# A New Quantum Lower Bound Method, with Applications to Direct Product Theorems and Time-Space Tradeoffs

Andris Ambainis*
University of Waterloo
ambainis@math.uwaterloo.ca

Robert Špalek†
CWI, Amsterdam
sr@cwi.nl

Ronald de Wolf‡
CWI, Amsterdam
rdewolf@cwi.nl

## ABSTRACT

We give a new version of the adversary method for proving lower bounds on quantum query algorithms. The new method is based on analyzing the eigenspace structure of the problem at hand. We use it to prove a new and optimal strong direct product theorem for 2-sided error quantum algorithms computing $k$ independent instances of a symmetric Boolean function: if the algorithm uses significantly less than $k$ times the number of queries needed for one instance of the function, then its success probability is exponentially small in $k$. We also use the polynomial method to prove a direct product theorem for 1-sided error algorithms for $k$ threshold functions with a stronger bound on the success probability. Finally, we present a quantum algorithm for evaluating solutions to systems of linear inequalities, and use our direct product theorems to show that the time-space tradeoff of this algorithm is close to optimal.

## Categories and Subject Descriptors

F.1.2 [**Computation by Abstract Devices**]: Modes of Computation; F.1.3 [**Computation by Abstract Devices**]: Complexity Measures and Classes—*Relations among complexity measures*; F.2.3 [**Analysis of Algorithms and Problem Complexity**]: Tradeoffs between Complexity Measures

## General Terms

Algorithms, Theory

*Institute for Quantum Computing and Department of Combinatorics and Optimization, University of Waterloo. Supported by NSERC, ARO, CIAR and IQC University Professorship.

†Supported in part by the European Commission under projects RESQ, IST-2001-37559, and QAP, IST-015848.

‡Supported by a Veni grant from the Netherlands Organization for Scientific Research (NWO) and partially supported by the EU projects RESQ and QAP.


*STOC'06,* May 21–23, 2006, Seattle, Washington, USA.


## Keywords

quantum computing, lower bounds, direct product theorems, time-space tradeoffs

## 1. INTRODUCTION

### 1.1 A new adversary method

Most of the known quantum algorithms work in the black-box model of computation. Here one accesses the $n$-bit input via *queries* and our measure of complexity is the number of queries made by the algorithm. In between the queries, the algorithm can make unitary transformations for free. This model includes for instance the algorithms of Grover, Deutsch and Jozsa, Simon, quantum counting, the recent quantum walk-based algorithms, and even Shor's period-finding algorithm (which is the quantum core of his factoring algorithm).

Much work has focused on proving *lower bounds* in this model. The two main methods known are the polynomial method and the adversary method. The polynomial method [24, 8] works by lower-bounding the degree of a polynomial that in some way represents the desired success probability.

The adversary method was originally introduced by Ambainis [3]. Many different versions have since been given [19, 7, 4, 21, 30], but they are all equivalent [29]. Roughly speaking, the adversary method works as follows. Suppose we have a $T$-query quantum algorithm that computes some function $f$ with high success probability. Let $|\psi_x^t\rangle$ denote the algorithm's state on input $x$ after making the $t$-th query. Suppose $x$ and $y$ are two inputs with distinct function values. At the start of the algorithm ($t = 0$), the states $|\psi_x^0\rangle$ and $|\psi_y^0\rangle$ are the same (the input has not been queried yet), so their inner product is $\langle\psi_x^0|\psi_y^0\rangle = 1$. But at the end of the algorithm ($t = T$), the inner product $\langle\psi_x^T|\psi_y^T\rangle$ must be less than some small constant depending on the error probability, otherwise the algorithm cannot give the correct answer for both $x$ and $y$. The adversary method takes a (weighted) sum of such inner products (for $x, y$ pairs with $f(x) \neq f(y)$) and analyzes how quickly this sum can go down after each new query. If it cannot decrease quickly in one step, then it follows that we need many steps and we obtain a lower bound on $T$.

The two lower bound methods are incomparable. On the one hand, the adversary method proves stronger bounds than the polynomial method for certain iterated functions [4], and also gives tight lower bounds for constant-depth AND-OR trees [3, 18], where we do not know how to analyze the polynomial degree. On the other hand, the polyno-

mial method works well for analyzing zero-error or low-error quantum algorithms [8, 12] and gives optimal lower bounds for the collision problem and element distinctness [1]. The adversary method fails for the latter problem (and also for other problems like triangle-finding), because the best bound provable with it is $O(\sqrt{C^0(f)C^1(f)})$ [29, 30]. Here $C^0(f)$ and $C^1(f)$ are the certificate complexities of $f$ on 0-inputs and 1-inputs. In the case of element distinctness and triangle-finding, one of these complexities is constant. Hence the adversary method in its present form(s) can prove at most an $\Omega(\sqrt{N})$ bound, while the true bound is $\Theta(N^{2/3})$ [5] in the case of element distinctness and the best known algorithm for triangle-finding costs $O(N^{13/20})$ [22]. A second limitation of the adversary method is that it cannot deal well with the case where there are many different possible outputs, and a success probability much smaller than 1/2 would still be considered good.

In this paper we describe a new version of the adversary method that does not suffer from the second limitation, and possibly also not from the first—though we have not found an example yet where the new method breaks through the $\sqrt{C^0(f)C^1(f)}$ barrier.

*Very* roughly speaking, the new method works as follows. We view the algorithm as acting on a 2-register state space $\mathcal{H}_A \otimes \mathcal{H}_I$. Here the actual algorithm's operations take place in the first register, while the second contains (a superposition of) the inputs. In particular, the query operation on $\mathcal{H}_A$ is now conditioned on the basis states in $\mathcal{H}_I$. We start the analysis with a superposition of 0-inputs and 1-inputs in the input register, and then track how this register evolves as the computation moves along. Let $\rho_t$ be the state of this register (tracing out the $\mathcal{H}_A$-register) after making the $t$-th query. By employing symmetries in the problem's structure, such as invariances of the function under certain permutations of its input, we can decompose the input space into orthogonal subspaces $S_0, \ldots, S_m$. We can decompose the state accordingly:

$$\rho_t = \sum_{i=0}^{m} p_{t,i}\sigma_i,$$

where $\sigma_i$ is a density matrix in subspace $S_i$. Thus the $t$-th state can be fully described by a probability distribution $p_{t,0}, \ldots, p_{t,m}$ that describes how the input register is distributed over the various subspaces. Crucially, only some of the subspaces are "good", meaning that the algorithm will only work if most of the weight is concentrated in the good subspaces at the end of the computation. At the start of the computation, hardly any weight will be in the good subspaces. If we can show that in each query, not too much weight can move from the bad subspaces to the good subspaces, then we again get a lower bound on $T$.

This idea was first introduced by Ambainis in [6] and used there to reprove the "strong direct product theorem" for the OR-function of [20] (we'll explain this in a minute). In this paper we extend it and use it to prove direct product theorems for all *symmetric* functions.

## 1.2 Direct product theorems for symmetric functions

Consider an algorithm that simultaneously needs to compute $k$ independent instances of a function $f$ (denoted $f^{(k)}$). Direct product theorems deal with the optimal tradeoff between the resources and success probability of such algorithms. Suppose we need $t$ "resources" to compute a single instance $f(x)$ with bounded error probability. These resources could for example be time, space, ink, queries, communication, etc. A typical direct product theorem (DPT) has the following form:

> Every algorithm with $T \leq \alpha kt$ resources for computing $f^{(k)}$ has success probability $\sigma \leq 2^{-\Omega(k)}$ (where $\alpha > 0$ is some small constant).

This expresses our intuition that essentially the best way to compute $f^{(k)}$ on $k$ independent instances is to run separate $t$-resource algorithms for each of the instances. Since each of those will have success probability less than 1, we expect that the probability of simultaneously getting all $k$ instances right goes down exponentially with $k$. DPT's can be stated for classical algorithms or quantum algorithms, and $\sigma$ could measure worst-case success probability or average-case success probability under some input distribution. DPT's are generally hard to prove, and Shaltiel [28] even gives general examples where they are just not true (with $\sigma$ average success probability), the above intuition notwithstanding. Klauck, Špalek, and de Wolf [20] recently examined the case where the resource is query complexity and $f = \mathrm{OR}$, and proved an optimal DPT both for classical algorithms and for quantum algorithms (with $\sigma$ worst-case success probability). This strengthened a slightly earlier result of Aaronson [2], who proved that the success probability goes down exponentially with $k$ if the number of queries is bounded by $\alpha\sqrt{kn}$ rather than the $\alpha k\sqrt{n}$ of [20].

Here we generalize their results to the case where $f$ can be any symmetric function, i.e., a function depending only on the Hamming weight $|x|$ of its input. In the case of classical algorithms the situation is quite simple. Every $n$-bit symmetric function $f$ has classical bounded-error query complexity $R_2(f) = \Theta(n)$ and block sensitivity $bs(f) = \Theta(n)$, hence an optimal classical DPT follows immediately from [20, Theorem 3]. Classically, all symmetric functions essentially "cost the same" in terms of query complexity. This is different in the quantum world. For instance, the OR function has bounded-error quantum query complexity $Q_2(\mathrm{OR}) = \Theta(\sqrt{n})$ [17, 11], while Parity needs $n/2$ quantum queries [8, 15]. If $f$ is a $t$-threshold function ($f(x) = 1$ iff $|x| \geq t$, with $t \leq n/2$), then $Q_2(f) = \Theta(\sqrt{tn})$ [8].

Our main result is an essentially optimal quantum DPT for all symmetric functions:

> There is a constant $\alpha > 0$ such that for every symmetric $f$ and every positive integer $k$: Every 2-sided error quantum algorithm with $T \leq \alpha k Q_2(f)$ queries for computing $f^{(k)}$ has success probability $\sigma \leq 2^{-\Omega(k)}$.

Our new direct product theorem generalizes the polynomial-based results of [20] (which strengthened the polynomial-based [2]), but our current proof uses the above-mentioned version of the adversary method.

We have not been able to prove this result using the polynomial method. We can, however, use the polynomial method to prove an incomparable DPT. This result is worse than our main result in applying only to *1-sided error* quantum algorithms[1] for *threshold* functions; but it's better in

[1]The error is 1-sided if 1-bits in the $k$-bit output vector are always correct.

giving a much stronger upper bound on the success probability:

> There is a constant $\alpha > 0$ such that for every $t$-threshold function $f$ and every positive integer $k$: Every 1-sided error quantum algorithm with $T \leq \alpha k Q_2(f)$ queries for computing $f^{(k)}$ has success probability $\sigma \leq 2^{-\Omega(kt)}$.

A similar theorem can be proven for the $k$-fold $t$-search problem, where in each of $k$ inputs of $n$ bits, we want to find at least $t$ ones. The different error bounds $2^{-\Omega(kt)}$ and $2^{-\Omega(k)}$ for 1-sided and 2-sided error algorithms intuitively say that imposing the 1-sided error constraint makes deciding each of the $k$ threshold problems as hard as actually *finding* $t$ ones in each of the $k$ inputs.

### 1.3 Time-Space tradeoffs for evaluating solutions to systems of linear inequalities

As an application we obtain near-optimal time-space tradeoffs for evaluating solutions to systems of linear equalities. Such tradeoffs between the two main computational resources are well known classically for problems like sorting, element distinctness, hashing, etc. In the quantum world, essentially optimal time-space tradeoffs were recently obtained for sorting and for Boolean matrix multiplication [20], but little else is known.

Let $A$ be a fixed $N \times N$ matrix of nonnegative integers. Our inputs are column vectors $x = (x_1, \ldots, x_N)$ and $b = (b_1, \ldots, b_N)$ of nonnegative integers. We are interested in the system

$$Ax \geq b$$

of $N$ linear inequalities, and want to find out which of these inequalities hold (we could also mix $\geq$, $=$, and $\leq$, but omit that for ease of notation).[2] Note that the output is an $N$-bit vector. We want to analyze the tradeoff between the time $T$ and space $S$ needed to solve this problem. Lower bounds on $T$ will be in terms of query complexity. For simplicity we omit polylog factors in the following discussion.

In the classical world, the optimal tradeoff is $TS = N^2$, independent of the values in $b$. This follows from [20, Section 7]. The upper bounds are for deterministic algorithms and the lower bounds are for 2-sided error algorithms. In the quantum world the situation is more complex. Let us put an upper bound $\max\{b_i\} \leq t$. We have two regimes for 2-sided error quantum algorithms:

- Quantum regime. If $S \leq N/t$ then the optimal tradeoff is $T^2 S = tN^3$ (better than classical).
- Classical regime. If $S > N/t$ then the optimal tradeoff is $TS = N^2$ (same as classical).

Our lower bounds hold even for the constrained situation where $b$ is fixed to the all-$t$ vector, $A$ and $x$ are Boolean, and $A$ is sparse in having only $O(N/S)$ non-zero entries in each row.

Since our DPT for 1-sided error algorithms is stronger by an extra factor of $t$ in the exponent, we obtain a stronger lower bound for 1-sided error algorithms:

- If $t \leq S \leq N/t^2$ then the optimal tradeoff for 1-sided error algorithms is $T^2 S \geq t^2 N^3$.
- If $S > N/t^2$ then the optimal tradeoff for 1-sided error algorithms is $TS = N^2$.

We do not know whether the lower bound in the first case is optimal (probably it is not), but note that it is stronger than the optimal bounds that we have for 2-sided error algorithms. This is the first separation of 2-sided and 1-sided error algorithms in the context of quantum time-space tradeoffs.[3]

**Remarks:**

1. Klauck et al. [20] gave direct product theorems not only for quantum query complexity, but also for 2-party quantum communication complexity, and derived some communication-space tradeoffs in analogy to the time-space tradeoffs. This was made possible by a translation of communication protocols to polynomials due to Razborov [26], and the fact that the DPTs of [20] were polynomial-based. Some of the results in this paper can similarly be ported to a communication setting, though only the ones that use the polynomial method.

2. The time-space tradeoffs for 2-sided error algorithms for $Ax \geq b$ similarly hold for a system of $N$ equalities, $Ax = b$. The upper bound clearly carries over, while the lower holds for equalities as well, because our DPT holds even under the promise that the input has weight $t$ or $t-1$. In contrast, the stronger 1-sided error time-space tradeoff does not automatically carry over to systems of equalities, because we do not know how to prove the DPT with bound $2^{-\Omega(kt)}$ under this promise.

## 2. PRELIMINARIES

We assume familiarity with quantum computing [23] and sketch the model of quantum query complexity, referring to [13] for more details, also on the close relation between query complexity and degrees of multivariate polynomials. Suppose we want to compute some function $f$. For input $x \in \{0,1\}^N$, a *query* gives us access to the input bits. It corresponds to the unitary transformation

$$O_x : |i, b, z\rangle \mapsto |i, b \oplus x_i, z\rangle.$$

Here $i \in [N] = \{1, \ldots, N\}$ and $b \in \{0,1\}$; the $z$-part corresponds to the workspace, which is not affected by the query. We assume the input can be accessed only via such queries. A $T$-query quantum algorithm has the form $A = U_T O_x U_{T-1} \cdots O_x U_1 O_x U_0$, where the $U_k$ are fixed unitary transformations, independent of $x$. This $A$ depends on $x$ via the $T$ applications of $O_x$. The algorithm starts in initial $S$-qubit state $|0\rangle$ and its *output* is the result of measuring a dedicated part of the final state $A|0\rangle$. For a Boolean function $f$, the output of $A$ is obtained by observing the leftmost qubit of the final superposition $A|0\rangle$, and its *acceptance probability* on input $x$ is its probability of outputting 1. We mention some well known quantum algorithms that we use as subroutines.

- **Quantum search.** Grover's search algorithm [17, 10] can find an index of a 1-bit in an $n$-bit input in expected number of $O(\sqrt{n/(|x|+1)})$ queries, where $|x|$ is the Hamming weight (number of ones) in the input.

[2] Note that if $A$ and $x$ are Boolean and $b = (t, \ldots, t)$, this gives $N$ overlapping $t$-threshold functions.

[3] Strictly speaking, there's a quadratic gap for OR, but space $\log n$ suffices for the fastest 1-sided and 2-sided error algorithms so there's no real tradeoff in that case.

