



Fourier Spectrum of Noisy Quantum Algorithms

Uma Girish*

Abstract

Quantum computing promises exponential speedups for certain problems, yet fully universal quantum computers remain out of reach and near-term devices are inherently noisy. Motivated by this, we study noisy quantum algorithms and the landscape between BQP and BPP. We build on a powerful technique to differentiate quantum and classical algorithms called the level- ℓ Fourier growth (the sum of absolute values of Fourier coefficients of sets of size ℓ) and show that it can also be used to differentiate quantum algorithms based on the types of resources used. We show that noise acting on a quantum algorithm dampens its Fourier growth in ways intricately linked to the type of noise.

Concretely, we study noisy models of quantum computation where highly mixed states are prevalent, namely: DQC_k algorithms, where k qubits are clean and the rest are maximally mixed, and $\frac{1}{2}\text{BQP}$ algorithms, where the initial state is maximally mixed, but the algorithm is given knowledge of the initial state at the end of the computation. We establish upper bounds on the Fourier growth of DQC_k , $\frac{1}{2}\text{BQP}$ and BQP algorithms and leverage the differences between these bounds to derive oracle separations between these models. In particular, we show that 2-FORRELATION and 3-FORRELATION require $N^{\Omega(1)}$ queries in the DQC_1 and $\frac{1}{2}\text{BQP}$ models respectively. Our results are proved using a new matrix decomposition lemma that might be of independent interest.

Contents

1	Introduction	3
1.1	The Space Below BQP	3
1.2	Fourier Growth	5
1.3	Our Results	7
1.4	Applications	9
1.5	Technical Highlight: Matrix Decomposition Lemma	10
1.6	Proof Sketch	12
1.7	Outlook & Future Directions	15
1.8	Organization.	16
2	Preliminaries & Notation	16
2.1	Fourier Growth	17
2.2	Quantum Query Complexity	18
2.3	DQC_k algorithms	18
2.4	$\frac{1}{2}\text{BQP}$ algorithms	19

*Columbia University. Email: ug2150@columbia.edu

3	Main Technical Tool: Matrix Decomposition Lemma	20
3.1	Matrix Decomposition Lemma for DQC_1	25
3.2	Matrix Decomposition Lemma for $\frac{1}{2}\text{BQP}$	25
4	Fourier Growth of DQC_k: Level-Two	26
4.1	Tightness of our Bounds for DQC_1	28
5	Fourier Growth of $\frac{1}{2}\text{BQP}$: Level-Three	29
6	Higher Level Fourier Growth from Lower Levels	34
6.1	Improved Bootstrapping for $\frac{1}{2}\text{BQP}$ algorithms	36
7	Acknowledgements	38
A	Appendix	40
A.1	Proof of Lemma 3.4	40
A.2	Proof of Lemma 3.6	42
A.3	Quantum Algorithms with Classical Pre-Processing.	54
A.4	Simulating DQC_k algorithms by DQC_{k-t} algorithms.	57
A.5	Acceptance Probability of Quantum Algorithms	57
A.6	Fourier Coefficients of Quantum Algorithms	58

1 Introduction

Quantum computing promises to solve certain problems exponentially faster than classical computers, as evidenced by numerous query complexity separations or oracle separations [DJ92, BV97, Sim97, Aar10, AA15]. Yet, we haven't been able to harness this, as we are far from being able to build fully universal quantum computers. While BQP algorithms generally assume noiseless computation, noise is arguably the most significant issue faced by near-term quantum computers and all current quantum devices are inherently noisy. To better understand what quantum resources are truly responsible for quantum advantage, researchers have proposed numerous intermediate models of quantum computing like IQP, DQC₁, NISQ and Boson Sampling [SB08, KL98, CCHL23, AA11, ABKM17]. These models isolate specific quantum features – such as having a few clean qubits or limited adaptivity – and allow us to probe the quantum landscape below BQP. Although these models likely do not capture the full power of quantum computing, their precise relationship to BQP and to each other remains poorly understood. This raises a natural question:

What does the landscape of quantum computation below BQP look like?

