which the best classical algorithms (branching programs) for Boolean matrix multiplication $A \bullet B$ using time T and space S require $T \cdot S$ that is* $\begin{cases} \Theta(n^{3.5}) & \text{for } T \le c_1 n^{2.5} \\ \Theta(n^3) & \text{for } T \ge c_2 n^{2.5}. \end{cases}$

For quantum circuits, Klauck, Špalek, and de Wolf [KŠdW07] proved the following time-space tradeoff lower bound which proves that the quantum algorithm in Proposition 5.1 is nearly optimal when the space S is $O(\log n)$.

Proposition 5.3 (Theorem 25 in [KŠdW07])**.** *Any bounded error quantum circuit that computes the $n \times n$ Boolean matrix multiplication $A \bullet B$ with T queries and space S requires $T^2 S$ to be $\Omega(n^5)$, or equivalently that T is $\Omega(n^{2.5}/S^{0.5})$.*

A key difference between the methods used in Abrahamson's bounds and those in this proof is that for quantum (and classical) circuits, unlike the case for branching programs, it is reasonable to assume that the set of output values produced in each part of the computation is fixed independent of the input. Such an assumption was essential for the time-space lower bounds in [KŠdW07, AŠdW09], although the bound for multiple disjoint collision pairs in [HM21] and our results in Sections 3 and 4 apply to quantum circuits without such a restriction on output production. Fixing the output values produced in each part of the computation allows one to go beyond using a single hard distribution on inputs, and instead choose hard distributions for each part of the computation depending on the target outputs. To give a sense of how this works we sketch the lower bound method of [KŠdW07] for boolean matrix multiplication, which relies on a strong direct product lemma for the function OR_n^k:

Proposition 5.4 (Strong Direct Product Theorem for OR_n^k [KŠdW07])**.** *There are positive constants ε and γ such that the following hold:*

(a) Any randomized algorithm making at most εkn queries has success probability at most $2^{-\gamma k}$ in computing OR_n^k.

(b) Any quantum algorithm making at most $\varepsilon k\sqrt{n}$ queries has success probability at most $2^{-\gamma k}$ in computing OR_n^k.

Proof Sketch for Proposition 5.3. Without loss of generality S is $o(n)$ since the number of queries must be $\Omega(n^2)$ in the worst case.[4] The circuit is sliced into layers of height $\alpha\sqrt{Sn}$ which is $o(n)$. The claim is that such a slice can produce $k = O(S)$ outputs with probability only exponentially small

[4]Note that this is not completely obvious since quantum algorithms for some problems may have a sublinear numbers of queries.

in k: The idea is simply to partition $[n]$ into k blocks of size n/k and associate each output position (i,j) in the product with one of those blocks. For each i, matrix A will have 1's in block (i,j) of row i for each j such that (i,j) is an output in the slice and will be 0 elsewhere. For each j, matrix B will be 0 in column j for all blocks (i',j') that are not associated with some block (i,j) for some $i \in [n]$. The k outputs (i,j) will then be the result of k disjoint OR computations on n/k-bit blocks of B. By the strong direct product theorem for OR Proposition 5.4 for ε a sufficiently small constant, any circuit of height at most $\varepsilon k\sqrt{n/k} = \varepsilon\sqrt{kn}$ is correct with probability at most $2^{-\gamma k}$ for some constant $\gamma > 0$. There are n^2 outputs in total, so for $k = \Theta(S)$ there must be $\Omega(n^2/S)$ slices, each of $\Omega(\sqrt{Sn})$ queries so the total time is $\Omega(n^2\sqrt{n/S})$ and hence T^2S is $\Omega(n^5)$. □

Our improved lower bound

Theorem 5.5. *Any quantum circuit computing $n \times n$ Boolean matrix multiplication $A \bullet B$ with T queries and space S and success probability more than 2^{-S} must have T that is $\Omega(n^{2.5}/S^{1/3})$.*

This theorem follows from the following key lemma which improves on the corresponding bound in [KŠdW07] by a factor of $\Theta(k^{1/6})$.