If $|x|$ is known, the algorithm can be made to find the index in exactly $O(\sqrt{n/(|x|+1)})$ queries, instead of the expected number [11]. By repeated search, we can find $t$ ones in an $n$-bit input with $|x| \geq t$, using $\sum_{i=|x|-t+1}^{|x|} O(\sqrt{n/(i+1)}) = O(\sqrt{tn})$ queries.

- **Quantum counting** [11, Theorem 13]. There is a quantum algorithm that uses $M$ queries to $n$-bit $x$ to compute an estimate $w$ of $|x|$ such that with probability at least $8/\pi^2$

$$|w - |x|| \leq 2\pi \frac{\sqrt{|x|(n-|x|)}}{M} + \pi^2 \frac{n}{M^2}.$$

For investigating time-space tradeoffs we use the circuit model. A circuit accesses its input via an oracle like a query algorithm. Time corresponds to the number of gates in the circuit. We will, however, usually consider the number of queries to the input, which is obviously a lower bound on time. A circuit uses space $S$ if it works with $S$ bits/qubits only. We require that the outputs are made at predefined gates in the circuit, by writing their value to some extra bits/qubits that may not be used later on.

# 3. DIRECT PRODUCT THEOREM FOR SYMMETRIC FUNCTIONS (2-SIDED)

The main result of this paper is the following theorem.

THEOREM 1. *There is a constant $\alpha > 0$ such that for every symmetric $f$ and every positive integer $k$: Every 2-sided error quantum algorithm with $T \leq \alpha k Q_2(f)$ queries for computing $f^{(k)}$ has success probability $\sigma \leq 2^{-\Omega(k)}$.*

Let us first say something about $Q_2(f)$ for a symmetric function $f : \{0,1\}^n \to \{0,1\}$. Let $t$ denote the smallest nonnegative integer such that $f$ is constant on the interval $|x| \in [t, n-t]$. We call this value $t$ the "implicit threshold" of $f$. For instance, functions like OR and AND have $t = 1$, while Parity and Majority have $t = n/2$. If $f$ is the $t$-threshold function, then the implicit threshold is just the threshold. The implicit threshold is related to the parameter $\Gamma(f)$ introduced by Paturi [25] via $t = n/2 - \Gamma(f)/2 \pm 1$. It characterizes the bounded-error quantum query complexity of $f$: $Q_2(f) = \Theta(\sqrt{tn})$ [8]. Hence our resource bound in the above theorem will be $\alpha k\sqrt{tn}$ for some small constant $\alpha > 0$.

We actually prove a stronger statement, applying to any Boolean function $f$ (total or partial) for which $f(x) = 0$ if $|x| = t-1$ and $f(x) = 1$ if $|x| = t$. In this section we give an outline of the proof. Most of the proofs of technical claims are deferred to Appendix A.

Let $\mathcal{A}$ be an algorithm that computes $k$ instances of this weight-$(t-1)$ versus weight-$t$ problem. We recast $\mathcal{A}$ into a different form, using a register that stores the input $x^1, \ldots, x^k$. Let $\mathcal{H}_A$ be the Hilbert space on which $\mathcal{A}$ operates. Let $\mathcal{H}_I$ be an $(\binom{n}{t-1} + \binom{n}{t})^k$-dimensional Hilbert space whose basis states correspond to inputs $(x^1, \ldots, x^k)$ with Hamming weights $|x^1| \in \{t-1, t\}, \ldots, |x^k| \in \{t-1, t\}$. We transform $\mathcal{A}$ into a sequence of transformations on a Hilbert space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_I$. A non-query transformation $U$ on $\mathcal{H}_A$ is replaced with $U \otimes I$ on $\mathcal{H}$. A query is replaced by a transformation $O$ that is equal to $O_{x^1,\ldots,x^k} \otimes I$ on the subspace consisting of states of the form $|s\rangle_A \otimes |x^1 \ldots x^k\rangle_I$.

The starting state of the algorithm on Hilbert space $\mathcal{H}$ is $|\varphi_0\rangle = |\psi_{start}\rangle_A \otimes |\psi_0\rangle_I$ where $|\psi_{start}\rangle$ is the starting state of $\mathcal{A}$ as an algorithm acting on $\mathcal{H}_A$ and $|\psi_0\rangle = |\psi_{one}\rangle^{\otimes k}$ is a tensor product of $k$ copies of the state $|\psi_{one}\rangle$ in which half of the weight is on $|x\rangle$ with $|x| = t$, the other half is on $|x\rangle$ with $|x| = t-1$, and any two states $|x\rangle$ with the same $|x|$ have equal amplitudes:

$$|\psi_{one}\rangle = \frac{1}{\sqrt{2\binom{n}{t}}} \sum_{x:|x|=t} |x\rangle + \frac{1}{\sqrt{2\binom{n}{t-1}}} \sum_{x:|x|=t-1} |x\rangle.$$

Let $|\varphi_d\rangle$ be the state of the algorithm $\mathcal{A}$, as a sequence of transformations on $\mathcal{H}$, after the $d$-th query. Let $\rho_d$ be the mixed state in $\mathcal{H}_I$ obtained from $|\varphi_d\rangle$ by tracing out the $\mathcal{H}_A$ register.

We define two decompositions of $\mathcal{H}_I$ into a direct sum of subspaces. We have $\mathcal{H}_I = (\mathcal{H}_{one})^{\otimes k}$ where $\mathcal{H}_{one}$ is the input Hilbert space for one instance, with basis states $|x\rangle$, $x \in \{0,1\}^n, |x| \in \{t-1, t\}$. Let

$$|\psi^0_{i_1,\ldots,i_j}\rangle = \frac{1}{\sqrt{\binom{n-j}{t-1-j}}} \sum_{\substack{x_1,\ldots,x_n:\\ x_1+\cdots+x_n=t-1,\\ x_{i_1}=\cdots=x_{i_j}=1}} |x_1 \ldots x_n\rangle$$

and let $|\psi^1_{i_1,\ldots,i_j}\rangle$ be a similar state with $x_1 + \cdots + x_n = t$ instead of $x_1 + \cdots + x_n = t-1$. Let $T_{j,0}$ (resp. $T_{j,1}$) be the space spanned by all states $|\psi^0_{i_1,\ldots,i_j}\rangle$ (resp. $|\psi^1_{i_1,\ldots,i_j}\rangle$) and let $S_{j,a} = T_{j,a} \cap T^{\perp}_{j-1,a}$. For a subspace $S$, we use $\Pi_S$ to denote the projector onto $S$. Let $|\tilde{\psi}^a_{i_1,\ldots,i_j}\rangle = \Pi_{T^{\perp}_{j-1,a}} |\psi^a_{i_1,\ldots,i_j}\rangle$. For $j < t$, let $S_{j,+}$ be the subspace spanned by the states

$$\frac{|\tilde{\psi}^0_{i_1,\ldots,i_j}\rangle}{\|\tilde{\psi}^0_{i_1,\ldots,i_j}\|} + \frac{|\tilde{\psi}^1_{i_1,\ldots,i_j}\rangle}{\|\tilde{\psi}^1_{i_1,\ldots,i_j}\|}$$

and $S_{j,-}$ be the subspace spanned by

$$\frac{|\tilde{\psi}^0_{i_1,\ldots,i_j}\rangle}{\|\tilde{\psi}^0_{i_1,\ldots,i_j}\|} - \frac{|\tilde{\psi}^1_{i_1,\ldots,i_j}\rangle}{\|\tilde{\psi}^1_{i_1,\ldots,i_j}\|}$$

For $j = t$, we define $S_{t,-} = S_{t,1}$ and there is no subspace $S_{t,+}$. Thus $\mathcal{H}_{one} = \bigoplus_{j=0}^{t-1}(S_{j,+} \oplus S_{j,-}) \oplus S_{t,-}$. Let us try to give some intuition. In the spaces $S_{j,+}$ and $S_{j,-}$, we may be said to "know" the positions of $j$ of the ones. In the $S_{j,-}$ subspaces we have distinguished the 0-inputs from 1-inputs by the relative phase, while in the $S_{j,+}$ subspace we have not distinguished them. Accordingly, the algorithm is doing well on this one instance if most of the state sits in the "good" subspaces $S_{j,-}$.

For the space $\mathcal{H}_I$ (representing $k$ independent inputs for our function) and $r_1, \ldots, r_k \in \{+,-\}$, we define

$$S_{j_1,\ldots,j_k,r_1,\ldots,r_k} = S_{j_1,r_1} \otimes S_{j_2,r_2} \otimes \cdots \otimes S_{j_k,r_k}.$$

Let $\mathcal{S}_{m-}$ be the direct sum of all $S_{j_1,\ldots,j_k,r_1,\ldots,r_k}$ such that exactly $m$ of the signs $r_1, \ldots, r_k$ are equal to $-$. Then $\mathcal{H}_I = \bigoplus_m \mathcal{S}_{m-}$. This is the first decomposition.

The above intuition for one instance carries over to $k$ instances: the more minuses the better for the algorithm. Conversely, if most of the input register sits in $S_{m-}$ for low $m$, then its success probability will be small. More precisely, in Appendix A.1 we prove:

LEMMA 2. *Let $\rho$ be the reduced density matrix of $\mathcal{H}_I$. If the support of $\rho$ is contained in $\mathcal{S}_{0-} \oplus \mathcal{S}_{1-} \oplus \cdots \oplus \mathcal{S}_{m-}$, then*

*the probability that measuring $\mathcal{H}_A$ gives the correct answer is at most* $\frac{\sum_{m'=0}^{m}\binom{k}{m'}}{2^k}$.

Note that this probability is exponentially small in $k$ for, say, $m = k/3$. The following consequence of this lemma is proven in Appendix A.2:

COROLLARY 3. *Let $\rho$ be the reduced density matrix of $\mathcal{H}_I$. The probability that measuring $\mathcal{H}_A$ gives the correct answer is at most*

$$\frac{\sum_{m'=0}^{m}\binom{k}{m'}}{2^k} + 4\sqrt{\operatorname{Tr}\Pi_{(\mathcal{S}_{0-}\oplus\mathcal{S}_{1-}\oplus\cdots\oplus\mathcal{S}_{m-})^\perp}\rho}.$$

To define the second decomposition, we express $\mathcal{H}_{one} = \bigoplus_{j=0}^{t/2} R_j$ with $R_j = S_{j,+}$ for $j < t/2$ and

$$R_{t/2} = \bigoplus_{j\geq t/2} S_{j,+} \oplus \bigoplus_{j\geq 0} S_{j,-}.$$

Intuitively, all subspaces except for $R_{t/2}$ are "bad" for the algorithm, since they equal the "bad" $S_{j,+}$ subspaces. Let $\mathcal{R}_\ell$ be the direct sum of all $R_{j_1}\otimes\cdots\otimes R_{j_k}$ satisfying $j_1 + \cdots + j_k = \ell$. Then $\mathcal{H}_I = \bigoplus_{\ell=0}^{tk/2}\mathcal{R}_\ell$. This is the second decomposition.

Intuitively, the algorithm can only have good success probability if for most of the $k$ instances, most of the input register sits in $R_{t/2}$. Aggregated over all $k$ instances, this means that the algorithm will only work well if most of the $k$-input register sits in $\mathcal{R}_\ell$ for $\ell$ large, meaning fairly close to $kt/2$. Our goal below is to show that this cannot happen if the number of queries is small.

Let $\mathcal{R}'_j = \bigoplus_{\ell=j}^{tk/2}\mathcal{R}_\ell$. Note that $\mathcal{S}_{m-} \subseteq \mathcal{R}'_{tm/2}$ for every $m$: $\mathcal{S}_{m-}$ is the direct sum of subspaces $S = S_{j_1,r_1}\otimes\cdots\otimes S_{j_k,r_k}$ having $m$ minuses among $r_1,\ldots,r_k$; each such minus-subspace sits in the corresponding $R_{t/2}$ and hence $S \subseteq \mathcal{R}'_{tm/2}$. This implies

$$(\mathcal{S}_{0-}\oplus\mathcal{S}_{1-}\oplus\cdots\oplus\mathcal{S}_{(m-1)-})^\perp \subseteq \mathcal{R}'_{tm/2}.$$

Accordingly, if we prove an upper bound on $\operatorname{Tr}\Pi_{\mathcal{R}'_{tm/2}}\rho_T$, where $T$ is the total number of queries, this bound together with Corollary 3 implies an upper bound on the success probability of $\mathcal{A}$. To bound $\operatorname{Tr}\Pi_{\mathcal{R}'_{tm/2}}\rho_T$, we consider the following potential function

$$P(\rho) = \sum_{m=0}^{tk/2} q^m \operatorname{Tr}\Pi_{\mathcal{R}_m}\rho,$$

where $q = 1 + \frac{1}{t}$. Then for every $d$

$$\operatorname{Tr}\Pi_{\mathcal{R}'_{tm/2}}\rho_d \leq P(\rho_d)q^{-tm/2} = P(\rho_d)e^{-(1+o(1))m/2}. \quad (1)$$

$P(\rho_0) = 1$, because the initial state $|\psi_0\rangle$ is a tensor product of the states $|\psi_{one}\rangle$ on each copy of $\mathcal{H}_{one}$ and $|\psi_{one}\rangle$ belongs to $S_{0,+}$, hence $|\psi_0\rangle$ belongs to $\mathcal{R}_0$. In Appendix A.4 we prove

LEMMA 4. *There is a constant $C$ such that*

$$P(\rho_{j+1}) \leq \left(1 + \frac{C}{\sqrt{tn}}(q^{t/2}-1) + \frac{C\sqrt{t}}{\sqrt{n}}(q-1)\right)P(\rho_j).$$

Since $q = 1 + \frac{1}{t}$, Lemma 4 means that $P(\rho_{j+1}) \leq (1 + \frac{C\sqrt{e}}{\sqrt{tn}})P(\rho_j)$ and $P(\rho_j) \leq (1 + \frac{C\sqrt{e}}{\sqrt{tn}})^j \leq e^{2Cj/\sqrt{tn}}$. By equation (1), for the final state after $T$ queries we have

$$\operatorname{Tr}\Pi_{\mathcal{R}'_{tm/2}}\rho_T \leq e^{2CT/\sqrt{tn}-(1+o(1))m/2}.$$

We take $m = k/3$. Then if $T \leq m\sqrt{tn}/8C$, this expression is exponentially small in $k$. Together with Corollary 3, this implies the theorem.

# 4. DIRECT PRODUCT THEOREM FOR THRESHOLD FUNCTIONS (1-SIDED)

The previous section used the adversary method to prove a direct product theorem for 2-sided error algorithms computing $k$ instances of some symmetric function. In this section we use the polynomial method to obtain stronger direct product theorems for 1-sided error algorithms for threshold functions. An algorithm for $f^{(k)}$ has 1-sided error if the 1's in its $k$-bit output vector are always correct.

Our use of polynomials is a relatively small extension of the argument in [20]. We use three results about polynomials, also used in [12, 20]. The first is by Coppersmith and Rivlin [14, p. 980] and gives a general bound for polynomials bounded by 1 at integer points:

THEOREM 5 (COPPERSMITH & RIVLIN [14]). *Every polynomial $p$ of degree $d \leq n$ that has absolute value*

$$|p(i)| \leq 1 \text{ for all integers } i \in [0,n],$$

*satisfies*

$$|p(x)| < ae^{bd^2/n} \text{ for all real } x \in [0,n],$$

*where $a, b > 0$ are universal constants (no explicit values for $a$ and $b$ are given in [14]).*

The other two results concern the Chebyshev polynomials $T_d$, defined as in [27]:

$$T_d(x) = \frac{1}{2}\left(\left(x+\sqrt{x^2-1}\right)^d + \left(x-\sqrt{x^2-1}\right)^d\right).$$

$T_d$ has degree $d$ and its absolute value $|T_d(x)|$ is bounded by 1 if $x \in [-1,1]$. On the interval $[1,\infty)$, $T_d$ exceeds all others polynomials with those two properties ([27, p.108] and [25, Fact 2]):

THEOREM 6. *If $q$ is a polynomial of degree $d$ such that $|q(x)| \leq 1$ for all $x \in [-1,1]$ then $|q(x)| \leq |T_d(x)|$ for all $x \geq 1$.*

LEMMA 7 (PATURI [25]). *$T_d(1+\mu) \leq e^{2d\sqrt{2\mu+\mu^2}}$ for all $\mu \geq 0$.*

PROOF. For $x = 1+\mu$: $T_d(x) \leq (x+\sqrt{x^2-1})^d = (1+\mu+\sqrt{2\mu+\mu^2})^d \leq (1+2\sqrt{2\mu+\mu^2})^d \leq e^{2d\sqrt{2\mu+\mu^2}}$. □

The following lemma is key. It analyzes polynomials that are 0 on the first $m$ integer points, and that significantly "jump" a bit later.