In our work, we study this question from a Fourier analytic perspective. In particular, we study the level- ℓ Fourier growth of the acceptance probability of algorithms (Definition 1.2). This is a measure of how well-spread the Fourier coefficients are. In our work, we show that Fourier growth is not just a tool for distinguishing quantum and classical models; it is a fine-grained tool capable of differentiating quantum models based on the kinds of quantum resources they utilize. We focus on noisy quantum algorithms and demonstrate that noise dampens the Fourier growth in ways that are intricately linked to the type of noise present.

In particular, we study noisy models like DQC _{k} , where k qubits are clean and the rest are maximally mixed [KL98, MFF14], and $\frac{1}{2}$ BQP, where the initial state is maximally mixed, i.e., a uniformly random computational basis state, but the algorithm is given knowledge of this initial state at the end of the computation [ABKM17, JM24]. We prove Fourier growth bounds on the acceptance probability of DQC _{k} , $\frac{1}{2}$ BQP and BQP algorithms (Theorems 1.5 to 1.7) and use the differences in these bounds to derive oracle separations between these models. In particular, we show that 2-FORRELATION and 3-FORRELATION, which can be solved with two queries in the $\frac{1}{2}$ BQP and BQP models respectively, require $N^{\Omega(1)}$ queries in the DQC₁ and $\frac{1}{2}$ BQP models respectively (Corollaries 1.9 and 1.10), resolving two conjectures from [JM24] and establishing the first oracle separation between $\frac{1}{2}$ BQP and DQC _{k} , as well as a new oracle separation between BQP and $\frac{1}{2}$ BQP.

We believe that the noise-induced dampening of Fourier growth is a more general phenomenon, and that the techniques developed here could shed light on other noisy models such as NISQ. Our results are proved using a new matrix decomposition lemma that encodes information about indices in a matrix product that might be of independent interest.

1.1 The Space Below BQP

The landscape of computational models between BQP and BPP is vast and intricate. There are numerous intermediate models of quantum computation like IQP, DQC₁, NISQ and Boson Sampling [SB08, KL98, CCHL23, AA11] with constraints on the quantum resources. The study of such intermediate models serves two key purposes: (1) to systematically delineate the boundary between classical and quantum algorithms and pinpoint the minimal resources for quantum speedups, and (2) to model the physical constraints of near-term quantum devices and reason about them.

One important issue that affects near-term quantum computers is noise. Unlike classical systems, quantum computers are highly susceptible to various types of errors due to decoherence,

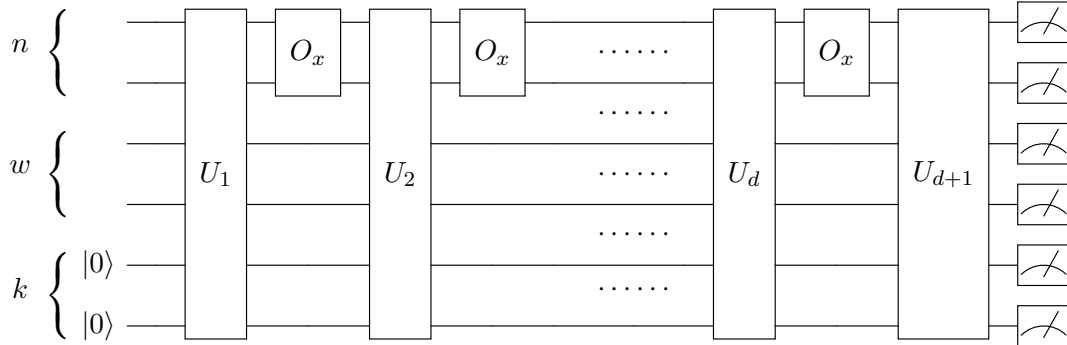


Figure 1: A d -query DQC_k algorithm. The initial state on the first $n+w$ qubits is maximally mixed.

imperfect gates, and environmental interactions. This prompts a natural question – how much noise can quantum algorithms tolerate? How does noise change the quantum computational complexity landscape? This is challenging to answer in general, as there are many different kinds of noise that affect quantum algorithms. One way to simplify this challenge is to consider models that have an extreme amount of noise. In this endeavor, researchers have proposed highly noisy models of quantum computation like DQC_k and $\frac{1}{2}\text{BQP}$ [KL98, MFF14, JM24] where all the noise is pushed onto the initial state – the qubits start maximally or nearly maximally mixed, while the gates are noiseless. These models provide a framework for understanding the minimal number of clean qubits required to achieve quantum speedups. We describe these models below.