Lemma 5.6. *There are constants $c, c' > 0$ such that the following holds. Let $k \leq (n/\sqrt{3/2})^{3/2}/4$ be an integer. For any quantum circuit $\mathcal{C}$ with at most $c'k^{2/3}n^{1/2}$ queries to x, the probability that $\mathcal{C}$ produces k correct output values of $n \times n$ Boolean matrix multiplication $A \bullet B$ is at most $2^{-c'k}$.*

The main idea behind the proof of this key lemma is an improved method for embedding the direct product of *OR* functions into outputs of the Boolean matrix multiplication problem. This is based on the following definition of an L-coloring of subsets of $[n] \times [n]$.

Definition 5.7. For $E \subseteq [n] \times [n]$ an *L-coloring* of E is a map $\chi : E \to [L]$ such that

- within each color class either all rows are distinct or all columns are distinct, and
- for every $(i,j) \in E$, and every color ℓ, there do not exist $i' \neq i$ and $j' \neq j$ such that both (i,j') and (i',j) have color ℓ.

The second condition is equivalent to saying that for each color class ℓ there is a rectangle given by sets $R_\ell \subseteq [n]$ of rows and $C_\ell \subseteq [n]$ of columns such that all points in $E \cap (R_\ell \times C_\ell)$ have color ℓ. (Note that these rectangles may overlap, but their overlap must not contain any points in E.)

The motivation for this definition is given by the following lemma.

Lemma 5.8. *Let $E \subseteq [n] \times [n]$ with $|E| = k$ and $L \leq n$ be an integer with $L \leq n/2$. If E has an L-coloring then $OR^k_{\lfloor n/L \rfloor}$ is a sub-function of the function that produces the k outputs of $A \bullet B$ indexed by E for $n \times n$ Boolean matrices A and B.*

Proof. Write $E = \dot{\bigcup}_{\ell=1}^{L} E_\ell$ where E_ℓ is the set of (i,j) in E in color class ℓ. We now divide $[n]$ into L disjoint blocks $b_1, \ldots, b_L$ of at least $\lfloor n/L \rfloor \geq 2$ elements each. Given the coloring and division into blocks, we define a partial assignment to the matrices A and B as follows:

- If color class ℓ consists of pairs (i,j) that do not share a column, for each i such that $(i,j) \in E_\ell$, we set row i of matrix A to be 1 in all entries in b_ℓ and leave the entries of $B_{b_\ell,j}$ to be unset for every j for which there is some element in E_ℓ with column j.

- If color class ℓ consists of pairs (i,j) that do not share a row, for each j such that $(i,j) \in E_\ell$, we set column j of matrix B to be 1 in all entries in b_ℓ and leave the entries of A_{i,b_ℓ} to be unset for every i for which there is some element in E_ℓ with row i.

- All entries of A and B that are not defined by the above two cases are set to 0.

In particular, this means that if E_ℓ does not contain any element of the form $(i,\cdot)$ then the submatrix A_{i,b_ℓ} is all 0 and if E_ℓ does not contain any element of the form $(\cdot,j)$ then the submatrix $B_{b_\ell,j}$ is all 0.

It remains to show that the outputs in E of this matrix product are k disjoint ORs on at least $\lfloor n/L \rfloor$ bits each. Observe that for $(i,j) \in E$, the (i,j) entry of the product $A \bullet B$ will be 0 unless row i and column j both contain elements of some color class ℓ. There can only one such color ℓ, namely the color of (i,j), since the second condition for the coloring rules out any other color class.

If color class ℓ has elements that don't share a column then, by construction, the output for each is the OR of the n/L unrestricted elements of $B_{b_\ell,j}$. Since the elements of E_ℓ don't share a column, given ℓ and j there is a unique i such that $(i,j) \in E_\ell$. Therefore the vectors $B_{b_\ell,j}$ for all color classes ℓ of this type and all j with some $(\cdot,j) \in E_\ell$ are disjoint. The analogous properties apply to the color classes ℓ that don't share rows. In that case, each output $(i,j) \in E_\ell$ is the OR of the $\lfloor n/L \rfloor$ elements in some unrestricted vector A_{i,b_ℓ} and the vectors A_{i,b_ℓ} are disjoint from each other.