LEMMA 8. *Suppose $E, N, m$ are integers satisfying $10 \leq E \leq \frac{N}{2m}$, and let $p$ be a degree-$D$ polynomial such that*

$p(i) = 0$ *for all* $i \in \{0, \ldots, m-1\}$,
$p(8m) = \sigma$,
$p(i) \in [0,1]$ *for all* $i \in \{0, \ldots, N\}$.

*Then* $\sigma \leq 2^{O(D^2/N + D\sqrt{Em/N} - m \log E)}$.

PROOF. Divide $p$ by $\prod_{j=0}^{m-1}(x-j)$ to obtain

$$p(x) = q(x) \prod_{j=0}^{m-1} (x-j),$$

where $d = \deg(q) = D - m$. This implies the following about the values of the polynomial $q$:

$$\begin{aligned} |q(8m)| &\geq \sigma/(8m)^m, \\ |q(i)| &\leq 1/((E-1)m)^m \quad \text{for } i \in \{Em, \ldots, N\}. \end{aligned}$$

Theorem 5 implies that there are constants $a, b > 0$ such that

$$\begin{aligned} |q(x)| &\leq \frac{a}{((E-1)m)^m} e^{bd^2/(N-Em)} = B \\ &\quad \text{for all real } x \in [Em, N]. \end{aligned}$$

We now divide $q$ by $B$ to normalize it, and rescale the interval $[Em, N]$ to $[1, -1]$ to get a degree-$d$ polynomial $t$ satisfying

$$\begin{aligned} |t(x)| &\leq 1 \quad \text{for all } x \in [-1, 1], \\ t(1+\mu) &= q(8m)/B \quad \text{for } \mu = 2(E-8)m/(N-Em). \end{aligned}$$

Since $t$ cannot grow faster than the degree-$d$ Chebyshev polynomial, Theorem 6 and Lemma 7 imply

$$t(1+\mu) \leq e^{2d\sqrt{2\mu+\mu^2}}.$$

Combining our upper and lower bounds on $t(1+\mu)$ gives

$$\frac{\sigma}{(8m)^m} \cdot \frac{((E-1)m)^m}{ae^{O(d^2/N)}} \leq e^{O(d\sqrt{Em/N})},$$

which implies the lemma. □

THEOREM 9. *There exists* $\alpha > 0$ *such that for every threshold function* $T_t$ *and positive integer* $k$*: Every 1-sided error quantum algorithm with* $T \leq \alpha k Q_2(T_t)$ *queries for computing* $T_t^{(k)}$ *has success probability* $\sigma \leq 2^{-\Omega(kt)}$.

PROOF. We assume without loss of generality that $t \leq n/20$, the other cases can easily be reduced to this. We know that $Q_2(T_t) = \Theta(\sqrt{tn})$ [8]. Consider a quantum algorithm $A$ with $T \leq \alpha k\sqrt{tn}$ queries that computes $f^{(k)}$ with success probability $\sigma$. Roughly speaking, we use $A$ to solve one big threshold problem on the total input, and then invoke the polynomial lemma to upper bound the success probability.

Define a new quantum algorithm $B$ on an input $x$ of $N = kn$ bits, as follows: $B$ runs $A$ on a random permutation $\pi(x)$, and then outputs 1 iff the $k$-bit output vector has at least $k/2$ ones.

Let $m = kt/2$. Note that if $|x| < m$, then $B$ always outputs 0 because the 1-sided error output vector must have fewer than $k/2$ ones. Now suppose $|x| = 8m = 4kt$. Call an $n$-bit input block "full" if $\pi(x)$ contains at least $t$ ones in that block. Let $F$ be the random variable counting how many of the $k$ blocks are full. We claim that $\Pr[F \geq k/2] \geq 1/9$. To prove this, observe that the number $B$ of ones in one fixed block is a random variable distributed according to a hypergeometric distribution ($4kt$ balls into $N$ boxes, $n$ of which count as success) with expectation $\mu = 4t$ and variance $V \leq 4t$. Using Chebyshev's inequality we bound the probability that this block is not full:

$$\begin{aligned} \Pr[B < t] &\leq \Pr[|B - \mu| > 3t] \leq \Pr[|B - \mu| > (3\sqrt{t}/2)\sqrt{V}] \\ &< \frac{1}{(3\sqrt{t}/2)^2} \leq \frac{4}{9}. \end{aligned}$$

Hence the probability that the block is full ($B \geq t$) is at least 5/9. This is true for each of the $k$ blocks, so using linearity of expectation we have

$$\frac{5k}{9} \leq \mathrm{Exp}[F] \leq \Pr[F \geq k/2] \cdot k + (1 - \Pr[F \geq k/2]) \cdot \frac{k}{2}.$$

This implies $\Pr[F \geq k/2] \geq 1/9$, as claimed. But then on all inputs with $|x| = 8m$, $B$ outputs 1 with probability at least $\sigma/9$.

Algorithm $B$ uses $\alpha k\sqrt{tn}$ queries. By [8] and symmetrization, $B$'s acceptance probability is a single-variate polynomial $p$ of degree $D \leq 2\alpha k\sqrt{tn}$ such that

$p(i) = 0$ for all $i \in \{0, \ldots, m-1\}$,
$p(8m) \geq \sigma/9$,
$p(i) \in [0,1]$ for all $i \in \{0, \ldots, N\}$.

The result now follows by applying Lemma 8 with $N = kn$, $m = kt/2$, $E = 10$, and $\alpha$ a sufficiently small positive constant. □

# 5. TIME-SPACE TRADEOFF FOR SYSTEMS OF LINEAR INEQUALITIES

Let $A$ be a fixed $N \times N$ matrix of nonnegative integers and let $x, b$ be two input vectors of $N$ nonnegative integers smaller or equal to $t$. A *matrix-vector product with upper bound*, denoted by $y = (Ax)_{\leq b}$, is a vector $y$ such that $y_i = \min((Ax)[i], b_i)$. An *evaluation of a system of linear inequalities* $Ax \geq b$ is the $N$-bit vector of the truth values of the individual inequalities. Here we present a quantum algorithm for matrix-vector product with upper bound that satisfies time-space tradeoff $T^2 S = O(tN^3 (\log N)^5)$. We then use our direct product theorems to show this is close to optimal.

## 5.1 Upper bound

It is easy to prove that matrix-vector products with upper bound $t$ can be computed by a classical algorithm with $TS = O(N^2 \log t)$, as follows. Let $S' = S/\log t$ and divide the matrix $A$ into $(N/S')^2$ blocks of size $S' \times S'$ each. The output vector is evaluated row-wise as follows: (1) Clear $S'$ counters, one for each row, and read $b_i$. (2) For each block, read $S'$ input variables, multiply them by the corresponding submatrix of $A$, and update the counters, but do not let them grow larger than $b_i$. (3) Output the counters. The space used is $O(S' \log t) = O(S)$ and the total query complexity is $T = O(\frac{N}{S'} \cdot \frac{N}{S'} \cdot S') = O(N^2 \log t / S)$.

The quantum algorithm BOUNDED MATRIX PRODUCT works in a similar way and it is outlined in Table 1. We compute the matrix product in groups of $S' = S/\log N$ rows, read input variables, and update the counters accordingly. The advantage over the classical algorithm is that we use the faster quantum search and quantum counting for finding non-zero entries.

BOUNDED MATRIX PRODUCT (fixed matrix $A_{N\times N}$, threshold $t$, input vectors $x$ and $b$ of length $N$)
returns output vector $y = (Ax)_{\le b}$:

- For $i = 1, 2, \ldots, \frac{N}{S'}$, where $S' = S/\log N$:
  1. Run SMALL MATRIX PRODUCT on the $i$-th block of $S'$ rows of $A$.
  2. Output the $S'$ obtained results for those rows.

SMALL MATRIX PRODUCT (fixed $A_{S'\times N}$, input $x_{N\times 1}$ and $b_{S'\times 1}$) returns $y_{S'\times 1} = (Ax)_{\le b}$:

1. Initialize $y := (0, 0, \ldots, 0)$, $p := 1$, $U := \{1, \ldots, S'\}$, and read $b$. Let $a_{1\times N}$ denote an on-line computed row-vector with $a_j = 1$ if $A[u, j] = 1$ for some $u \in U$, and $a_j = 0$ otherwise.
2. While $p \le N$ and $U \ne \emptyset$, do the following:
   (a) Let $\tilde{c}_{p,k}$ denote an estimate of the scalar product
   $$c_{p,k} = \sum_{j=p}^{p+k-1} a_j x_j.$$
   Initialize $k = S'$. First, while $p + k - 1 < N$ and $\tilde{c}_{p,k} < S'$, double $k$. Second, find by binary search the maximal $\ell \in [\frac{k}{2}, k]$ such that $p + \ell - 1 \le N$ and $\tilde{c}_{p,\ell} \le 2S'$.
   (b) Use quantum search to find the set $J$ of all positions $j \in [p, p + \ell - 1]$ such that $a_j x_j > 0$.
   (c) For all $j \in J$, read $x_j$, and then do the following for all $u \in U$:
   - Increase $y_u$ by $A[u, j]x_j$.
   - If $y_u \ge b_u$, set $y_u := b_u$ and remove $u$ from $U$.

   (d) Increase $p$ by $\ell$.
3. Return $y$.

**Table 1: Algorithm Bounded Matrix Product**

The $u$-th row is called *open* if its counter hasn't yet reached $b_u$. The subroutine SMALL MATRIX PRODUCT maintains a set of open rows $U \subseteq \{1, \ldots, S'\}$ and counters $0 \le y_u \le b_u$ for all $u \in U$. We process the input $x$ in blocks, each containing between $S' - O(\sqrt{S'})$ and $2S' + O(\sqrt{S'})$ non-zero numbers at the positions $j$ where $A[u, j] \ne 0$ for some $u \in U$. The length $\ell$ of such a block is first found by iterated quantum counting (with number of queries specified in the proof below) and the non-zero input numbers are then found by a Grover search. For each such number, we update all counters $y_u$ and close all rows that exceeded their threshold $b_u$.

THEOREM 10. BOUNDED MATRIX PRODUCT *has bounded error probability, its space complexity is $O(S)$, and its query complexity is $T = O(N^{3/2}\sqrt{t} \cdot (\log N)^{5/2}/\sqrt{S})$.*

PROOF. The space complexity of SMALL MATRIX PRODUCT is $O(S' \log N) = O(S)$, because it stores a subset $U \subseteq \{1, \ldots, S'\}$, integer vectors $y, b$ of length $S'$ with numbers at most $t \le N$, the set $J$ of size $O(S')$ with numbers at most $N$, and a few counters. Let us compute its query complexity.

Consider the $i$-th block found by SMALL MATRIX PRODUCT; let $p_i$ be its left column, let $\ell_i$ be its length, and let $U_i$ be the set of open rows at the beginning of processing of this block. The scalar product $c_{p_i,\ell_i}$ is estimated by quantum counting with $M = \sqrt{\ell_i}$ queries. Finding a proper $\ell_i$ requires $O(\log \ell_i)$ iterations. Let $r_i$ be the number of rows closed during processing of this block and let $s_i$ be the total number added to the counters for other (still open) rows in this block. The numbers $\ell_i, r_i, s_i$ are random variables. If we instantiate them at the end of the quantum subroutine, the following inequalities hold:

$$\sum_i \ell_i \le N, \quad \sum_i r_i \le S', \text{ and } \quad \sum_i s_i \le tS'.$$

The iterated Grover search finds ones for two purposes: closing rows and increasing counters. Since each $b_i \le t$, the total cost in the $i$-th block is at most $\sum_{j=1}^{r_i t} O(\sqrt{\ell_i/j}) + \sum_{j=1}^{s_i} O(\sqrt{\ell_i/j}) = O(\sqrt{\ell_i r_i t} + \sqrt{\ell_i s_i})$. By a Cauchy-Schwarz inequality, the total number of queries that SMALL MATRIX PRODUCT spends in the Grover searches is at most

$$\sum_{i=1}^{\#\text{blocks}} (\sqrt{\ell_i r_i t} + \sqrt{\ell_i s_i}) \le \sqrt{\sum_i \ell_i}\sqrt{t\sum_i r_i} + \sqrt{\sum_i \ell_i}\sqrt{\sum_i s_i}$$
$$\le \sqrt{N}\sqrt{tS'} + \sqrt{N}\sqrt{tS'} = O(\sqrt{NS't}).$$

The error probability of the Grover searches can be made polynomially small in a logarithmic overhead. It remains to analyze the outcome and error probability of quantum counting. Let $c_i = c_{p_i,\ell_i} \in [S', 2S']$. One quantum counting call with $M = \sqrt{\ell_i}$ queries gives an estimate $w$ such that

$$|w - c_i| = O\left(\sqrt{\frac{c_i(\ell_i - c_i)}{\ell_i}} + \frac{\ell_i}{\ell_i}\right) = O(\sqrt{c_i}) = O(\sqrt{S'})$$

with probability at least $8/\pi^2 \approx 0.8$. We do it $O(\log N)$ times and take the median, hence we obtain an estimate $\tilde{c}$ of $c_i$ with accuracy $O(\sqrt{S'})$ with polynomially small error probability. The result of quantum counting is compared with the given threshold, that is with $S'$ or $2S'$. Binary search for $\ell \in [\frac{k}{2}, k]$ costs another factor of $\log k \le \log N$. By a Cauchy-Schwarz inequality, the total number of queries spent in the quantum counting is at most $(\log N)^2$ times

$$\sum_i \sqrt{\ell_i} \le \sqrt{\sum_i \ell_i}\sqrt{\sum_i 1} \le \sqrt{N}\sqrt{\#\text{blocks}}$$
$$\le \sqrt{N}\sqrt{S' + t} \le \sqrt{NS't},$$

because in every block the algorithm closes a row or adds $\Theta(S')$ in total to the counters. The number of closed rows is at most $S'$ and the number $S'$ can be added at most $t$ times.

The total query complexity of SMALL MATRIX PRODUCT is thus $O(\sqrt{NS't} \cdot (\log N)^2)$ and the query complexity of BOUNDED MATRIX PRODUCT is $N/S'$-times bigger. The overall error probability is at most the sum of the polynomially small error probabilities of the different subroutines, hence it can be kept below 1/3. □

## 5.2 Lower bound

Here we use our direct product theorems to lower-bound the quantity $T^2S$ for $T$-query, $S$-space quantum algorithms for systems of linear inequalities. The lower bound even holds if we fix $b$ to the all-$t$ vector $\vec{t}$ and let $A$ and $x$ be Boolean.

THEOREM 11. *Let $S = \min(O(N/t), o(N/\log N))$. There exists an $N \times N$ Boolean matrix $A$ such that every 2-sided error quantum algorithm that uses $T$ queries and $S$ qubits of space to decide a system $Ax \geq \vec{t}$ of $N$ inequalities, satisfies $T^2S = \Omega(tN^3)$.*

PROOF. The proof is a modification of Theorem 22 of [20] (quant-ph version). They use the probabilistic method to establish the following

**Fact:** For every $k = o(N/\log N)$, there exists an $N \times N$ Boolean matrix $A$, such that all rows of $A$ have weight $N/2k$, and every set of $k$ rows of $A$ contains a set $R$ of $k/2$ rows with the following property: each row in $R$ contains at least $n = N/6k$ ones that occur in no other row of $R$.