DQC_k Drawing inspiration from the NMR approach to quantum computing where mixed states are ubiquitous, Knill and Laflamme [KL98] introduced the one-clean qubit or DQC_1 model as an idealized version of a noisy quantum computer. In this model, one qubit is clean (noiseless) and the rest are maximally noisy, and the algorithm can apply (noiseless) unitary gates on these qubits and measure at the end. This model was later generalized to DQC_k to allow k clean qubits [MFF14, FKM⁺15]. This model does not seem to be universal for quantum computing since all qubits except a few are maximally noisy and many oracle problems like Simon’s problem and order finding are not believed to be solvable in this model. Despite this, DQC_1 can solve problems that are believed to be classically hard, like estimating the trace and Pauli coefficients of a unitary matrix described by a quantum circuit [KL98, DFC05], Jones polynomials [SJ08], partition functions [CSS21]. Under complexity theoretic assumptions, this model is not classically simulable [MFF14, FKM⁺18, Mor17]. There are exponential oracle separations between DQC_1 and BPP [She10]. The communication version of the one clean qubit model provides exponential speedups over classical randomized communication [AGL23]. The fact that quantum speedups persist even under such extreme noise makes DQC_1 a particularly intriguing model for further study – it challenges our understanding of what minimal quantum resources are required for speedups.

$\frac{1}{2}\text{BQP}$. The $\frac{1}{2}\text{BQP}$ model was originally defined by [ABKM17] to capture the power of permutational computations on special input states. This model was revisited by [JM24] in the context of delineating the boundary between BQP and DQC_1 . In this model, the initial state is maximally mixed, i.e., a uniformly random computational basis state, but the algorithm learns this state at the end of the computation and decides whether to accept or reject. One can equivalently define this model as a quantum algorithm acting on one half of a maximally entangled EPR state and in the end, we measure both halves and do classical postprocessing on the measurement outcomes. This model is not believed to be universal for quantum computing as it allows a significant amount of

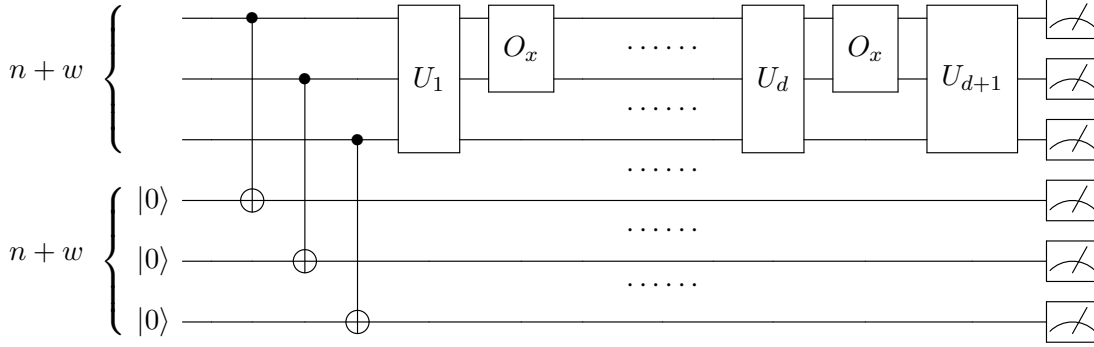


Figure 2: A d -query $\frac{1}{2}$ BQP algorithm. The initial state on the first $n + w$ qubits can be thought of as maximally mixed, or as the pure state $2^{-(n+w)/2} \sum_{I \in \{0,1\}^{n+w}} |I\rangle$; the resulting circuits are equivalent.

noise, yet, this model encapsulates many known quantum speedups. It can solve the factoring problem and numerous oracle-based problems including Simon’s problem, Deutsch-Jozsa, order finding, and the Forrelation problem and can simulate DQC_k for any small k as well as IQP [JM24]. It appears to be the weakest quantum model that is unlikely to be universal and yet captures most known BQP speedups despite operating on maximally mixed states.