The net result is that the k outputs in E are k disjoint ORs on at least $\lfloor n/L \rfloor$ unrestricted bits each as required. □

The lower bound of [KŠdW07] is derived by showing that $OR^k_{\lfloor n/k \rfloor}$ can be embedded into any set E of k outputs of $A \bullet B$. Their argument corresponds to the trivial k-coloring that assigns each element of E to its own color class.

Definition 5.9. For integer $k > 0$ define $L(k)$ to be the minimum number of colors L such that for all subsets $E \subseteq [n] \times [n]$ with $|E| \le k$, there is an L-coloring of E.

Lemma 5.10. *There are constants $c, c' > 0$ such that the following holds. Let k be an integer such that $L(k) \le n/2$. For any quantum circuit $\mathcal{C}$ with at most $ckn^{1/2}/L(k)^{1/2}$ queries to x, the probability that $\mathcal{C}$ produces k correct output values of $n \times n$ Boolean matrix product $A \bullet B$ is at most $2^{-c'k}$.*

Proof. Let E be any fixed set of k output positions in $A \bullet B$. States with different choices of E are orthogonal to each other so we show that for each fixed value of E the probability that the algorithm is correct has the given probability bound. Let $L \le L(k)$ be such that there is an L-coloring of E. By Lemma 5.8, $OR^k_{\lfloor n/L \rfloor}$ is a sub-function of the k outputs indexed by the set E. Since $L \le n/2$, $\lfloor n/L \rfloor \ge 2n/(3L)$ and $\sqrt{\lfloor n/L \rfloor} \ge 4\sqrt{n/L}/5$. Choose $c = 4\varepsilon/5$ and $c' = \gamma$ for ε and γ given in Proposition 5.4. By that proposition, the probability that $\mathcal{C}$ produces these k outputs correctly is at most $2^{-\gamma k} = 2^{-c'k}$. □

Then Lemma 5.6 is an immediate corollary of Lemma 5.10 and the following bound on $L(k)$.

Lemma 5.11. $\sqrt{2k} \le L(k) \le \sqrt{3/2}\, k^{2/3}$.

Proof. The lower bound follows from the case that E consists of the diagonal and the upper-triangular portion of a grid with side L which contains $k = L(L+1)/2$ points which has two trivial L-colorings consisting of either the rows and the columns. The second condition for a coloring ensures that the diagonal must all be in different classes since they cannot share a row or column, so those trivial colorings are optimal.

To prove the upper bound we use the following definition: We say that i is a *row of E* iff E contains (i, j) for some j; similarly, we say that j is a *column of E* iff E contains (i, j) for some i. We say that two rows of $E \subseteq [n] \times [n]$ indexed by i and i' intersect iff there is some column j such that (i, j) and (i', j) are both in E. The definition is analogous for intersecting columns.

Claim: If there is an $E \subseteq [n] \times [n]$ that requires at least L colors, then there is an E' requiring L colors with $|E'| = |E|$ such that every pair of rows and columns of E intersect.

Proof of Claim. Suppose that rows i and i' of E do not intersect. Define E' to be the same as E except that we replace every $(i', j) \in E$ in E' with (i, j). Any coloring of E' can be converted to a coloring of E by coloring $(i', j) \in E$ with the color given to (i, j) in E'. Any row-oriented color set will have the same set of columns, any column-oriented color set in E' containing (i, j) will remain column-oriented with row i replaced by row i'. Finally, if the last condition is violated for the coloring of E then it would have been violated for the coloring of E'. The analogous property holds for columns of E that do not intersect.

Repeat the above procedure until there are no pairs of non-intersecting rows or columns. □

We now prove the coloring bound. Assume without loss of generality by the claim that every pair of rows and columns of E is intersecting. Define R to be the number of rows of E and assume without loss of generality that $R > \sqrt{3/2}\, k^{2/3}$; otherwise we are done.