Fix a matrix $A$ for $k = cS$, for some constant $c$ to be chosen later. Consider a quantum circuit with $T$ queries and space $S$ that solves the problem with success probability at least 2/3. We "slice" the quantum circuit into disjoint consecutive slices, each containing $Q = \alpha\sqrt{tNS}$ queries, where $\alpha$ is the constant from our direct product theorem (Theorem 1). The total number of slices is $L = T/Q$. Together, these disjoint slices contain all $N$ output gates. Our aim below is to show that with sufficiently small constant $\alpha$ and sufficiently large constant $c$, no slice can produce more than $k$ outputs. This will imply that the number of slices is $L \geq N/k$, hence

$$T = LQ \geq \frac{\alpha N^{3/2}\sqrt{t}}{c\sqrt{S}}.$$

Now consider any slice. It starts with an $S$-qubit state that is delivered by the previous slice and depends on the input, then it makes $Q$ queries and outputs some $\ell$ results that are jointly correct with probability at least 2/3. Suppose, by way of contradiction, that $\ell \geq k$. Then there exists a set of $k$ rows of $A$ such that our slice produces the $k$ corresponding results ($t$-threshold functions) with probability at least 2/3. By the above Fact, some set $R$ of $k/2$ of those rows has the property that each row in $R$ contains a set of $n = N/6k = \Theta(N/S)$ ones that do not occur in any of the $k/2 - 1$ other rows of $R$. By setting all other $N - kn/2$ bits of $x$ to 0, we naturally get that our slice, with the appropriate $S$-qubit starting state, solves $k/2$ independent $t$-threshold functions $T_t$ on $n$ bits each. (Note that we need $t \leq n/2 = O(N/S)$; this follows from our assumption $S = O(N/t)$ with appropriately small constant in the $O(\cdot)$.) Now we replace the initial $S$-qubit state by the completely mixed state, which has "overlap" $2^{-S}$ with every $S$-qubit state. This turns the slice into a stand-alone algorithm solving $T_t^{(k/2)}$ with success probability

$$\sigma \geq \frac{2}{3}2^{-S}.$$

But this algorithm uses only $Q = \alpha\sqrt{tNS} = O(\alpha k\sqrt{tn})$ queries, so our direct product theorem (Theorem 1) with sufficiently small constant $\alpha$ implies

$$\sigma \leq 2^{-\Omega(k/2)} = 2^{-\Omega(cS/2)}.$$

Choosing $c$ a sufficiently large constant (independent of this specific slice), our upper and lower bounds on $\sigma$ contradict. Hence the slice must produce fewer than $k$ outputs. □

It is easy to see that the case $S \geq N/t$ (equivalently, $t \geq N/S$) is at least as hard as the $S = N/t$ case, for which we have the lower bound $T^2S = \Omega(tN^3) = \Omega(N^4/S)$, hence $TS = \Omega(N^2)$. But that lower bound matches the *classical* deterministic upper bound up to a logarithmic factor and hence is essentially tight also for quantum. We thus have two different regimes for space: for small space, a quantum computer is faster than a classical one in solving systems of linear inequalities, while for large space it is not.

A similar slicing proof using Theorem 9 (with each slice of $Q = \alpha\sqrt{NS}$ queries producing at most $S/t$ outputs) gives the following lower bound on time-space tradeoffs for 1-sided error algorithms.

THEOREM 12. *Let $t \leq S \leq \min(O(N/t^2), o(N/\log N))$. There exists an $N \times N$ Boolean matrix $A$ such that every 1-sided error quantum algorithm that uses $T$ queries and $S$ qubits of space to decide a system $Ax \geq \vec{t}$ of $N$ inequalities, satisfies $T^2S = \Omega(t^2N^3)$.*

Note that our lower bound $\Omega(t^2N^3)$ for 1-sided error algorithms is higher by a factor of $t$ than the best upper bounds for 2-sided error algorithms. This lower bound is probably not optimal. If $S > N/t^2$ then the essentially optimal classical tradeoff $TS = \Omega(N^2)$ takes over.

# 6. SUMMARY

In this paper we described a new version of the adversary method for quantum query lower bounds, based on analyzing the eigenspace structure of the problem we want to lower bound. We proved two new quantum direct product theorems, the first using the new adversary method, the second using the polynomial method:

- For every symmetric function $f$, every 2-sided error quantum algorithm for $f^{(k)}$ using fewer than $\alpha kQ_2(f)$ queries has success probability at most $2^{-\Omega(k)}$.
- For every $t$-threshold function $f$, every 1-sided error quantum algorithm for $f^{(k)}$ using fewer than $\alpha kQ_2(f)$ queries has success probability at most $2^{-\Omega(kt)}$.

Both results are tight up to constant factors. From these results we derived the following time-space tradeoffs for quantum algorithms that decide a system $Ax \geq b$ of $N$ linear inequalities (where $A$ is a fixed $N \times N$ matrix of nonnegative integers, $x, b$ are variable, and $b_i \leq t$ for all $i$):

- Every $T$-query, $S$-space 2-sided error quantum algorithm for evaluating $Ax \geq b$ satisfies $T^2S = \Omega(tN^3)$ if $S \leq N/t$, and satisfies $TS = \Omega(N^2)$ if $S > N/t$. We gave an algorithm matching these bounds up to polylog factors.
- Every $T$-query, $S$-space 1-sided error quantum algorithm for evaluating $Ax \geq b$ satisfies $T^2S = \Omega(t^2N^3)$ if $t \leq S \leq N/t^2$, and satisfies $TS = \Omega(N^2)$ if $S > N/t^2$. We do not have a matching algorithm in the first case and conjecture that this bound is not tight.

# 7. REFERENCES


[1] S. Aaronson and Y. Shi. Quantum lower bounds for the collision and the element distinctness problems *Journal of the ACM*, 51(4):595–605, 2004

[2] S. Aaronson. Limitations of quantum advice and one-way communication. In *Proc. of 19th Conference on Computational Complexity*, p. 320–332, 2004.

[3] A. Ambainis. Quantum lower bounds by quantum arguments. In *Proc. of 32nd STOC*, p. 636–643, 2000.

[4] A. Ambainis. Polynomial degree vs quantum query complexity. In *Proc. of 44th FOCS*, p. 30–239, 2003.

[5] A. Ambainis. Quantum walk algorithm for element distinctness. In *Proc. of 45st FOCS*, p. 22–31, 2004.

[6] A. Ambainis. A new quantum lower bound method, with an application to strong direct product theorem for quantum search. quant-ph/0508200, 26 Aug 2005.

[7] H. Barnum, M. Saks, and M. Szegedy. Quantum query complexity and semi-definite programming. In *Proc. of 18th Conference on Computational Complexity*, p. 179–193, 2003.

[8] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. de Wolf. Quantum lower bounds by polynomials. *Journal of the ACM*, 48(4):778–797, 2001.

[9] E. Bernstein and U. Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

[10] M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4–5):493–505, 1998.

[11] G. Brassard, P. Høyer, M. Mosca, and A. Tapp. Quantum amplitude amplification and estimation. In *Quantum Computation and Quantum Information: A Millennium Volume*, volume 305 of *AMS Contemporary Mathematics Series*, p. 53–74. 2002.

[12] H. Buhrman, R. Cleve, R. de Wolf, and C. Zalka. Bounds for small-error and zero-error quantum algorithms. In *Proc. of 40th FOCS*, p. 358–368, 1999.

[13] H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: A survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

[14] D. Coppersmith and T. J. Rivlin. The growth of polynomials bounded at equally spaced points. *SIAM Journal on Mathematical Analysis*, 23(4):970–983, 1992.

[15] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. A limit on the speed of quantum computation in determining parity. *Physical Review Letters*, 81:5442–5444, 1998.

[16] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley, second edition, 1994.

[17] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proc. of 28th STOC*, p. 212–219, 1996.

[18] P. Høyer, M. Mosca, and R. de Wolf. Quantum search on bounded-error inputs. In *Proc. of 30th ICALP'03*, volume 2719 of *LNCS*, p. 291–299. Springer, 2003.

[19] P. Høyer, J. Neerbek, and Y. Shi. Quantum complexities of ordered searching, sorting, and element distinctness. *Algorithmica*, 34(4):429–448, 2002.

[20] H. Klauck, R. Špalek, and R. de Wolf. Quantum and classical strong direct product theorems and optimal time-space tradeoffs. In *Proc. of 45th FOCS*, p. 12–21, 2004.

[21] S. Laplante and F. Magniez. Lower bounds for randomized and quantum query complexity using Kolmogorov arguments. In *Proc. of 19th Conference on Computational Complexity*, p. 294–304, 2004.

[22] F. Magniez, M. Santha, and M. Szegedy. Quantum algorithms for the triangle problem. In *Proc. of 16th SODA*, p. 1109–1117, 2005.

[23] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[24] N. Nisan and M. Szegedy. On the degree of Boolean functions as real polynomials. *Computational Complexity*, 4(4):301–313, 1994.

[25] R. Paturi. On the degree of polynomials that approximate symmetric Boolean functions. In *Proc. of 24th STOC*, p. 468–474, 1992.

[26] A. Razborov. Quantum communication complexity of symmetric predicates. *Izvestiya of the Russian Academy of Science, mathematics*, 67(1):159–176, 2003.

[27] T. J. Rivlin. *Chebyshev Polynomials: From Approximation Theory to Algebra and Number Theory*. Wiley-Interscience, second edition, 1990.

[28] R. Shaltiel. Towards proving strong direct product theorems. In *Proc. of 16th Conference on Computational Complexity*, p. 107–119, 2001.

[29] R. Špalek and M. Szegedy. All quantum adversary methods are equivalent. *Theory of Computing*, 2(1):1–18, 2006.

[30] S. Zhang. On the power of Ambainis's lower bounds. *Theoretical Computer Science*,339(2–3):241–256,2005.

# APPENDIX

# A. PROOFS FROM SECTION 3

## A.1 Proof of Lemma 2

The measurement of $\mathcal{H}_A$ decomposes the state in the $\mathcal{H}_I$ register as follows:

$$\rho = \sum_{a_1,\ldots,a_k \in \{0,1\}} p_{a_1,\ldots,a_k} \sigma_{a_1,\ldots,a_k},$$

with $p_{a_1,\ldots,a_k}$ being the probability of the measurement giving the answer $(a_1,\ldots,a_k)$ (where $a_j = 1$ means the algorithm outputs—not necessarily correctly—that $|x^j| = t$ and $a_j = 0$ means $|x^j| = t-1$) and $\sigma_{a_1,\ldots,a_k}$ being the density matrix of $\mathcal{H}_I$, conditional on this outcome of the measurement. Since the support of $\rho$ is contained in $\mathcal{S}_{0-} \oplus \cdots \oplus \mathcal{S}_{m-}$, the support of the states $\sigma_{a_1,\ldots,a_k}$ is also contained in $\mathcal{S}_{0-} \oplus \cdots \oplus \mathcal{S}_{m-}$. The probability that the answer $(a_1,\ldots,a_k)$ is correct is equal to

$$\operatorname{Tr} \Pi_{\otimes_{j=1}^k \oplus_{l=0}^{t-1+a_j} S_{l,a_j}} \sigma_{a_1,\ldots,a_k}. \tag{2}$$

We show that, for any $\sigma_{a_1,\ldots,a_k}$ with support contained in $\mathcal{S}_{0-} \oplus \cdots \oplus \mathcal{S}_{m-}$, (2) is at most $\frac{\sum_{m'=0}^{m} \binom{k}{m'}}{2^k}$.

For brevity, we now write $\sigma$ instead of $\sigma_{a_1,\ldots,a_k}$. A measurement w.r.t. $\otimes_{j=1}^k \oplus_l S_{l,a_j}$ and its orthogonal complement commutes with a measurement w.r.t. the collection of subspaces

$$\otimes_{j=1}^k (S_{l_j,0} \oplus S_{l_j,1}),$$

where $l_1,\ldots,l_k$ range over $\{0,\ldots,t\}$. Therefore

$$\operatorname{Tr} \Pi_{\otimes_{j=1}^k \oplus_l S_{l,a_j}} \sigma = \sum_{l_1,\ldots,l_k} \operatorname{Tr} \Pi_{\otimes_{j=1}^k \oplus_l S_{l,a_j}} \Pi_{\otimes_{j=1}^k (S_{l_j,0} \oplus S_{l_j,1})} \sigma.$$

| | |
|---|---|
| $\lvert\psi^a_{i_1,\ldots,i_j}\rangle$ | uniform superposition of states with $\lvert x\rvert = t-1+a$ and with $j$ fixed bits set to 1 |
| $T_{j,a}$ | spanned by $\lvert\psi^a_{i_1,\ldots,i_j}\rangle$ for all $j$-tuples $i$ |
| $S_{j,a} = T_{j,a} \cap T^{\perp}_{j-1,a}$ | that is, we remove the lower-dimensional subspace |
| $\lvert\tilde{\psi}^a_{i_1,\ldots,i_j}\rangle$ | projection of $\lvert\psi^a_{i_1,\ldots,i_j}\rangle$ onto $S_{j,a}$ |
| $S_{j,\pm}$ | spanned by $\frac{\lvert\tilde{\psi}^0\rangle}{\lVert\tilde{\psi}^0\rVert} \pm \frac{\lvert\tilde{\psi}^1\rangle}{\lVert\tilde{\psi}^1\rVert}$ |
| $R_j = S_{j,+}$ | for $j < \frac{t}{2}$ ... bad subspaces |
| $R_{t/2}$ | direct sum of $S_{j,+}$ for $j \geq t/2$, and all $S_{j,-}$ ... good subspaces |
| $\mathcal{S}_{m-} = \bigoplus_{\substack{\lvert r\rvert=m \\ j}} \bigotimes_{i=1}^{k} S_{j_i,r_i}$ | where $\lvert r\rvert$ is the number of minuses in $r = r_1,\ldots,r_k$ |
| $\mathcal{R}_m = \bigoplus_{\lvert j\rvert_1=m} \bigotimes_{i=1}^{k} R_{j_i}$ | where $\lvert j\rvert_1$ is the sum of all entries in $j = j_1,\ldots,j_k$ |
| $\mathcal{R}'_j = \bigoplus_{m\geq j} \mathcal{R}_m$ | |
| $\lvert\psi^{a,b}_{i_1,\ldots,i_j}\rangle$ | uniform superposition of states with $\lvert x\rvert = t-1+a$, with $j$ fixed bits set to 1, and $x_1 = b$ |
| $T_{j,a,b}$ | spanned by $\lvert\psi^{a,b}_{i_1,\ldots,i_j}\rangle$ for all $j$-tuples $i$ |
| $S_{j,a,b} = T_{j,a,b} \cap T^{\perp}_{j-1,a,b}$ | that is, we remove the lower-dimensional subspace |
| $\lvert\tilde{\psi}^{a,b}_{i_1,\ldots,i_j}\rangle$ | projection of $\lvert\psi^{a,b}_{i_1,\ldots,i_j}\rangle$ into $S_{j,a,b}$ |
| $S^{\alpha,\beta}_{j,a}$ | spanned by $\alpha\frac{\lvert\tilde{\psi}^{a,0}\rangle}{\lVert\tilde{\psi}^{a,0}\rVert} + \beta\frac{\lvert\tilde{\psi}^{a,1}\rangle}{\lVert\tilde{\psi}^{a,1}\rVert}$ |

**Table 2: States and subspaces used in the proof**

Hence to bound (2) it suffices to prove the same bound with

$$\sigma' = \Pi_{\otimes_{j=1}^{k}(S_{l_j,0}\oplus S_{l_j,1})}\sigma.$$

instead of $\sigma$. Since

$$\left(\otimes_{j=1}^{k}(S_{l_j,0} \oplus S_{l_j,1})\right) \cap \left(\otimes_{j=1}^{k}(\oplus_l S_{l,a_j})\right) = \otimes_{j=1}^{k} S_{l_j,a_j},$$

we have

$$\operatorname{Tr} \Pi_{\otimes_{j=1}^{k}(\oplus_l S_{l,a_j})}\sigma' = \operatorname{Tr} \Pi_{\otimes_{j=1}^{k} S_{l_j,a_j}}\sigma'. \tag{3}$$

We prove this bound for the case when $\sigma'$ is a pure state: $\sigma' = \lvert\psi\rangle\langle\psi\rvert$. Then equation (3) is equal to

$$\lVert\Pi_{\otimes_{j=1}^{k} S_{l_j,a_j}}\psi\rVert^2. \tag{4}$$

The bound for mixed states $\sigma'$ follows by decomposing $\sigma'$ as a mixture of pure states $\lvert\psi\rangle$, bounding (4) for each of those states and then summing up the bounds.