A powerful and natural framework to study the differences between $\text{DQC}_k, \frac{1}{2}$ BQP and BQP is query complexity. In this setting, there is a boolean function $f : \{-1, 1\}^N \rightarrow \{-1, 1\}$ and the goal is to compute $f(x)$ for $x \in \{-1, 1\}^N$ while minimizing the number of queries to the oracle O_x . This model strips away implementation details and captures the essence of what makes different computational models powerful. The aforementioned quantum models can be formalized using this framework and are depicted in Figures 1 and 2. (See Definitions 2.9 and 2.11 for more details.) Query complexity has long been one of the most fruitful arenas for understanding the differences between quantum and classical computation and gives us strong evidence for quantum advantage, including provable exponential oracle separations between BQP and BPP. Over the years, the field has also developed an impressive arsenal of lower-bound techniques for both quantum and classical algorithms. While these techniques are powerful for distinguishing quantum from classical, they are not designed to distinguish between quantum algorithms. Indeed, many of these methods – including the polynomial method – apply uniformly to all bounded low-degree polynomials and cannot capture the subtle differences between $\text{DQC}_1, \frac{1}{2}$ BQP and BQP. This motivates the search for more fine-grained analytic techniques.

The central contribution of this paper is to show that a Fourier analytic concept known as Fourier growth provides exactly such a tool. While Fourier growth was historically used to distinguish between quantum and classical algorithms, we demonstrate that it can also serve as a lens to separate quantum models from each other. We show that noise dampens the Fourier growth of quantum algorithms in ways that are intricately tied to the noise patterns. We now introduce Fourier growth, provide its historical context and describe its importance.

1.2 Fourier Growth

Fourier growth has emerged as a central concept that allows us to distinguish quantum and classical algorithms. To formally define Fourier growth, recall that every boolean function $f :$

$\{-1, 1\}^N \rightarrow \mathbb{R}$ can be uniquely represented as a multi-linear polynomial

$$f(x) = \sum_{S \subseteq [N]} \widehat{f}(S) \cdot \prod_{i \in S} x_i$$

where $\widehat{f}(S)$ are called the Fourier coefficients of f .

Definition 1.1 (Signed Fourier Growth). *For level $\ell \in \mathbb{N}$, and signs $\alpha_S \in [-1, 1]$ for $S \subseteq [N]$ with $|S| = \ell$, define the α -signed level- ℓ Fourier growth of f , denoted by $L_{1,\ell}^\alpha(f)$ as*

$$L_{1,\ell}^\alpha(f) := \sum_{\substack{S \subseteq [N] \\ |S| = \ell}} \alpha_S \cdot \widehat{f}(S),$$

Definition 1.2 (Fourier Growth). *For level $\ell \in \mathbb{N}$, the level- ℓ Fourier growth of f , denoted by $L_{1,\ell}(f)$, is the ℓ_1 -norm of the level- ℓ Fourier coefficients of f ,*

$$L_{1,\ell}(f) := \sum_{\substack{S \subseteq [N] \\ |S| = \ell}} |\widehat{f}(S)| = \max_{\alpha \in [-1, 1]^{\binom{N}{\ell}}} L_{1,\ell}^\alpha(f).$$