This means that if we have a row or column with at least $r \ge (3/2)k^{1/3}$ elements of E then we can include a set for that row or column and apply induction to color the remaining $k' = k - r$ elements of E'. By induction there will be at most $\sqrt{3/2}\,(k')^{2/3} = \sqrt{3/2}\,(k-r)^{2/3}$ colors needed to color E'. Now

$$
\begin{aligned}
(k-r)^2 &\le (k - (3/2)k^{1/3})^2 = k^2 - 3k^{4/3} + (9/4)k^{2/3} \\
&\le k^2 - 3k^{4/3} + 3k^{2/3} - 1 \qquad \text{since } (3/4)k^{2/3} \ge 1 \text{ for } k \ge 2 \\
&= (k^{2/3} - 1)^3.
\end{aligned}
$$

Therefore, at most $\sqrt{3/2}(k^{2/3} - 1) < \sqrt{3/2}\, k^{2/3} - 1$ colors are needed to color E' and hence at most $\sqrt{3/2}\, k^{2/3}$ colors needed to color E.

Since $R > \sqrt{3/2}\, k^{2/3}$ and there are only k elements of E in total, there must be some row with $c < k^{1/3}/\sqrt{2/3}$ elements of E. Since $R > \sqrt{3/2}\, k^{2/3}$, the c columns of these elements must be used to intersect each of the $R - 1$ other rows and hence there must be one of these columns that is in at least $(R-1)/c > \frac{\sqrt{3/2}\, k^{2/3}}{k^{1/3}/\sqrt{3/2}} - 1 = (3/2)k^{1/3} - 1$ other rows. In other words, it has length at least $(3/2)k^{1/3}$ and hence we can apply induction to finish the proof.

Therefore we have shown that the number of colors $L(k)$ is at most $\sqrt{3/2}\, k^{2/3}$. □

Theorem 5.5 is a immediate corollary of the above bound on $L(k)$ and the following theorem.

Theorem 5.12. *Any quantum circuit computing $n \times n$ Boolean matrix multiplication $A \bullet B$ with T queries and space S and success probability more than 2^{-S} must have T that is $\Omega(n^{2.5}/L(S)^{0.5})$.*

Proof. Since there are n^2 outputs we trivially have $T \geq n^2$ so we can assume that $L(S) < \alpha n$ for some arbitrarily small constant $\alpha > 0$ without loss of generality. Let ε and γ be the constants from Lemma 5.10. Let $c = 3/(2\gamma)$ and define $k = cS$. Since the function L is bounded by concave functions we have $L(k) \leq c\ L(S)$ and hence we can assume that $L(k) = L(cS) \leq 3n/4$. By Lemma 5.10, since $L(k) \leq 3n/4$ any quantum query algorithm with at most $\varepsilon k\sqrt{n/L(k)}$ queries has success probability at most $2^{-\gamma k} = 2^{-3S}$ of producing k correct outputs. If $T \leq \varepsilon n^2\sqrt{n/L(S)}/\sqrt{c} \leq \varepsilon n^2\sqrt{n/L(k)}$, when we divide $\mathcal{C}$ into layers with at most $\varepsilon k\sqrt{n/L(k)}$ quantum queries, there are at most n^2/k layers. Since there are a total of n^2 outputs there must be some layer i during which at least k outputs are produced. Let E be the set of the first k outputs produced in layer i. Since the space is at most S, by Proposition 2.5 the probability that these k outputs are correct given S qubits of input dependent initial state is at most 2^{-S}, which proves the lower bound. $\square$

We also obtain a general classical lower bound from these arguments.

Theorem 5.13. *Any classical circuit computing $n \times n$ Boolean matrix-multiplication with T queries and space S with success probability more than 2^{-S} must have T that is $\Omega(n^3/L(S))$.*

Proof. Again since there are n^2 outputs which is a trivial time lower bound we can assume that $L(S)$ is at most αn for some arbitrarily small constant $\alpha > 0$. Let $c = 2/\gamma$ for γ given by Proposition 5.4 and let $k = cS$. Again we can assume that $L(k) = L(cS) \leq n/2$ The main difference in parameters from the quantum case is that by applying Lemma 5.8 and Proposition 5.4, classical circuits of width S such that $L(k) \leq n/2$ and depth $2\varepsilon kn/(3L(k))$ computing k disjoint ORs of size $\lfloor n/L(k)\rfloor \geq n/(2\,L(k))$ have success probability at most $2^{-\gamma k} = 2^{-2S}$ probability of computing k correct outputs. There are at most 2^S choices of the values of the gates at a layer boundary and hence the probability that a layer of height $\varepsilon kn/L(k)$ correctly produces k correct outputs is at most 2^{-S}. Since there are n^2 outputs, any circuit of depth T at most $\varepsilon n^3/L(k)$ must have some layer of depth $\varepsilon kn/L(k)$ during which at most k outputs are produce and each must be correct, so the overall success probability is at most 2^{-S}. $\square$

Using our upper bound on $L(k)$ we obtain the following bound.