We have

$$(\mathcal{S}_{0-}\oplus\cdots\oplus\mathcal{S}_{m-})\cap(\bigotimes_{j=1}^{k}(S_{l_j,0}\oplus S_{l_j,1})) = \bigoplus_{\substack{r_1,\ldots,r_k\in\{+,-\}, \\ \lvert\{i:r_i=-\}\rvert\leq m}} \bigotimes_{j=1}^{k} S_{l_j,r_j}.$$

We express

$$\lvert\psi\rangle = \sum_{\substack{r_1,\ldots,r_k\in\{+,-\}, \\ \lvert\{i:r_i=-\}\rvert\leq m}} \alpha_{r_1,\ldots,r_k}\lvert\psi_{r_1,\ldots,r_k}\rangle,$$

with $\lvert\psi_{r_1,\ldots,r_k}\rangle \in \otimes_{j=1}^{k} S_{l_j,r_j}$. Therefore

$$\begin{aligned}\lVert\Pi_{\otimes_{j=1}^{k} S_{l_j,a_j}}\psi\rVert^2 &\leq \left(\sum_{r_1,\ldots,r_k} \lvert\alpha_{r_1,\ldots,r_k}\rvert\lVert\Pi_{\otimes_{j=1}^{k} S_{l_j,a_j}}\psi_{r_1,\ldots,r_k}\rVert\right)^2 \\ &\leq \sum_{r_1,\ldots,r_k} \lVert\Pi_{\otimes_{j=1}^{k} S_{l_j,a_j}}\psi_{r_1,\ldots,r_k}\rVert^2, \end{aligned} \tag{5}$$

where the second inequality uses Cauchy-Schwarz and

$$\lVert\psi\rVert^2 = \sum_{r_1,\ldots,r_k} \lvert\alpha_{r_1,\ldots,r_k}\rvert^2 = 1.$$

Claim 13. $\lVert\Pi_{\otimes_{j=1}^{k} S_{l_j,a_j}}\psi_{r_1,\ldots,r_k}\rVert^2 \leq \frac{1}{2^k}$.

Proof. Let $\lvert\varphi^{j,0}_i\rangle$, $i \in [\dim S_{l_j,0}]$ form a basis for the subspace $S_{l_j,0}$. Define a map $U_j : S_{l_j,0} \to S_{l_j,1}$ by $U_j\lvert\tilde{\psi}^0_{i_1,\ldots,i_{l_j}}\rangle = \lvert\tilde{\psi}^1_{i_1,\ldots,i_{l_j}}\rangle$. Then $U_j$ is a multiple of a unitary transformation: $U_j = c_jU'_j$ for some unitary $U'_j$ and a constant $c_j$. (This follows from Claim 16 in Appendix A.4.)

Let $\lvert\varphi^{j,1}_i\rangle = U'_j\lvert\varphi^{j,0}_i\rangle$. Since $U'_j$ is a unitary transformation, the states $\lvert\varphi^{j,1}_i\rangle$ form a basis for $S_{l_j,1}$. Therefore

$$\bigotimes_{j=1}^{k} \lvert\varphi^{j,a_j}_{i_j}\rangle \tag{6}$$

is a basis for $\otimes_{j=1}^{k} S_{l_j,a_j}$. Moreover, the states

$$\lvert\varphi^{j,+}_i\rangle = \frac{1}{\sqrt{2}}\lvert\varphi^{j,0}_i\rangle + \frac{1}{\sqrt{2}}\lvert\varphi^{j,1}_i\rangle, \quad \lvert\varphi^{j,-}_i\rangle = \frac{1}{\sqrt{2}}\lvert\varphi^{j,0}_i\rangle - \frac{1}{\sqrt{2}}\lvert\varphi^{j,1}_i\rangle$$

are a basis for $S_{l_j,+}$ and $S_{l_j,-}$, respectively. Therefore

$$\lvert\psi_{r_1,\ldots,r_k}\rangle = \sum_{i_1,\ldots,i_k} \alpha_{i_1,\ldots,i_k} \bigotimes_{j=1}^{k} \lvert\varphi^{j,r_j}_{i_j}\rangle. \tag{7}$$

The inner product between $\otimes_{i=1}^{k}\lvert\varphi^{j,a_j}_{i'_j}\rangle$ and $\otimes_{j=1}^{k}\lvert\varphi^{j,r_j}_{i_j}\rangle$ is

$$\prod_{j=1}^{k} \langle\varphi^{j,r_j}_{i_j}\lvert\varphi^{j,a_j}_{i'_j}\rangle.$$

Note that $r_j \in \{+,-\}$ and $a_j \in \{0,1\}$. The terms in this product are $\pm\frac{1}{\sqrt{2}}$ if $i'_j = i_j$ and 0 otherwise. This means that

$\otimes_{j=1}^{k}|\varphi_{i_j}^{j,r_j}\rangle$ has inner product $\pm\frac{1}{2^{k/2}}$ with $\otimes_{i=1}^{k}|\varphi_{i_j}^{j,a_j}\rangle$ and inner product 0 with all other basis states (6). Therefore,

$$\Pi_{\otimes_{j=1}^{k}S_{l_j,a_j}}\otimes_{j=1}^{k}|\varphi_{i_j}^{j,r_j}\rangle = \pm\frac{1}{2^{k/2}}\otimes_{i=1}^{k}|\varphi_{i_j}^{j,a_j}\rangle.$$

Together with equation (7), this means that

$$\|\Pi_{\otimes_{j=1}^{k}S_{l_j,a_j}}\psi_{r_1,\ldots,r_k}\| \le \frac{1}{2^{k/2}}\|\psi_{r_1,\ldots,r_k}\| = \frac{1}{2^{k/2}}.$$

Squaring both sides completes the proof of the claim. □

Since there are $\binom{k}{m'}$ tuples $(r_1,\ldots,r_k)$ with $r_1,\ldots,r_k \in \{+,-\}$ and $|\{i : r_i = -\}| = m'$, Claim 13 together with equation (5) implies

$$\|\Pi_{\otimes_{j=1}^{k}S_{l_j,a_j}}\psi\|^2 \le \frac{\sum_{m'=0}^{m}\binom{k}{m'}}{2^k}.$$

## A.2 Proof of Corollary 3

Let $|\psi\rangle$ be a purification of $\rho$ in $\mathcal{H}_A\otimes\mathcal{H}_I$. Let

$$|\psi\rangle = \sqrt{1-\delta}|\psi'\rangle + \sqrt{\delta}|\psi''\rangle$$

where $|\psi'\rangle$ is in the subspace $\mathcal{H}_A\otimes(\mathcal{S}_{0-}\oplus\mathcal{S}_{1-}\oplus\cdots\oplus\mathcal{S}_{m-})$ and $|\psi''\rangle$ is in the subspace $\mathcal{H}_A\otimes(\mathcal{S}_{0-}\oplus\mathcal{S}_{1-}\oplus\cdots\oplus\mathcal{S}_{m-})^{\perp}$. Then $\delta = \operatorname{Tr}\Pi_{(\mathcal{S}_{0-}\oplus\cdots\oplus\mathcal{S}_{m-})^{\perp}}\rho$.

The success probability of $\mathcal{A}$ is the probability that, if we measure both the register $\mathcal{H}_A$ containing the result of the computation and $\mathcal{H}_I$, then we get $a_1,\ldots,a_k$ and $x^1,\ldots,x^k$ such that $x^j$ contains $t-1+a_j$ ones for every $j\in\{1,\ldots,k\}$.

Consider the probability of getting $a_1,\ldots,a_k\in\{0,1\}$ and $x^1,\ldots,x^k\in\{0,1\}^n$ with this property, when measuring $|\psi'\rangle$ (instead of $|\psi\rangle$). By Lemma 2, this probability is at most $\frac{\sum_{m'=0}^{m}\binom{k}{m'}}{2^k}$. We have

$$\begin{aligned}\|\psi-\psi'\| &\le (1-\sqrt{1-\delta})\|\psi'\| + \sqrt{\delta}\|\psi''\| \\ &= (1-\sqrt{1-\delta}) + \sqrt{\delta} \le 2\sqrt{\delta}.\end{aligned}$$

We now apply

LEMMA 14 ([9]). *For any states $|\psi\rangle$ and $|\psi'\rangle$ and any measurement $M$, the variational distance between the probability distributions obtained by applying $M$ to $|\psi\rangle$ and $|\psi'\rangle$ is at most $2\|\psi-\psi'\|$.*

Hence the success probability of $\mathcal{A}$ is at most

$$\frac{\sum_{m'=0}^{m}\binom{k}{m'}}{2^k}+4\sqrt{\delta} = \frac{\sum_{m'=0}^{m}\binom{k}{m'}}{2^k}+4\sqrt{\operatorname{Tr}\Pi_{(\mathcal{S}_{0-}\oplus\cdots\oplus\mathcal{S}_{m-})^{\perp}}\rho}.$$

## A.3 Structure of the subspaces when asking one query

Let $|\psi_d\rangle$ be the state of $\mathcal{H}_A\otimes\mathcal{H}_I$ after $d$ queries. Write

$$|\psi_d\rangle = \sum_{i=0}^{kn} a_i|\psi_{d,i}\rangle,$$

with $|\psi_{d,i}\rangle$ being the part in which the query register contains $|i\rangle$. Let $\rho_{d,i} = \operatorname{Tr}_{\mathcal{H}_A}|\psi_{d,i}\rangle\langle\psi_{d,i}|$. Then

$$\rho_d = \sum_{i=0}^{kn} a_i^2\rho_{d,i}. \tag{8}$$

Because of

$$\operatorname{Tr}\Pi_{\mathcal{R}_m}\rho_d = \sum_{i=0}^{kn} a_i^2\operatorname{Tr}\Pi_{\mathcal{R}_m}\rho_{d,i},$$

we have $P(\rho_d) = \sum_{i=0}^{kn} a_i^2 P(\rho_{d,i})$. Let $\rho'_d$ be the state after the $d$-th query and let $\rho'_d = \sum_{i=0}^{kn} a_i^2\rho'_{d,i}$ be a decomposition similar to equation (8). Lemma 4 follows by showing

$$P(\rho'_{d,i}) \le \left(1+\frac{C}{\sqrt{tn}}(q^{t/2}-1)+\frac{C\sqrt{t}}{\sqrt{n}}(q-1)\right)P(\rho_{d,i}) \tag{9}$$

for each $i$. For $i=0$, the query does not change the state if the query register contains $|i\rangle$. Therefore, $\rho'_{d,0}=\rho_{d,0}$ and $P(\rho'_{d,0})=P(\rho_{d,0})$. This means that equation (9) is true for $i=0$. To prove the $i\in\{1,\ldots,kn\}$ case, it suffices to prove the $i=1$ case (because of symmetry).

Let $|\psi^{a,b}_{i_1,\ldots,i_j}\rangle$ (with $a,b\in\{0,1\}$ and $i_1,\ldots,i_j\in\{2,\ldots,n\}$) be the uniform superposition over basis states $|b,x_2,\ldots,x_n\rangle$ (of $\mathcal{H}_{one}$) with $b+x_2+\cdots+x_n = t-1+a$ and $x_{i_1}=\cdots=x_{i_j}=1$. Let $T_{j,a,b}$ be the space spanned by all states $|\psi^{a,b}_{i_1,\ldots,i_j}\rangle$ and let $S_{j,a,b} = T_{j,a,b}\cap T^{\perp}_{j-1,a,b}$. Let $|\tilde{\psi}^{a,b}_{i_1,\ldots,i_j}\rangle = \Pi_{T^{\perp}_{j-1,a,b}}|\psi^{a,b}_{i_1,\ldots,i_j}\rangle$.

Let $S^{\alpha,\beta}_{j,a}$ be the subspace spanned by all states

$$\alpha\frac{|\tilde{\psi}^{a,0}_{i_1,\ldots,i_j}\rangle}{\|\tilde{\psi}^{a,0}_{i_1,\ldots,i_j}\|}+\beta\frac{|\tilde{\psi}^{a,1}_{i_1,\ldots,i_j}\rangle}{\|\tilde{\psi}^{a,1}_{i_1,\ldots,i_j}\|}. \tag{10}$$

CLAIM 15. *Let $\alpha_a = \sqrt{\frac{n-(t-1+a)}{n-j}}\|\tilde{\psi}^{a,0}_{i_1,\ldots,i_j}\|$ and $\beta_a = \sqrt{\frac{(t-1+a)-j}{n-j}}\|\tilde{\psi}^{a,1}_{i_1,\ldots,i_j}\|$. Then (i) $S^{\alpha_a,\beta_a}_{j,a}\subseteq S_{j,a}$ and (ii) $S^{\beta_a,-\alpha_a}_{j,a}\subseteq S_{j+1,a}$.*

PROOF. For part (i), consider the states $|\psi^{a}_{i_1,\ldots,i_j}\rangle$ in $T_{j,a}$, for $1\notin\{i_1,\ldots,i_j\}$. We have

$$\begin{aligned}|\psi^{a}_{i_1,\ldots,i_j}\rangle = &\sqrt{\frac{n-(t-1+a)}{n-j}}|\psi^{a,0}_{i_1,\ldots,i_j}\rangle \\ &+\sqrt{\frac{(t-1+a)-j}{n-j}}|\psi^{a,1}_{i_1,\ldots,i_j}\rangle\end{aligned} \tag{11}$$

because among the states $|x_1\ldots x_n\rangle$ with $|x| = t-1+a$ and $x_{i_1}=\cdots=x_{i_j}=1$, a $\frac{n-(t-1+a)}{n-j}$ fraction have $x_1=0$ and the rest have $x_1=1$. The projections of these states to $T^{\perp}_{j-1,a,0}\cap T^{\perp}_{j-1,a,1}$ are

$$\sqrt{\frac{n-(t-1+a)}{n-j}}|\tilde{\psi}^{a,0}_{i_1,\ldots,i_j}\rangle+\sqrt{\frac{(t-1+a)-j}{n-j}}|\tilde{\psi}^{a,1}_{i_1,\ldots,i_j}\rangle$$

which, by equation (10) are exactly the states spanning $S^{\alpha_a,\beta_a}_{j,a}$. Furthermore, we claim that

$$T_{j-1,a}\subseteq T_{j-1,a,0}\oplus T_{j-1,a,1}\subseteq T_{j,a}. \tag{12}$$

The first containment is true because $T_{j-1,a}$ is spanned by the states $|\psi^{a}_{i_1,\ldots,i_{j-1}}\rangle$ which either belong to $T_{j-2,a,1}\subseteq T_{j-1,a,1}$ (if $1\in\{i_1,\ldots,i_{j-1}\}$) or are a linear combination of states $|\psi^{a,0}_{i_1,\ldots,i_{j-1}}\rangle$ and $|\psi^{a,1}_{i_1,\ldots,i_{j-1}}\rangle$ (by equation (11)), which belong to $T_{j-1,a,0}$ and $T_{j-1,a,1}$. The second containment follows because the states $|\psi^{a,1}_{i_1,\ldots,i_{j-1}}\rangle$ spanning $T_{j-1,a,1}$ are the same as the states $|\psi^{a}_{1,i_1,\ldots,i_{j-1}}\rangle$ which belong to $T_{j,a}$, and the states $|\psi^{a,0}_{i_1,\ldots,i_{j-1}}\rangle$ spanning $T_{j-1,a,0}$ can be expressed as linear combinations of $|\psi^{a}_{i_1,\ldots,i_{j-1}}\rangle$ and $|\psi^{a}_{1,i_1,\ldots,i_{j-1}}\rangle$ which both belong to $T_{j,a}$.

The first part of (12) now implies

$$S^{\alpha_a,\beta_a}_{j,a}\subseteq T^{\perp}_{j-1,a,0}\cap T^{\perp}_{j-1,a,1}\subseteq T^{\perp}_{j-1,a}.$$

Also, $S_{j,a}^{\alpha_a,\beta_a} \subseteq T_{j,a}$, because $S_{j,a}^{\alpha_a,\beta_a}$ is spanned by the states

$$\Pi_{T_{j-1,a,0}^{\perp} \cap T_{j-1,a,1}^{\perp}} |\psi_{i_1,\ldots,i_j}^a\rangle = |\psi_{i_1,\ldots,i_j}^a\rangle - \Pi_{T_{j-1,a,0} \oplus T_{j-1,a,1}} |\psi_{i_1,\ldots,i_j}^a\rangle$$

and $|\psi_{i_1,\ldots,i_j}^a\rangle$ belongs to $T_{j,a}$ by the definition of $T_{j,a}$ and $\Pi_{T_{j-1,a,0} \oplus T_{j-1,a,1}} |\psi_{i_1,\ldots,i_j}^a\rangle$ belongs to $T_{j,a}$ because of the second part of (12). Therefore, $S_{j,a}^{\alpha_a,\beta_a} \subseteq T_{j,a} \cap T_{j-1,a}^{\perp} = S_{j,a}$.