Fourier growth bounds have been extensively studied and established for various classical models¹, including small-width DNFs/CNFs [Man95], AC^0 circuits [Tal17], low-depth decision trees [Tal20, SSW23], low-degree $\text{GF}(2)$ polynomials [CHLT19], low-depth parity decision trees [GTW21], low-degree bounded polynomials [IRR⁺21], and more. Upper bounds on the Fourier growth, even for the first few levels, give rise to quantum versus classical separations. Intuitively, while both quantum and classical algorithms of small query complexity can be represented by low-degree polynomials, the polynomials associated with quantum algorithms are a lot “denser” compared to their classical analogues, and this density is captured by Fourier growth. In particular, it was shown by [Tal20, SSW23] that for d -query classical algorithms, $L_{1,\ell}(f)$ is at most $\tilde{O}_\ell(d^{\ell/2})$; on the other hand, for d -query quantum algorithms, $L_{1,\ell}(f)$ is at most $O_\ell(d^\ell) \cdot N^{(\ell-1)/2}$ [IRR⁺21] and this can be tight for certain algorithms. A key problem that exploits this difference in the Fourier growth is the Forrelation problem. This was originally introduced by Aaronson and Ambainis [Aar10, AA15] to show an oracle separation between BQP and BPP and was subsequently used by Raz and Tal [RT22] in their breakthrough oracle separation of BQP and PH. Building on this, [BS21] generalized this to the k -Forrelation problem and used it to show optimal separations between BQP and BPP. We describe this problem below.

Definition 1.3 (k -Forrelation function). *Let $N = 2^n$. For $x^{(1)}, \dots, x^{(k)} \in \{-1, 1\}^N$, define*

$$\text{forr}^{(k)}(x^{(1)}, \dots, x^{(k)}) := \langle v | H_N \cdot O_{x^{(1)}} \cdot H_N \cdot O_{x^{(2)}} \cdots \cdots H_N \cdot O_{x^{(k)}} \cdot H_N | v \rangle$$

where H_N is the $N \times N$ unitary Hadamard matrix as in Definition 2.1 and $|v\rangle = |0 \dots 0\rangle$.

Definition 1.4 (k -FORRELATION problem with parameter $\varepsilon = \Theta(1/\log^k N)$). *Given input $x \in \{-1, 1\}^{kN}$, return -1 if $\text{forr}^{(k)}(x) \geq 2\varepsilon$ and 1 if $\text{forr}^{(k)}(x) \leq \varepsilon$.*

¹By Fourier growth of a model, we refer to the Fourier growth of the acceptance probability of an algorithm in this model.

Fourier Growth of d -query algorithms	$\ell = 1$	$\ell = 2$	$\ell = 3$
BQP [IRR ⁺ 21], Theorem 1.7	d	$\tilde{O}(d\sqrt{N})$	$\tilde{O}(dN)$
$\frac{1}{2}$ BQP Theorem 1.6	$O(d)$	$\tilde{O}(d\sqrt{N})$	$O(d^7\sqrt{N})$
DQC ₁ Theorem 1.5	$O(d)$	$O(d^3)$	$\tilde{O}(d^3\sqrt{N})$
BPP [Tal20, SSW23]	$O(\sqrt{d})$	$O(d\sqrt{\log N})$	$O(\sqrt{d^3} \log N)$

Table 1: Upper Bounds on the Fourier growth of the acceptance probability of various d -query algorithms.

Quantum algorithms in the BQP model can solve k -FORRELATION using $\lceil k/2 \rceil$ quantum queries. Furthermore, the results of [RT22, CHLT19, RT22, BS21] imply that any family of algorithms solving k -FORRELATION must have large Fourier growth at levels $k, 2k, \dots, k(k-1)$ (see Theorem 2.4 and Theorem 2.6 for a precise statement). These results effectively reduce the task of proving lower bounds for the Forrelation problem to the task of establishing Fourier growth bounds. In particular, 2-FORRELATION involves level-2 bounds and 3-FORRELATION involves level-3 and level-6 bounds. Since classical algorithms have small Fourier growth at all levels, it follows from the aforementioned works that they cannot solve the Forrelation problem.

1.3 Our Results

In our work, we go beyond the idea of using Fourier growth to distinguish between quantum and classical algorithms. We show that although Fourier growth can be large for quantum algorithms, just how large it can be depends on the kind of quantum resources used and the types of noise present. In particular, we establish Fourier growth bounds for DQC _{k} , $\frac{1}{2}$ BQP and BQP algorithms. The bounds we obtain for $\ell = 1, 2, 3$ are summarized in Table 1 and depicted in Figure 3.

DQC _{k} algorithms.