Corollary 5.14. *Any classical circuit computing $n \times n$ Boolean matrix-multiplication with T queries and space S with success probability more than 2^{-S} must have T that is $\Omega(n^3/S^{2/3})$.*

Note that this bound is larger than the bound of Abrahamson [Abr90] when S is $\omega(1)$ and $o(n^{0.5})$ since in that case Abrahamson's query lower bound is $\Omega(n^3/S)$. Of course Abrahamson's lower bound is for the branching program model which allows for output orders that depend on the input. (The classical lower bound of [KŠdW07] for circuits is exactly the same as that of Abrahamson in this range of space bounds.) Note also that $S = \Theta(n^{3/2})$ is the precise point at which both this bound and Abrahamson's bound on the number of queries become the trivial $\Theta(n^2)$ and, in particular, this is tight for the distribution used in Abrahamson's paper.

Using the same proof idea as in Corollary 4.5, the bounds in Theorem 5.5 and Corollary 5.14 immediately imply lower bounds for boolean matrix squaring.

Corollary 5.15. *Any quantum circuit computing $n \times n$ Boolean matrix squaring on all inputs with T queries, space S, and success probability more than 2^{-S} must have T that is $\Omega(n^{2.5}/S^{1/3})$. Any such classical circuit must have T that is $\Omega(n^3/S^{2/3})$.*

5.2 Boolean matrix-vector product

Though [Abr90] does not contain an explicit theorem statement on time-space tradeoffs for Boolean matrix-vector products that is the analog of the linear algebra bound in [Abr91] or our Theorem 3.5, [Abr90] contains the claim that analogous results do indeed hold for this problem using the same ideas. (The bound would be a factor n smaller lower bound.)

For quantum circuits, Klauck, Špalek, and de Wolf [KŠdW07] prove the following results for computing Boolean matrix-vector products. (They prove a similar result for the case of classical circuits also, though that does not apply to branching programs, which can vary the output order depending on the input values.)

Proposition 5.16 (Theorem 23 in [KŠdW07])**.** *For every S in $o(n/\log n)$, there is an $n \times n$ Boolean matrix $A^{(S)}$ such that every bounded-error quantum circuit with space at most S that computes Boolean matrix-vector product $A^{(S)} \bullet x$ in T queries requires that T is $\Omega(\sqrt{n^3/S}) = \Omega(n^{1.5}/S^{0.5})$.*

This result is weaker than a standard time-space tradeoff since the function involved is not independent of the circuits that might compute it. In particular, [KŠdW07] does not find a single function that is hard for all space bounds, as the matrix $A^{(S)}$ that they use changes depending on the value of S.

For $S = o(n/\log n)$, the matrix $A^{(S)}$ is produced via the probabilistic method using the following distribution: Choose k to be a sufficiently large constant multiple of S. This distribution chooses matrices $A \subseteq \{0,1\}^{n \times n}$ by selecting a uniformly random subset of $n/(2k)$ positions in each row to set to 1, with the remainder of the entries in each row being 0. They show that with positive probability over the choice of A, for all sets $I \subseteq [n]$ of size k, at least $k/2$ of the rows of A_I contain at least $N/(6k)$ 1's that are unique in their column of A_I; that is, those columns are 0 in all of the $k-1$ other rows of A_I. $A_{(S)}$ is then some fixed matrix for which this property is true.