For part (ii), we have

$$S_{j,a}^{\alpha_a,\beta_a} \subseteq S_{j,a,0} \oplus S_{j,a,1} \subseteq T_{j,a,0} \oplus T_{j,a,1} \subseteq T_{j+1,a},$$

where the first containment is true because $S_{j,a}^{\alpha_a,\beta_a}$ is spanned by linear combinations of vectors $|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\rangle$ (which belong to $S_{j,a,0}$) and vectors $|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\rangle$ (which belong to $S_{j,a,1}$) and the last containment is true because of the second part of equation (12). Now let

$$|\psi\rangle = \beta_a \frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\|} - \alpha_a \frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\|} \tag{13}$$

be one of the vectors spanning $S_{j,a}^{\beta_a,-\alpha_a}$. To prove that $|\psi\rangle$ is in $S_{j+1,a} = T_{j+1,a} \cap T_{j,a}^{\perp}$, it remains to prove that $|\psi\rangle$ is orthogonal to $T_{j,a}$. This is equivalent to proving that $|\psi\rangle$ is orthogonal to each of the vectors $|\psi_{i'_1,\ldots,i'_j}^a\rangle$ spanning $T_{j,a}$. We distinguish two cases (note that $1 \notin \{i_1,\ldots,i_j\}$):

**Case 1.** $1 \in \{i'_1,\ldots,i'_j\}$.

For simplicity, assume $1 = i'_j$. Then $|\psi_{i'_1,\ldots,i'_j}^a\rangle$ is the same as $|\psi_{i'_1,\ldots,i'_{j-1}}^{a,1}\rangle$, which belongs to $T_{j-1,a,1}$. By definition, the vector $|\psi\rangle$ belongs to $T_{j-1,a,0}^{\perp} \cap T_{j-1,a,1}^{\perp}$ and is therefore orthogonal to $|\psi_{i'_1,\ldots,i'_{j-1}}^{a,1}\rangle$.

**Case 2.** $1 \notin \{i'_1,\ldots,i'_j\}$.

We will prove this case by induction on $\ell = |\{i'_1,\ldots,i'_j\} - \{i_1,\ldots,i_j\}|$.

In the base step ($\ell = 0$), we have $\{i'_1,\ldots,i'_j\} = \{i_1,\ldots,i_j\}$. Since $|\psi\rangle$ belongs to $T_{j-1,a,0}^{\perp} \cap T_{j-1,a,1}^{\perp}$, it suffices to prove $|\psi\rangle$ is orthogonal to the projection of $|\psi_{i_1,\ldots,i_j}^a\rangle$ to $T_{j-1,a,0}^{\perp} \cap T_{j-1,a,1}^{\perp}$ which, by the discussion after equation (11), equals

$$\alpha_a \frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\|} + \beta_a \frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\|}. \tag{14}$$

From equations (13) and (14), we see that the inner product of the two states is $\alpha_a\beta_a - \beta_a\alpha_a = 0$.

For the inductive step ($\ell \geq 1$), assume $i'_j \notin \{i_1,\ldots,i_j\}$. Up to renormalization, we have

$$|\psi_{i'_1,\ldots,i'_{j-1}}^a\rangle = \sum_{i' \notin \{i'_1,\ldots,i'_{j-1}\}} |\psi_{i'_1,\ldots,i'_{j-1},i'}^a\rangle.$$

Because $|\psi_{i'_1,\ldots,i'_{j-1}}^a\rangle$ is in $T_{j-1,a,0} \oplus T_{j-1,a,1}$, we have

$$\sum_{i' \notin \{i'_1,\ldots,i'_{j-1}\}} \langle\psi_{i'_1,\ldots,i'_{j-1},i'}^a|\psi\rangle = \langle\psi_{i'_1,\ldots,i'_{j-1}}^a|\psi\rangle = 0. \tag{15}$$

As proven in the previous case, $\langle\psi_{i'_1,\ldots,i'_{j-1},1}^a|\psi\rangle = 0$. Moreover, by the induction hypothesis we have $\langle\psi_{i'_1,\ldots,i'_{j-1},i'}^a|\psi\rangle = 0$ whenever $i' \in \{i_1,\ldots,i_j\}$. Therefore equation (15) reduces to

$$\sum_{i' \notin \{i'_1,\ldots,i'_{j-1},i_1,\ldots,i_j,1\}} \langle\psi_{i'_1,\ldots,i'_{j-1},i'}^a|\psi\rangle = 0. \tag{16}$$

By symmetry, the inner products in this sum are the same for every $i'$. Hence they are all 0, in particular for $i' = i'_j$. □

## A.4 Proof of Lemma 4

CLAIM 16. *The maps $U_{01} : S_{j,0,0} \to S_{j,0,1}$, $U_{10} : S_{j,0,0} \to S_{j,1,0}$ and $U_{11} : S_{j,0,0} \to S_{j,1,1}$ defined by $U_{ab}|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle = |\tilde{\psi}_{i_1,\ldots,i_j}^{a,b}\rangle$ are multiples of unitary transformations: $U_{ab} = c_{ab}U'_{ab}$ for some unitary $U'_{ab}$ and some constant $c_{ab}$.*

PROOF. We define $M : T_{j,0,0} \to T_{j,0,1}$ by

$$M|0x_2\ldots x_n\rangle = \sum_{\ell:x_\ell=1} |1x_2\ldots x_{\ell-1}0x_{\ell+1}\ldots x_n\rangle.$$

Note that $M$ does not depend on $j$. We claim

$$\begin{aligned} M|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle &= c|\tilde{\psi}_{i_1,\ldots,i_j}^{0,1}\rangle, \\ M^\dagger|\tilde{\psi}_{i_1,\ldots,i_j}^{0,1}\rangle &= c'|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle, \end{aligned} \tag{17}$$

for some constants $c$ and $c'$ that may depend on $n, t$ and $j$ but not on $i_1,\ldots,i_j$. To prove that, we need to prove two things. First, we claim that

$$M|\psi_{i_1,\ldots,i_j}^{0,0}\rangle = c|\psi_{i_1,\ldots,i_j}^{0,1}\rangle + |\psi'\rangle, \tag{18}$$

where $|\psi'\rangle \in T_{j-1,0,1}$ (note that $1 \notin \{i_1,\ldots,i_j\}$). Equation (18) follows by

$$\begin{aligned} M|\psi_{i_1,\ldots,i_j}^{0,0}\rangle &= \frac{1}{\sqrt{\binom{n-j-1}{t-1-j}}} \sum_{\substack{x:|x|=t-1,x_1=0 \\ x_{i_1}=\cdots=x_{i_j}=1,}} M|x\rangle \\ &= \frac{1}{\sqrt{\binom{n-j-1}{t-1-j}}} \sum_{\substack{x:|x|=t-1,x_1=0 \\ x_{i_1}=\cdots=x_{i_j}=1}} \sum_{\ell:x_\ell=1} |1x_2\ldots x_{\ell-1}0x_{\ell+1}\ldots x_n\rangle \\ &= \frac{n-t+1}{\sqrt{\binom{n-j-1}{t-1-j}}} \sum_{\substack{y:|y|=t-1,y_1=1 \\ y_{i_1}=\cdots=y_{i_j}=1}} |y\rangle \\ &\quad + \frac{1}{\sqrt{\binom{n-j-1}{t-1-j}}} \sum_{\ell=1}^{j} \sum_{\substack{y:|y|=t-1,y_1=1,y_{i_\ell}=0 \\ y_{i_1}=\cdots=y_{i_j}=1}} |y\rangle \\ &= \frac{n-t-j+1}{\sqrt{\binom{n-j-1}{t-1-j}}} \sum_{\substack{y:|y|=t-1,y_1=1 \\ y_{i_1}=\cdots=y_{i_j}=1}} |y\rangle \\ &\quad + \frac{1}{\sqrt{\binom{n-j-1}{t-1-j}}} \sum_{\ell=1}^{j} \sum_{\substack{y:|y|=t-1,y_1=1 \\ y_{i_1}=\cdots=y_{i_{\ell-1}}=1 \\ y_{i_{\ell+1}}=\cdots=y_{i_j}=1}} |y\rangle \\ &= (n-t-j+1)\sqrt{\frac{t-1-j}{n-t+1}}|\psi_{i_1,\ldots,i_j}^{0,1}\rangle \\ &\quad + \sqrt{\frac{n-j}{n-t+1}} \sum_{\ell=1}^{j} |\psi_{i_1,\ldots,i_{\ell-1},i_{\ell+1},\ldots,i_j}^{0,1}\rangle. \end{aligned}$$

This proves (18), with $|\psi'\rangle$ equal to the second term.

Second, for every $j$, $M(T_{j,0,0}) \subseteq T_{j,0,1}$ and $M(T_{j,0,0}^{\perp}) \subseteq T_{j,0,1}^{\perp}$. The first statement follows from equation (18), because the subspaces $T_{j,0,0}$, $T_{j,0,1}$ are spanned by the states $|\psi_{i_1,\ldots,i_j}^{0,0}\rangle$ and $|\psi_{i_1,\ldots,i_j}^{0,1}\rangle$, respectively, and $T_{j-1,0,1} \subseteq T_{j,0,1}$. To prove the second statement, let $|\psi\rangle \in T_{j,0,0}^{\perp}$, $|\psi\rangle = \sum_x a_x|x\rangle$. We would like to prove $M|\psi\rangle \in T_{j,0,1}^{\perp}$. This is equivalent to $\langle\psi_{i_1,\ldots,i_j}^{0,1}|M|\psi\rangle = 0$ for all $i_1,\ldots,i_j$. We have

$$\begin{aligned}\langle\psi_{i_1,\ldots,i_j}^{0,1}|M|\psi\rangle &= \frac{1}{\sqrt{\binom{n-j-1}{t-j-2}}} \sum_{\substack{y:|y|=t-1,y_1=1\\ y_{i_1}=\cdots=y_{i_j}=1}} \langle y|M|\psi\rangle \\ &= \frac{1}{\sqrt{\binom{n-j-1}{t-j-2}}} \sum_{\substack{x:|x|=t-1,x_1=0\\ x_{i_1}=\cdots=x_{i_j}=1}} \sum_{\substack{\ell:x_\ell=1\\ \ell\notin\{i_1,\ldots,i_j\}}} a_x \\ &= \frac{t-1-j}{\sqrt{\binom{n-j-1}{t-j-2}}} \sum_{\substack{x:|x|=t-1,x_1=0\\ x_{i_1}=\cdots=x_{i_j}=1}} a_x = 0.\end{aligned}$$

The first equality follows by writing out $\langle\psi_{i_1,\ldots,i_j}^{0,1}|$, the second equality follows by writing out $M$. The third equality follows because, for every $x$ with $|x| = t-1$ and $x_{i_1} = \cdots = x_{i_j} = 1$, there are $t-1-j$ more $\ell \in [n]$ satisfying $x_\ell = 1$. The fourth equality follows because $\sum_{\substack{x:|x|=t-1,x_1=0\\ x_{i_1}=\cdots=x_{i_j}=1}} a_x$ is a constant times $\langle\psi_{i_1,\ldots,i_j}^{0,0}|\psi\rangle$, and $\langle\psi_{i_1,\ldots,i_j}^{0,0}|\psi\rangle = 0$ because $|\psi\rangle \in T_{j,0,0}^{\perp}$.

To deduce equation (17), we write

$$|\psi_{i_1,\ldots,i_j}^{0,0}\rangle = |\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle + \Pi_{T_{j-1,0,0}}|\psi_{i_1,\ldots,i_j}^{0,0}\rangle.$$

Since $M(T_{j-1,0,0}) \subseteq T_{j-1,0,1}$ and $M(T_{j-1,0,0}^{\perp}) \subseteq T_{j-1,0,1}^{\perp}$,

$$\begin{aligned}M|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle &= \Pi_{T_{j-1,0,1}^{\perp}} M|\psi_{i_1,\ldots,i_j}^{0,0}\rangle \\ &= c\Pi_{T_{j-1,0,1}^{\perp}}|\psi_{i_1,\ldots,i_j}^{0,1}\rangle = c|\tilde{\psi}_{i_1,\ldots,i_j}^{0,1}\rangle,\end{aligned}$$

with the second equality following from (18) and $|\psi'\rangle \in T_{j-1,0,1}$. This proves the first half of (17). The second half follows similarly. Therefore

$$\langle\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}|M^\dagger M|\tilde{\psi}_{i'_1,\ldots,i'_j}^{0,0}\rangle = c\cdot c'\langle\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}|\tilde{\psi}_{i'_1,\ldots,i'_j}^{0,0}\rangle.$$

Hence $M$ is a multiple of a unitary transformation. By equation (17), $U_{01} = M/c$ and, therefore, $U_{01}$ is also a multiple of a unitary transformation.

Next, we define $M$ by $M|0x_2\ldots x_n\rangle = |1x_2\ldots x_n\rangle$. Then $M$ is a unitary transformation from the space spanned by $|0x_2\ldots x_n\rangle$, $x_2+\cdots+x_2 = t-1$, to the space spanned by $|1x_2\ldots x_n\rangle$, $1+x_2+\cdots+x_n = t$. We claim that $U_{11} = M$. To prove that, we first observe that

$$\begin{aligned}M|\psi_{i_1,\ldots,i_j}^{0,0}\rangle &= \frac{1}{\sqrt{\binom{n-j-1}{t-j-1}}} \sum_{\substack{x_2,\ldots,x_n:\\ x_{i_1}=\cdots=x_{i_j}=1}} M|0x_2\ldots x_n\rangle \\ &= \frac{1}{\sqrt{\binom{n-j-1}{t-j-1}}} \sum_{\substack{x_2,\ldots,x_n:\\ x_{i_1}=\cdots=x_{i_j}=1}} |1x_2\ldots x_n\rangle = |\psi_{i_1,\ldots,i_j}^{1,1}\rangle.\end{aligned}$$

Since $T_{j,a,b}$ is defined as the subspace spanned by all $|\psi_{i_1,\ldots,i_j}^{a,b}\rangle$, this means that $M(T_{j,0,0}) = T_{j,1,1}$ and similarly $M(T_{j-1,0,0}) = T_{j-1,1,1}$. Since $M$ is unitary, this implies $M(T_{j-1,0,0}^{\perp}) = T_{j-1,1,1}^{\perp}$ and

$$\begin{aligned}M|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle &= M\Pi_{T_{j-1,0,0}^{\perp}}|\psi_{i_1,\ldots,i_j}^{0,0}\rangle \\ &= \Pi_{T_{j-1,1,1}^{\perp}}|\psi_{i_1,\ldots,i_j}^{1,1}\rangle = |\tilde{\psi}_{i_1,\ldots,i_j}^{1,1}\rangle.\end{aligned}$$

Finally, we have $U_{10} = U''_{10}U_{11}$, where $U''_{10}$ is defined by $U''_{10}|\tilde{\psi}_{i_1,\ldots,i_j}^{1,1}\rangle = |\tilde{\psi}_{i_1,\ldots,i_j}^{1,0}\rangle$. Since $U_{11}$ is unitary, it suffices to prove that $U''_{10}$ is a multiple of a unitary transformation and this follows similarly to $U_{01}$ being a multiple of a unitary transformation. □

Let $|\psi_{00}\rangle$ be an arbitrary state in $S_{j,0,0}$ for some $j \in \{0,\ldots,t-1\}$. Define $|\psi_{ab}\rangle = U'_{ab}|\psi_{00}\rangle$ for $ab \in \{01,10,11\}$. Let $|\psi_2\rangle,\ldots,|\psi_k\rangle$ be vectors from subspaces $R_{j_2},\ldots,R_{j_k}$, for some $j_2,\ldots,j_k$. We first analyze the case when $\rho_{d,1}$ belongs to the subspace $\mathcal{H}_4$ spanned by $|\psi_{ab}\rangle\otimes|\psi_2\rangle\otimes\cdots\otimes|\psi_k\rangle$.