Theorem 1.5. *Let $f(x)$ be the acceptance probability of a d -query DQC _{k} algorithm and $\rho \in \{-1, 1, *\}^N$ be any restriction. Then, for all $\ell \geq 2$, we have*

$$L_{1,\ell}(f|\rho) \leq c^\ell \cdot d^3 \cdot \sqrt{K} \cdot N^{(\ell-2)/2} \cdot \log^{\ell-2}(N) \cdot \sqrt{\ell!}$$

for some constant $c > 0$ where $K = 2^k$.

We prove the $\ell = 2$ version of this in Section 4 and higher levels in Section 6 and show that the dependence on k and N are individually optimal in Section 4.1. Here, the dependence on N is particularly interesting. As we will see in Theorem 1.7, the Fourier growth of DQC₁ algorithms falls short of the growth of general BQP algorithms by a factor of \sqrt{N} at each level.

$\frac{1}{2}$ BQP algorithms. For the $\frac{1}{2}$ BQP model, we are unable to prove $L_{1,3}$ and $L_{1,6}$ bounds. Currently, we do not have any upper bounds on $L_{1,3}, L_{1,6}$ that are stronger than the ones for general BQP

algorithms. Nevertheless, for our applications to Forrelation lower bounds, it turns out that we only need to deal with a certain family of signs, which we are able to do ¹.

Theorem 1.6. *Let $f(x)$ be the acceptance probability of a d -query $\frac{1}{2}$ BQP algorithm and $\rho \in \{-1, 1, *\}^{3N}$ be any restriction. Let $\gamma, \gamma' \in [-1, 1]^{3N}$ and $\alpha(\gamma) \in [-1, 1]^{\binom{3N}{3}}, \beta(\gamma, \gamma') \in [-1, 1]^{\binom{3N}{6}}$ be signs as in [Definition 2.5](#). Then,*

$$L_{1,3}^{\alpha(\gamma)}(f|_{\rho}) \leq O(d^7) \cdot \sqrt{N},$$

$$L_{1,6}^{\beta(\gamma, \gamma')}(f|_{\rho}) \leq O(d^{10}) \cdot \sqrt{N^3}.$$

We prove the level-3 version in [Section 5](#) and the higher levels in [Section 6.1](#). We are unaware if this bound is tight, or if one can derive a similar bound for all families of signs (see [Section 1.7](#))².

BQP algorithms.

Theorem 1.7. *Let $f(x)$ be the acceptance probability of a d -query BQP algorithm and $\rho \in \{-1, 1, *\}^N$ be any restriction. Then,*

$$L_{1,\ell}(f|_{\rho}) \leq c^{\ell} \cdot d \cdot N^{(\ell-1)/2} \cdot \log^{\ell-1}(N) \cdot \sqrt{\ell!}$$

We prove this in [Section 6](#). The dependence on N is tight due to the k -FORRELATION problem. The best-known bound prior to this work is an upper bound of $d^{\ell} \cdot \exp\left(\binom{\ell+1}{2}\right) \cdot N^{(\ell-1)/2}$ for bounded degree- d polynomials due to [\[IRR⁺21\]](#). We see in this expression that the dependence on d, ℓ is of the form $d^{\ell} \cdot \exp(\ell^2/2)$, which is quite large for $\ell \gtrsim \sqrt{d}$, in contrast to our dependence, which is at most $d \cdot \sqrt{\ell!}$. In particular, in the regime where $d \geq \Omega(\sqrt{\log(N)})$ and ℓ is constant, our bound is an improvement. We are not aware if this dependence is tight and leave this for future work (see [Section 1.7](#)).

We remark that variants of [Theorem 1.5](#) and [Theorem 1.7](#) hold even with classical pre-processing. The proof of this is quite simple and similar to ideas in [\[GSTW24\]](#) and is deferred to [Appendix A.3](#).