More precisely, when we fix a row $j \in I$ and the $n/(2k)$ columns where it is 1, the expected number of the $(k-1)n/(2k) < n/2$ 1's among the rows in $I \setminus \{j\}$ that land in those $n/(2k)$ columns is less than $n/(4k)$. By a Hoeffding bound, the number of those 1's is at most $n/(3k)$ except with probability exponentially small in n/k, which is $n^{-\omega(1)}$ since $k = O(S) = o(n/\log n)$. Hence, except with probability $n^{-\omega(1)}$, a row $j \in I$ is *good for I* in that at least $n/(2k) - n/(3k) = n/(6k)$ of the 1's in row j are unique in their respective columns in A_I. For a fixed I, the probability that there is no $J \subseteq I$ of size $k/2$ all of whose rows are good for I is less than the probability that there are $k/2$ rows of I that are not good for I. This happens with probability at most $n^{-\omega(k)}$ since are at most $\binom{k}{k/2}$ such subsets of rows of size $k/2$, each of which is not good for I with probability $n^{-\omega(k)}$ (and the probabilities are negatively associated). Since there are only $\binom{n}{k}$ choices of I, the total probability that A does not have desired properties is only $n^{-\omega(k)}$.

The proof of Proposition 5.16 follows from the usual time-space lower bound methodology and the following lemma:

Lemma 5.17. *There is an $\alpha > 0$ such for every quantum circuit $\mathcal{C}$ that makes at most $\alpha\sqrt{kn}$ queries to $x \in \{0,1\}^n$, the probability that $\mathcal{C}$ produces at least k correct outputs of $A^{(S)} \bullet x$ is at most $2^{-\Omega(k)}$.*

Proof. Let $I \subseteq [n]$ be the set of indices of the first k outputs of $A^{(S)} \bullet x$ produced by $\mathcal{C}$. Let $J \subseteq I$ be the set of size $k/2$ rows that are good for I guaranteed by the properties of $A^{(S)}$. We show that the probability that $\mathcal{C}$ produces all outputs even for the rows in J is exponentially small in k: For each row $j \in J$ there is a set C_j of $n/(6k)$ columns of $A_I^{(S)}$ where the unique 1 is in row j. Consider the restriction to input vectors $x \in \{0,1\}^n$ that are 0 outside of $\bigcup_{j\in J} C_j$. Then the outputs for $j \in J$ are a direct product of $k/2$ OR functions of size $n/(6k)$ on the bits of $\bigcup_{j\in J} C_j$. By a strong direct product theorem for OR (Theorem 14 of [KŠdW07]), for ε a sufficiently small constant, any circuit of height at most $\varepsilon(k/2)\sqrt{n/(6k)} = \varepsilon\sqrt{kn/24}$ is correct with probability at most $2^{-\gamma k}$ for some constant $\gamma > 0$. □

On the algorithmic side, we have the following:

Proposition 5.18. *For every $c > 0$ and every Boolean matrix $A \in \{0,1\}^{m\times n}$ there is a quantum circuit using space $O(\log n)$ and time $O(mn^{1/2}\log m)$ that computes Boolean matrix-vector product $A \bullet x$ with error at most m^{-c}. More precisely, the algorithm runs in time $O(|A|_{1/2}\log m)$ where $|A|_{1/2} = \sum_{i=1}^m \sqrt{|A_i|_1}$.*

Proof. For each row in turn, run Grover's algorithm to compute the OR of the bits indexed by the 1's of A_i, the i-th row of A with probability of error at most m^{-c-1} per row for a total error of at most m^{-c}. □

We note that for the fixed matrix $A^{(S)}$, each row has $\Theta(n/S)$ 1's so $|A^{(S)}|_{1/2} = \Theta(n^{3/2}/S^{1/2})$. This is an odd situation in that the matrix $A^{(S)}$ designed to require large time for space S algorithms can be solved in nearly the same time bound by space $O(\log n)$ algorithms.

On the other hand, consider the following space S algorithm that works for all inputs x with Hamming weight $|x|_1 \leq S/\log n$: Run Grover's algorithm $O(S)$ times to find and record the locations of all $O(S/\log n)$ 1's in input string x. This takes $O(\sqrt{Sn/\log n})$ queries. Then compute the m entries of $A \bullet x$, one after another, which doesn't require any additional queries. Note that this is always more efficient than $m\sqrt{n/S}$ queries.

Systems of linear inequalities The same space dependent matrix $A^{(S)}$ in Proposition 5.16 was also used in [AŠdW09] for systems of inequalities.