Claim 17. *Let* $\alpha'_a = \sqrt{\frac{n-(t-1+a)}{n-j}}\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\|$, $\beta'_a = \sqrt{\frac{(t-1+a)-j}{n-j}}\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\|$, $\alpha_a = \frac{\alpha'_a}{\sqrt{(\alpha'_a)^2+(\beta'_a)^2}}$, $\beta_a = \frac{\beta'_a}{\sqrt{(\alpha'_a)^2+(\beta'_a)^2}}$. *Then*

1. $|\phi_1\rangle = \alpha_0|\psi_{00}\rangle + \beta_0|\psi_{01}\rangle + \alpha_1|\psi_{10}\rangle + \beta_1|\psi_{11}\rangle$ *belongs to* $S_{j,+}$;
2. $|\phi_2\rangle = \beta_0|\psi_{00}\rangle - \alpha_0|\psi_{01}\rangle + \beta_1|\psi_{10}\rangle - \alpha_1|\psi_{11}\rangle$ *belongs to* $S_{j+1,+}$;
3. *Any linear combination of* $|\psi_{00}\rangle$, $|\psi_{01}\rangle$, $|\psi_{10}\rangle$ *and* $|\psi_{11}\rangle$ *which is orthogonal to* $|\phi_1\rangle$ *and* $|\phi_2\rangle$ *belongs to* $S_- = \bigoplus_{j=0}^{t} S_{j,-}$.

Proof. Let $i_1,\ldots,i_j$ be $j$ distinct elements of $\{2,\ldots,n\}$. As shown in the beginning of the proof of Claim 15,

$$\begin{aligned}|\tilde{\psi}_{i_1,\ldots,i_j}^{a}\rangle &= \sqrt{\frac{n-(t-1+a)}{n-j}}|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\rangle \\ &\quad + \sqrt{\frac{(t-1+a)-j}{n-j}}|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\rangle \\ &= \alpha'_a\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\|} + \beta'_a\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\|}.\end{aligned}$$

This means that $\|\tilde{\psi}_{i_1,\ldots,i_j}^{a}\| = \sqrt{(\alpha'_a)^2+(\beta'_a)^2}$ and

$$\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a}\|} = \alpha_a\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,0}\|} + \beta_a\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,1}\|}.$$

Since the states $|\tilde{\psi}_{i_1,\ldots,i_j}^{0}\rangle$ span $S_{j,0}$, the state $|\psi_{00}\rangle$ is a linear combination of states $\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\|}$. By Claim 16, the states $|\psi_{ab}\rangle$ are linear combinations of $\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{a,b}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{a,b}\|}$ with the same coefficients. Therefore, $|\phi_1\rangle$ is a linear combination of

$$\alpha_0\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{0,0}\|} + \beta_0\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{0,1}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{0,1}\|} + \alpha_1\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{1,0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{1,0}\|} + \beta_1\frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{1,1}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{1,1}\|}$$

$$= \frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{0}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{0}\|} + \frac{|\tilde{\psi}_{i_1,\ldots,i_j}^{1}\rangle}{\|\tilde{\psi}_{i_1,\ldots,i_j}^{1}\|},$$

each of which, by definition, belongs to $S_{j,+}$.

Let $i_1, \dots, i_j$ be distinct elements of $\{2, \dots, n\}$. We claim

$$\frac{|\tilde{\psi}^a_{1,i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^a_{1,i_1,\dots,i_j}\|} = \beta_a \frac{|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\|} - \alpha_a \frac{|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\|}. \tag{19}$$

By Claim 15, the right hand side of (19) belongs to $S_{j+1,a}$. We need to show that it is equal to $|\tilde{\psi}^a_{1,i_1,\dots,i_j}\rangle$. We have

$$\begin{aligned}|\tilde{\psi}^a_{1,i_1,\dots,i_j}\rangle &= \Pi_{T^\perp_{j,a}}|\psi^a_{1,i_1,\dots,i_j}\rangle = \Pi_{T^\perp_{j,a}}|\psi^{a,1}_{i_1,\dots,i_j}\rangle \\ &= \Pi_{T^\perp_{j,a}}\Pi_{T^\perp_{j-1,a,1}}|\psi^{a,1}_{i_1,\dots,i_j}\rangle = \Pi_{T^\perp_{j,a}}|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\rangle,\end{aligned}$$

where the third equality follows from $T_{j-1,a,1} \subseteq T_{j,a}$. This is because the states $|\psi^{a,1}_{i_1,\dots,i_{j-1}}\rangle$ spanning $T_{j-1,a,1}$ are the same as the states $|\psi^a_{1,i_1,\dots,i_{j-1}}\rangle$ in $T_{j,a}$. Write

$$|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\rangle = c_1|\delta_1\rangle + c_2|\delta_2\rangle$$

where

$$\begin{aligned}|\delta_1\rangle &= \alpha_a \frac{|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\|} + \beta_a \frac{|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\|}, \\ |\delta_2\rangle &= \beta_a \frac{|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\|} - \alpha_a \frac{|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\|}.\end{aligned}$$

By Claim 15, we have $|\delta_1\rangle \in S_{j,a} \subseteq T_{j,a}$, $|\delta_2\rangle \in S_{j+1,a} \subseteq T^\perp_{j,a}$. Therefore, $\Pi_{T^\perp_{j,a}}|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\rangle = c_2|\delta_2\rangle$ and

$$\frac{|\tilde{\psi}^a_{1,i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^a_{1,i_1,\dots,i_j}\|} = |\delta_2\rangle = \beta_a \frac{|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\|} - \alpha_a \frac{|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\|},$$

proving (19).

Similarly to the argument for $|\phi_1\rangle$, equation (19) implies that $|\phi_2\rangle$ is a linear combination of

$$\beta_0 \frac{|\tilde{\psi}^{0,0}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{0,0}_{i_1,\dots,i_j}\|} - \alpha_0 \frac{|\tilde{\psi}^{0,1}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{0,1}_{i_1,\dots,i_j}\|} + \beta_1 \frac{|\tilde{\psi}^{1,0}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{1,0}_{i_1,\dots,i_j}\|} - \alpha_1 \frac{|\tilde{\psi}^{1,1}_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^{1,1}_{i_1,\dots,i_j}\|}$$

$$= \frac{|\tilde{\psi}^0_{1,i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^0_{1,i_1,\dots,i_j}\|} + \frac{|\tilde{\psi}^1_{1,i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^1_{1,i_1,\dots,i_j}\|}$$

and each of those states belongs to $S_{j+1,+}$.

To prove the third part of Claim 17, we observe that any vector orthogonal to $|\phi_1\rangle$ and $|\phi_2\rangle$ is a linear combination of

$$|\phi_3\rangle = \alpha_0|\psi_{00}\rangle + \beta_0|\psi_{01}\rangle - \alpha_1|\psi_{10}\rangle - \beta_1|\psi_{11}\rangle,$$

which, in turn, is a linear combination of vectors

$$\frac{|\tilde{\psi}^0_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^0_{i_1,\dots,i_j}\|} - \frac{|\tilde{\psi}^1_{i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^1_{i_1,\dots,i_j}\|}$$

and

$$|\phi_4\rangle = \beta_0|\psi_{00}\rangle - \alpha_0|\psi_{01}\rangle - \beta_1|\psi_{10}\rangle + \alpha_1|\psi_{11}\rangle$$

which is a linear combination of vectors

$$\frac{|\tilde{\psi}^0_{1,i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^0_{1,i_1,\dots,i_j}\|} - \frac{|\tilde{\psi}^1_{1,i_1,\dots,i_j}\rangle}{\|\tilde{\psi}^1_{1,i_1,\dots,i_j}\|}.$$

This means that we have $|\phi_3\rangle \in S_{j,-}$ and $|\phi_4\rangle \in S_{j+1,-}$. □

Claim 18. *Let $j < t/2$ and $x^{\underline{j}} = x(x-1)\cdots(x-j+1)$.*

1. $\|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\| = \sqrt{\dfrac{(n-t-a+b)^{\underline{j}}}{(n-j)^{\underline{j}}}}$.
2. $\|\tilde{\psi}^{a,0}_{i_1,\dots,i_j}\| \geq \dfrac{1}{\sqrt{2}}\|\tilde{\psi}^{a,1}_{i_1,\dots,i_j}\|$.
3. $\dfrac{\|\tilde{\psi}^{0,0}_{i_1,\dots,i_j}\| \cdot \|\tilde{\psi}^{1,1}_{i_1,\dots,i_j}\|}{\|\tilde{\psi}^{0,1}_{i_1,\dots,i_j}\| \cdot \|\tilde{\psi}^{1,0}_{i_1,\dots,i_j}\|} = 1 + O\left(\dfrac{1}{t}\right)$.

Proof. Define $t_a = t - 1 + a$. We calculate the vector

$$|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\rangle = \Pi_{T^\perp_{j-1,a,b}}|\psi^{a,b}_{i_1,\dots,i_j}\rangle.$$

Both vector $|\psi^{a,b}_{i_1,\dots,i_j}\rangle$ and subspace $T_{j-1,a,b}$ are fixed by

$$U_\pi|x\rangle = |x_{\pi(1)}\dots x_{\pi(n)}\rangle$$

for any permutation $\pi$ that fixes 1 and maps $\{i_1, \dots, i_j\}$ to itself. Hence $|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\rangle$ is fixed by any such $U_\pi$ as well. Therefore, the amplitude of $|x\rangle$ with $|x| = t_a$, $x_1 = b$ in $|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\rangle$ only depends on $|\{i_1, \dots, i_j\} \cap \{i : x_i = 1\}|$, so $|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\rangle$ is of the form

$$|\upsilon_{a,b}\rangle = \sum_{m=0}^{j} \kappa_m \sum_{\substack{x:|x|=t_a,x_1=b \\ |\{i_1,\dots,i_j\}\cap\{i:x_i=1\}|=m}} |x\rangle.$$

To simplify the following calculations, we multiply $\kappa_0, \dots, \kappa_j$ by the same constant so that $\kappa_j = 1/\sqrt{\binom{n-j-1}{t_a-j-b}}$. Then $|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\rangle$ remains a multiple of $|\upsilon_{a,b}\rangle$ but may no longer be equal to $|\upsilon_{a,b}\rangle$.

$\kappa_0, \dots, \kappa_{j-1}$ should be such that the state is orthogonal to $T_{j-1,a,b}$ and, in particular, orthogonal to the states $|\psi^{a,b}_{i_1,\dots,i_\ell}\rangle$ for all $\ell \in \{0, \dots, j-1\}$. By writing out $\langle\upsilon_{a,b}|\psi^{a,b}_{i_1,\dots,i_\ell}\rangle = 0$:

$$\sum_{m=\ell}^{j} \kappa_m \binom{n-j-1}{t_a-m-b}\binom{j-\ell}{m-\ell} = 0. \tag{20}$$

To show that, we first note that $|\psi^{a,b}_{i_1,\dots,i_\ell}\rangle$ is a uniform superposition of all $|x\rangle$ with $|x| = t_a$, $x_1 = b$, $x_{i_1} = \dots = x_{i_\ell} = 1$. If we want to choose $x$ subject to those constraints and also satisfying $|\{i_1, \dots, i_j\} \cap \{i : x_i = 1\}| = m$, then we have to set $x_i = 1$ for $m - \ell$ different $i \in \{i_{\ell+1}, \dots, i_j\}$ and for $t_a - m - b$ different $i \notin \{1, i_1, \dots, i_j\}$. This can be done in $\binom{j-\ell}{m-\ell}$ and $\binom{n-j-1}{t_a-m-b}$ different ways, respectively.

By solving the system of equations (20), starting from $\ell = j-1$ and going down to $\ell = 0$, we get that the only solution is

$$\kappa_m = (-1)^{j-m}\frac{\binom{n-j-1}{t_a-j-b}}{\binom{n-j-1}{t_a-m-b}}\kappa_j. \tag{21}$$

Let $|\upsilon'_{a,b}\rangle = \frac{|\upsilon_{a,b}\rangle}{\|\upsilon_{a,b}\|}$ be the normalized version of $|\upsilon_{a,b}\rangle$. Then

$$\begin{aligned}|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\rangle &= \langle\upsilon'_{a,b}|\psi^{a,b}_{i_1,\dots,i_j}\rangle|\upsilon'_{a,b}\rangle, \\ \|\tilde{\psi}^{a,b}_{i_1,\dots,i_j}\| &= \langle\upsilon'_{a,b}|\psi^{a,b}_{i_1,\dots,i_j}\rangle = \frac{\langle\upsilon_{a,b}|\psi^{a,b}_{i_1,\dots,i_j}\rangle}{\|\upsilon_{a,b}\|}.\end{aligned} \tag{22}$$

We have

$$\langle\upsilon_{a,b}|\psi^{a,b}_{i_1,\dots,i_j}\rangle = 1,$$

because $|\psi^{a,b}_{i_1,\dots,i_j}\rangle$ consists of $\binom{n-j-1}{t_a-j-b}$ basis states $|x\rangle$, $x_1 = b$, $x_{i_1} = \cdots = x_{i_j} = 1$, each having amplitude $1/\sqrt{\binom{n-j-1}{t_a-j-b}}$ in both $|\upsilon_{a,b}\rangle$ and $|\psi^{a,b}_{i_1,\dots,i_j}\rangle$. Furthermore,

$$\begin{aligned}
\|\upsilon_{a,b}\|^2 &= \sum_{m=0}^{j} \binom{j}{m}\binom{n-j-1}{t_a-m-b}\kappa_m^2 \\
&= \sum_{m=0}^{j} \binom{j}{m}\frac{\binom{n-j-1}{t_a-j-b}^2}{\binom{n-j-1}{t_a-m-b}}\kappa_j^2 \\
&= \sum_{m=0}^{j} \binom{j}{m}\frac{\binom{n-j-1}{t_a-j-b}}{\binom{n-j-1}{t_a-m-b}} \\
&= \sum_{m=0}^{j} \binom{j}{m}\frac{(t_a-m-b)!(n-t_a+m-j-1+b)!}{(t_a-j-b)!(n-t_a-1+b)!} \\
&= \sum_{m=0}^{j} \binom{j}{m}\frac{(t_a-m-b)^{\underline{j-m}}}{(n-t_a-1+b)^{\underline{j-m}}}. \qquad (23)
\end{aligned}$$

Here the first equality follows because there are $\binom{j}{m}\binom{n-j-1}{t_a-m-b}$ vectors $x$ such that $|x| = t_a$, $x_1 = b$, $x_i = 1$ for $m$ different $i \in \{i_1, \dots, i_j\}$ and $t_a - m$ different $i \notin \{1, i_1, \dots, i_j\}$, the second equality follows from equation (21) and the third equality follows from our choice $\kappa_j = 1/\sqrt{\binom{n-j-1}{t_a-j-b}}$.

From equations (22) and (23), $\|\tilde\psi^{a,b}_{i_1,\dots,i_j}\| = \frac{1}{\sqrt{A_{a,b}}}$ where $A_{a,b} = \sum_{m=0}^{\infty} C_{a,b}(m)$ and

$$C_{a,b}(m) = \binom{j}{m}\frac{(t_a-m-b)^{\underline{j-m}}}{(n-t_a-1+b)^{\underline{j-m}}}.$$

The terms with $m > j$ are zero because $\binom{j}{m} = 0$ for $m > j$.

We compute the combinatorial sum $A_{a,b}$ using hypergeometric series [16, Section 5.5]. Since

$$\frac{C_{a,b}(m+1)}{C_{a,b}(m)} = \frac{(m-j)(m+n-t_a-j+b)}{(m+1)(m-t_a+b)}$$

is a rational function of $m$, $A_{a,b}$ is a hypergeometric series and its value is

$$A_{a,b} = \sum_{m=0}^{\infty} C_{a,b}(m) = C_{a,b}(0)\cdot F\Big(\begin{matrix}-j,\ n-t_a-j+b\\ -t_a+b\end{matrix}\Big|1\Big).$$

We apply Vandermonde's convolution $F(\begin{smallmatrix}-j,\ x\\ y\end{smallmatrix}|1) = (x-y)^{\underline{j}}/(-y)^{\underline{j}}$ [16, Equation 5.93 on page 212], which holds for every integer $j \ge 0$, and obtain

$$A_{a,b} = \frac{(t_a-b)^{\underline{j}}}{(n-t_a-1+b)^{\underline{j}}}\cdot\frac{(n-j)^{\underline{j}}}{(t_a-b)^{\underline{j}}} = \frac{(n-j)^{\underline{j}}}{(n-t_a-1+b)^{\underline{j}}}.$$

This proves the first part of the claim, that $\|\tilde\psi^{a,b}_{i_1,\dots,i_j}\| = \sqrt{(n-t_a-1+b)^{\underline{j}}/(n-j)^{\underline{j}}}$.