Proposition 5.19 (Theorem 11 in [AŠdW09])**.** *Let $\vec{t}$ be the length n all-t vector. For every S in $\min(O(n/t), o(n/\log n))$ there exists an $n \times n$ Boolean matrix A^S such that every bounded error quantum circuit with space at most S that decides the system $Ax \geq \vec{t}$ of n inequalities requires that T is $\Omega(\sqrt{tn^3/S})$.*

Similar to [KŠdW07] this matrix is used so that any quantum circuit that computes $Ax \geq \vec{t}$ can be broken down into slices that solve independent instances of the t-threshold function.

Our results

Using Proposition 5.16, we can obtain a time-space tradeoff lower bound for quantum computation of Boolean matrix-vector product that has a only slightly lower weaker bound in terms of the matrix dimensions but, unlike the previous bound, defines a fixed computational problem whose definition is independent of the space bound allowed.

Theorem 5.20. *There is a fixed $m \times n$ Boolean matrix A with $m \leq n \log_2 n$ such that for every S that is $o(n/\log n)$ every bounded-error quantum circuit with space at most S that computes Boolean matrix-vector product $A \bullet x$ in T queries requires that T is $\Omega(\sqrt{n^3/S})$.*

Proof. The matrix A consists of a stacked version of the matrices $A_{(S_i)}$ from Proposition 5.16 for each choice of $S_i = 2^i \log_2 n$ and $0 \leq i \leq \log_2 n - 2\log_2 \log_2 n - \omega(1)$. Any quantum circuit computing $A \bullet x$ using space S must compute $A^{(S_i)} \bullet x$ for some S_i where $S_i \leq S$ is within factor of 2 of S. It is easy to see that the construction of $A_{(S)}$ for Proposition 5.16 is flexible in terms of the constant factor by which k exceeds S and hence computing matrix $A^{(S_i)} \bullet x$ also requires time T that is $\Omega(\sqrt{n^3/S})$ as required. □

Systems of linear inequalities This same matrix A can be substituted into Proposition 5.19 to obtain a time-space tradeoff for systems of inequalities.

Corollary 5.21. *Let $\vec{t}$ be the length n all-t vector. There is a fixed $m \times n$ Boolean matrix A with $m \leq n \log_2 n$ such that for every S in $\min(O(n/t), o(n/\log n))$ every bounded error quantum circuit with space at most S that decides the system $Ax \geq \vec{t}$ requires T that is $\Omega(\sqrt{tn^3/S})$.*

References

[Aar05] Scott Aaronson. Limitations of quantum advice and one-way communication. *Theory of Computing*, 1(1):1–28, 2005.

[Abr90] Karl R. Abrahamson. A time-space tradeoff for Boolean matrix multiplication. In *31st Annual Symposium on Foundations of Computer Science, St. Louis, Missouri, USA, October 22-24, 1990, Volume I*, pages 412–419. IEEE Computer Society, 1990.

[Abr91] Karl R. Abrahamson. Time-space tradeoffs for algebraic problems on general sequential machines. *Journal of Computer and System Sciences*, 43(2):269–289, 1991.

[Amb02] Andris Ambainis. Quantum lower bounds by quantum arguments. *Journal of Computer and System Sciences*, 64(4):750–767, 2002.

[AŠdW09] Andris Ambainis, Robert Špalek, and Ronald de Wolf. A new quantum lower bound method, with applications to direct product theorems and time-space tradeoffs. *Algorithmica*, 55(3):422–461, Nov 2009.

[BBC+01] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *J. Acm*, 48(4):778–797, jul 2001.

[BC82] Allan Borodin and Stephen A. Cook. A time-space tradeoff for sorting on a general sequential model of computation. *SIAM J. Comput.*, 11(2):287–297, 1982.

[Bea91] Paul Beame. A general sequential time-space tradeoff for finding unique elements. *SIAM J. Comput.*, 20(2):270–277, 1991.

[BFK+79] Allan Borodin, Michael J. Fischer, David G. Kirkpatrick, Nancy A. Lynch, and Martin Tompa. A time-space tradeoff for sorting on non-oblivious machines. In *20th Annual Symposium on Foundations of Computer Science (sfcs 1979)*, pages 319–327, 1979.

[BK23] Paul Beame and Niels Kornerup. Cumulative Memory Lower Bounds for Randomized and Quantum Computation. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 17:1–17:20, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[BT23] Ainesh Bakshi and Ewin Tang. An improved classical singular value transformation for quantum machine learning, 2023.

[BV97] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 1997.

[CCH+22] Nadiia Chepurko, Kenneth L. Clarkson, Lior Horesh, Honghao Lin, and David P. Woodruff. Quantum-inspired algorithms from randomized numerical linear algebra. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 3879–3900. PMLR, 2022.

[CGL+20a] Nai-Hui Chia, András Gilyén, Tongyang Li, Han-Hsuan Lin, Ewin Tang, and Chunhao Wang. Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2020, page 387–400, New York, NY, USA, 2020. Association for Computing Machinery.

[CGL+20b] Nai-Hui Chia, András Gilyén, Han-Hsuan Lin, Seth Lloyd, Ewin Tang, and Chunhao Wang. Quantum-Inspired Algorithms for Solving Low-Rank Linear Equation Systems with Logarithmic Dependence on the Dimension. In Yixin Cao, Siu-Wing Cheng, and Minming Li, editors, *31st International Symposium on Algorithms and Computation (ISAAC 2020)*, volume 181 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 47:1–47:17, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[CKS15] Andrew M. Childs, Robin Kothari, and Rolando D. Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM J. Comput.*, 46:1920–1950, 2015.

[DJ92] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London*, 1992.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery.

[GSLW19] 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*, STOC 2019, page 193–204, New York, NY, USA, 2019. Association for Computing Machinery.

[GST22] András Gilyén, Zhao Song, and Ewin Tang. An improved quantum-inspired algorithm for linear regression. *Quantum*, 6:754, jun 2022.

[HHL09] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15), oct 2009.

[HLS22] Yassine Hamoudi, Qipeng Liu, and Makrand Sinha. Quantum-classical tradeoffs in the random oracle model. *CoRR*, abs/2211.12954, 2022.

[HM21] Yassine Hamoudi and Frédéric Magniez. Quantum time-space tradeoff for finding multiple collision pairs. In Min-Hsiu Hsieh, editor, *16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021)*, volume 197 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 1:1–1:21, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[KŠdW07] Hartmut Klauck, Robert Špalek, and Ronald de Wolf. Quantum and classical strong direct product theorems and optimal time-space tradeoffs. *SIAM Journal on Computing*, 36(5):1472–1493, 2007.

[LC19] Guang Hao Low and Isaac L. Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3:163, jul 2019.

[LZ18] Qipeng Liu and Mark Zhandry. On finding quantum multi-collisions. *CoRR*, abs/1811.05385, 2018.

[MNT93] Yishay Mansour, Noam Nisan, and Prasoon Tiwari. The computational complexity of universal hashing. *Theor. Comput. Sci.*, 107(1):121–133, 1993.

[Ros21] Ansis Rosmanis. Tight bounds for inverting permutations via compressed oracle arguments. *CoRR*, abs/2103.08975, 2021.

[She11] Alexander A. Sherstov. Strong direct product theorems for quantum communication and query complexity. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, Stoc '11, page 41–50, New York, NY, USA, 2011. Association for Computing Machinery.

[She12] Alexander A. Sherstov. Strong direct product theorems for quantum communication and query complexity. *SIAM J. Comput.*, 41(5):1122–1165, 2012.

[Sim97] Daniel Simon. On the power of quantum computation. *SIAM Journal on Computing*, 97.

[Špa08] Robert Špalek. The multiplicative quantum adversary. In *2008 23rd Annual IEEE Conference on Computational Complexity*, pages 237–248, 2008.

[ŠS05] Robert Špalek and Mario Szegedy. All quantum adversary methods are equivalent. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *Automata, Languages and Programming*, pages 1299–1311, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[Tan18] Ewin Tang. A quantum-inspired classical algorithm for recommendation systems. *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, 2018.

[Tom78] Martin Tompa. Time-space tradeoffs for computing functions, using connectivity properties of their circuits. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC '78, page 196–204, New York, NY, USA, 1978. Association for Computing Machinery.

[Yes84] Yaacov Yesha. Time-space tradeoffs for matrix multiplication and the discrete Fourier transform on any general sequential random-access computer. *Journal of Computer and System Sciences*, 29(2):183–197, 1984.

[Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 239–268, Cham, 2019. Springer International Publishing.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il