The second part of the claim follows because

$$\begin{aligned}
\frac{\|\tilde\psi^{a,0}_{i_1,\dots,i_j}\|}{\|\tilde\psi^{a,1}_{i_1,\dots,i_j}\|} &= \sqrt{\frac{(n-t_a-1)^{\underline{j}}}{(n-t_a)^{\underline{j}}}} = \sqrt{\frac{n-t_a-j}{n-t_a}} \\
&= \sqrt{1-\frac{j}{n-t_a}} \ge \sqrt{1-\frac{n/4}{n/2}} = \frac{1}{\sqrt{2}},
\end{aligned}$$

because $j \le t_a/2$, and $t_a \le n/2$.

For the third part,

$$\begin{aligned}
\frac{A_{1,0}A_{0,1}}{A_{0,0}A_{1,1}} &= \frac{((n-t)^{\underline{j}})^2}{(n-t+1)^{\underline{j}}(n-t-1)^{\underline{j}}} \\
&= \frac{(n-t)(n-t-j+1)}{(n-t+1)(n-t-j)} \\
&= 1+\frac{j}{(n-t+1)(n-t-j)},
\end{aligned}$$

which is $1+\Theta(j/n^2) = 1+O(1/t)$ for $t \le n/2$ and $j \le t/2$. The expression in the third part of the claim is the square root of this value, hence it is $1+O(1/t)$. □

Claim 19. *If $j < t/2$, then $\beta_a \le \sqrt{\frac{2t}{n}}$.*

Proof. Define $t_a = t-1+a$. By Claim 18, $\|\tilde\psi^{a,0}_{i_1,\dots,i_j}\| \ge \frac{1}{\sqrt{2}}\|\tilde\psi^{a,1}_{i_1,\dots,i_j}\|$. That implies

$$\begin{aligned}
\alpha'_a &= \frac{\sqrt{n-t_a}}{\sqrt{n-j}}\|\tilde\psi^{a,0}_{i_1,\dots,i_j}\| \\
&\ge \frac{1}{\sqrt{2}}\frac{\sqrt{n-t_a}}{\sqrt{t_a-j}}\frac{\sqrt{t_a-j}}{\sqrt{n-j}}\|\tilde\psi^{a,1}_{i_1,\dots,i_j}\| = \frac{\sqrt{n-t_a}}{\sqrt{2(t_a-j)}}\beta'_a
\end{aligned}$$

and hence

$$\sqrt{(\alpha'_a)^2+(\beta'_a)^2} \ge \beta'_a\sqrt{\frac{n-t_a}{2(t_a-j)}+1} = \beta'_a\frac{\sqrt{n+t_a-2j}}{\sqrt{2(t_a-j)}}.$$

Then, using $j \le \frac{t_a}{2}$,

$$\beta_a = \frac{\beta'_a}{\sqrt{(\alpha'_a)^2+(\beta'_a)^2}} \le \frac{\sqrt{2(t_a-j)}}{\sqrt{n+t_a-2j}} \le \sqrt{\frac{2t}{n}}. \quad \square$$

Claim 20. *If $j < t/2$, then $|\alpha_0\beta_1 - \alpha_1\beta_0| = O\left(\frac{1}{\sqrt{tn}}\right)$.*

Proof. We first estimate

$$\frac{\alpha_0\beta_1}{\alpha_1\beta_0} = \frac{\alpha'_0\beta'_1}{\alpha'_1\beta'_0} = \frac{\sqrt{(n-t+1)(t-j)}}{\sqrt{(n-t)(t-1-j)}}\cdot\frac{\|\tilde\psi^{0,0}_{i_1,\dots,i_j}\|\|\tilde\psi^{1,1}_{i_1,\dots,i_j}\|}{\|\tilde\psi^{1,0}_{i_1,\dots,i_j}\|\|\tilde\psi^{0,1}_{i_1,\dots,i_j}\|}.$$

By Claim 18, we have

$$\frac{\alpha'_0\beta'_1}{\alpha'_1\beta'_0} = \left(1+O\left(\frac{1}{t}\right)\right)\frac{\sqrt{(n-t+1)(t-j)}}{\sqrt{(n-t)(t-1-j)}}.$$

Since $\frac{\sqrt{t-j}}{\sqrt{t-1-j}} = \sqrt{1+\frac{1}{t-1-j}} = 1+O(\frac{1}{t-1-j}) = 1+O(\frac{1}{t})$ and, similarly, $\frac{\sqrt{n-t+1}}{\sqrt{n-t}} = 1+O(\frac{1}{n-t}) = 1+O(\frac{1}{t})$, we have shown that $\frac{\alpha_0\beta_1}{\alpha_1\beta_0}$ is of order $1+O(\frac{1}{t})$. We thus have

$$|\alpha_0\beta_1-\beta_0\alpha_1| = O\left(\frac{1}{t}\right)|\beta_0\alpha_1| = O\left(\frac{1}{t}\cdot\sqrt{\frac{t}{n}}\right) = O\left(\frac{1}{\sqrt{tn}}\right),$$

thanks to Claim 19 and the fact that $|\alpha_1| \le 1$. □

We pick an orthonormal basis for $\mathcal{H}_4$ that has $|\phi_1\rangle$ and $|\phi_2\rangle$ as its first two vectors. Let $|\phi_3\rangle$ and $|\phi_4\rangle$ be the other two basis vectors. We define

$$|\chi_i\rangle = |\phi_i\rangle\otimes|\psi_2\rangle\otimes\cdots\otimes|\psi_k\rangle. \qquad (24)$$

By Claim 17, $|\chi_1\rangle$ belongs to $S_{j,+}\otimes R_{j_2}\otimes\cdots\otimes R_{j_k}$ which is contained in $\mathcal{R}_{\min(j,t/2)+j_2+\cdots+j_k}$. Similarly, $|\chi_2\rangle$ belongs to

$\mathcal{R}_{\min(j+1,t/2)+j_2+\cdots+j_k}$ and $|\chi_3\rangle$, $|\chi_4\rangle$ belong to $\mathcal{R}_{t/2+j_2+\cdots+j_k}$. If $j < t/2$, this means that

$$P(\rho_{d,1}) = q^{j_2+\cdots+j_k} \cdot \Big(q^j \langle\chi_1|\rho_{d,1}|\chi_1\rangle + q^{j+1}\langle\chi_2|\rho_{d,1}|\chi_2\rangle + q^{\frac{t}{2}}\langle\chi_3|\rho_{d,1}|\chi_3\rangle + q^{\frac{t}{2}}\langle\chi_4|\rho_{d,1}|\chi_4\rangle\Big) \quad (25)$$

If $j \geq t/2$, then $|\chi_1\rangle$, $|\chi_2\rangle$, $|\chi_3\rangle$, $|\chi_4\rangle$ are all in $\mathcal{R}_{t/2+j_2+\cdots+j_k}$. This means that $P(\rho_{d,1}) = q^{t/2+j_2+\cdots+j_k}$ and it remains unchanged by a query.

We define $\gamma_\ell = \langle\chi_\ell|\rho_{d,1}|\chi_\ell\rangle$. Since the support of $\rho_{d,1}$ is contained in the subspace spanned by $|\chi_\ell\rangle$, we have $\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 = \operatorname{Tr}\rho_{d,1} = 1$. This means that equation (25) can be rewritten as

$$\begin{aligned} P(\rho_{d,1}) &= q^{j+j_2+\cdots+j_k}\gamma_1 + q^{j+j_2+\cdots+j_k+1}\gamma_2 + \\ &\quad + q^{t/2+j_2+\cdots+j_k}(\gamma_3+\gamma_4) \\ &= q^{t/2+j_2+\cdots+j_k} + q^{j_2+\cdots+j_k}(q^{j+1}-q^{t/2})(\gamma_1+\gamma_2) + \\ &\quad + q^{j_2+\cdots+j_k}(q^j - q^{j+1})\gamma_1 \end{aligned} \quad (26)$$

$P(\rho'_{d,1})$ can be also expressed in a similar way, with $\gamma'_j = \langle\chi_j|\rho'_{d,1}|\chi_j\rangle$ instead of $\gamma_j$. By combining equations (26) for $P(\rho_{d,1})$ and $P(\rho'_{d,1})$, we get

$$\begin{aligned} P(\rho'_{d,1}) - P(\rho_{d,1}) &= q^{j+j_2+\cdots+j_k}(q^{t/2-j}-q) \\ &\quad \cdot (\gamma_1+\gamma_2-\gamma'_1-\gamma'_2) + q^{j+j_2+\cdots+j_k}(q-1)(\gamma_1-\gamma'_1). \end{aligned}$$

Therefore, it suffices to bound $|\gamma_1+\gamma_2-\gamma'_1-\gamma'_2|$ and $|\gamma_1-\gamma'_1|$. W.l.o.g. we can assume that $\rho_{d,1}$ is a pure state $|\varphi\rangle\langle\varphi|$. Let

$$|\varphi\rangle = (a|\psi_{00}\rangle + b|\psi_{01}\rangle + c|\psi_{10}\rangle + d|\psi_{11}\rangle) \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_k\rangle.$$

Then the state after a query is

$$|\varphi'\rangle = (a|\psi_{00}\rangle - b|\psi_{01}\rangle + c|\psi_{10}\rangle - d|\psi_{11}\rangle) \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_k\rangle$$

and we have to bound

$$\gamma_\ell - \gamma'_\ell = |\langle\chi_\ell|\varphi\rangle|^2 - |\langle\chi_\ell|\varphi'\rangle|^2$$

for $\ell \in \{1,2\}$. For $\ell = 1$, we have

$$\langle\chi_1|\varphi\rangle = a\alpha_0 + b\beta_0 + c\alpha_1 + d\beta_1.$$

The expression for $\varphi'$ is similar, with minus signs in front of $b\beta_0$ and $d\beta_1$. Therefore,

$$\begin{aligned} &\big||\langle\chi_1|\varphi\rangle|^2 - |\langle\chi_1|\varphi'\rangle|^2\big| \\ &\quad \leq 4|a||b|\alpha_0\beta_0 + 4|c||d|\alpha_1\beta_1 + 4|a||d|\alpha_0\beta_1 + 4|b||c|\alpha_1\beta_0. \end{aligned} \quad (27)$$

Since $|a|$, $|b|$, $|c|$, $|d|$ are all at most $\|\varphi\| = 1$ and $\alpha_0$, $\alpha_1$ are less than 1, equation (27) is at most $8\beta_0 + 8\beta_1$. By Claim 19, we have

$$|\gamma_1 - \gamma'_1| \leq 8\beta_0 + 8\beta_1 \leq 16\sqrt{\frac{2t}{n}}.$$

We also have

$$\begin{aligned} &\big|\gamma_1+\gamma_2-\gamma'_1-\gamma'_2\big| \\ &\quad = \big||\langle\chi_1|\varphi\rangle|^2 + |\langle\chi_2|\varphi\rangle|^2 - |\langle\chi_1|\varphi'\rangle|^2 - |\langle\chi_2|\varphi'\rangle|^2\big| \\ &\quad \leq 4|a||d||\alpha_0\beta_1 - \alpha_1\beta_0| + 4|b||c||\alpha_1\beta_0 - \alpha_0\beta_1| \\ &\quad \leq 8|\alpha_0\beta_1 - \alpha_1\beta_0| \leq \frac{8C}{\sqrt{tn}} \end{aligned}$$

where $C$ is the big-O constant from Claim 20. By taking into account that $P(\rho_{d,1}) \geq q^{j+j_2+\cdots+j_k}$,

$$\begin{aligned} &P(|\varphi'\rangle\langle\varphi'|) - P(|\varphi\rangle\langle\varphi|) \\ &\quad \leq \left((q^{t/2-j}-q)\frac{8C}{\sqrt{tn}} + (q-1)\frac{16\sqrt{2t}}{\sqrt{n}}\right)P(|\varphi\rangle\langle\varphi|) \\ &\quad \leq \left((q^{t/2}-1)\frac{8C}{\sqrt{tn}} + (q-1)\frac{16\sqrt{2t}}{\sqrt{n}}\right)P(|\varphi\rangle\langle\varphi|). \end{aligned} \quad (28)$$

This proves Lemma 4 for the case when the support of $\rho_{d,1}$ is contained in $\mathcal{H}_4$. (If $\rho_{d,1}$ is a mixed state, we just express it as a mixture of pure states $|\varphi\rangle$. The bound for $\rho_{d,1}$ follows by summing equations (28) for every $|\varphi\rangle$.)

For the general case, we divide the entire state space $\mathcal{H}_I$ into 4-dimensional subspaces. To do that, we first subdivide $\mathcal{H}_I$ into subspaces

$$(S_{j,0,0} \oplus S_{j,0,1} \oplus S_{j,1,0} \oplus S_{j,1,1}) \otimes R_{j_2} \otimes \cdots \otimes R_{j_k}. \quad (29)$$

Let states $|\psi^{0,0}_{1,i}\rangle$, $i \in [\dim S_{j,0,0}]$ form a basis for $S_{j,0,0}$ and let $|\psi^{a,b}_{1,i}\rangle = U'_{ab}|\psi^{0,0}_{1,i}\rangle$ for $(a,b) \in \{(0,1),(1,0),(1,1)\}$, where the $U'_{ab}$ are the unitaries from Claim 16. Then the $|\psi^{a,b}_{1,i}\rangle$ form a basis for $S_{j,a,b}$.

Let $|\psi_{l,i}\rangle$, $i \in [\dim R_{j_l}]$, form a basis for $R_{j_l}$, $l \in \{2,\ldots,k\}$. We subdivide (29) into 4-dimensional subspaces $H_{i_1,\ldots,i_k}$ spanned by

$$|\psi^{a,b}_{1,i_1}\rangle \otimes |\psi_{2,i_2}\rangle \otimes \cdots \otimes |\psi_{k,i_k}\rangle,$$

where $a, b$ range over $\{0,1\}$. Let $\mathcal{H}_{all}$ be the collection of all $H_{i_1,\ldots,i_k}$ obtained by subdividing all subspaces (29). We claim that

$$P(\rho) = \sum_{H\in\mathcal{H}_{all}} P(\Pi_H\rho). \quad (30)$$

Equation (30) together with equation (28) implies Lemma 4. Since $P(\rho)$ is defined as a weighted sum of traces $\operatorname{Tr}\Pi_{\mathcal{R}_m}\rho$, we can prove equation (30) by showing

$$\operatorname{Tr}\Pi_{\mathcal{R}_m}\rho_{d,1} = \sum_{H\in\mathcal{H}_{all}} \operatorname{Tr}\Pi_{\mathcal{R}_m}\Pi_H\rho_{d,1}. \quad (31)$$

To prove (31), we define a basis for $\mathcal{H}_I$ by first decomposing $\mathcal{H}_I$ into subspaces $H \in \mathcal{H}_{all}$, and then for each subspace, taking the basis consisting of $|\chi_1\rangle$, $|\chi_2\rangle$, $|\chi_3\rangle$ and $|\chi_4\rangle$ defined by equation (24). By Claim 17, each of the basis states belongs to one of the subspaces $\mathcal{R}_m$. This means that each $\mathcal{R}_m$ is spanned by some subset of this basis.

The left hand side of (31) is equal to the sum of squared projections of $\rho_{d,1}$ to basis states $|\chi_j\rangle$ that belong to $\mathcal{R}_m$. Each of the terms $\operatorname{Tr}\Pi_{\mathcal{R}_m}\Pi_H\rho_{d,1}$ on the right hand side is equal to the sum of squared projections to basis states $|\chi_j\rangle$ that belong to $\mathcal{R}_m \cap H$. Summing over all $H$ gives the sum of squared projections of $\rho_{d,1}$ to all $|\chi_j\rangle$ that belong to $\mathcal{R}_m$. Therefore, the two sides of (31) are equal.