Comparison to Prior Works. While Fourier growth has been extensively studied for classical algorithms, we are aware of only a few works that explicitly study the Fourier growth of quantum algorithms [\[AG23, GSTW24, IRR⁺21\]](#). Among these, [\[IRR⁺21\]](#) and [\[GSTW24\]](#) are closely related to our work. As mentioned before, [\[IRR⁺21\]](#) establishes bounds on the Fourier growth of BQP algorithms that is slightly weaker than ours; furthermore, their bounds apply to all bounded low-degree polynomials and consequently cannot be used to distinguish between BQP, $\frac{1}{2}$ BQP and DQC₁.

The work of [\[GSTW24\]](#) is especially closely related to our work. They study quantum algorithms with k rounds of parallel queries and show that reducing the number of rounds even by one can cause a large blowup in the quantum query complexity. They achieve this by showing Fourier growth bounds for k -round quantum algorithms and leveraging the differences between the bounds for different k . Our work shares some conceptual similarities with their work, particularly in

¹ We observe that [\[BS21\]](#) show that to establish lower bounds for 3-FORRELATION, one only needs to prove signed-Fourier growth bounds for a particular family of signs. (See [Definition 2.5](#) and [Theorem 2.6](#) for more details.) When we refer to the Fourier growth of $\frac{1}{2}$ BQP algorithms, we typically mean signed-Fourier growth for signs as in [Theorem 2.6](#) and [Definition 2.5](#).

²We remark for this family of signs, BQP algorithms can already achieve a significantly larger Fourier growth. In particular, consider the acceptance probability $f(x)$ of the two-query BQP algorithm that solves 3-FORRELATION. For $\gamma = (1, \dots, 1)$, one can show that $L_{1,3}^{\alpha(\gamma)}(f) = \Omega(N)$.

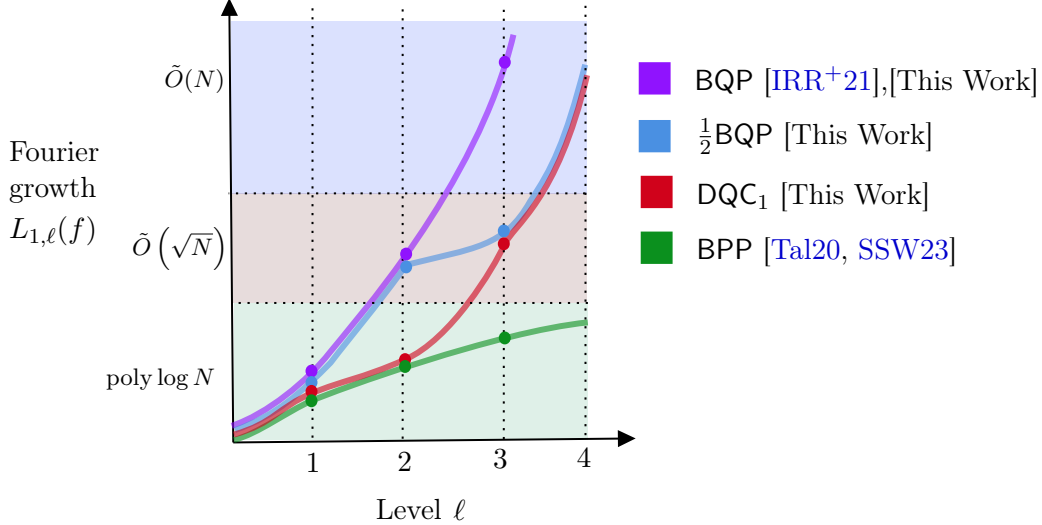


Figure 3: Fourier growth of acceptance probability of algorithms with $d = \text{poly log } N$ queries.

leveraging Fourier growth bounds to distinguish between quantum models, and also in using similar techniques for storing information about parities within matrix products. However, the models of quantum computation we consider are completely different. In [GSTW24], the number of rounds is constrained, while the number of clean qubits is unlimited and the initial state is the all-zeroes state. In contrast, in our setting, the number of clean qubits is constrained and the initial state is forced to be highly mixed, while the number of rounds is allowed to be large. These differing constraints lead to fundamentally different behaviors. Consequently, our techniques diverge from theirs and we require distinct ideas and develop new techniques.

It is worth emphasizing that the idea of using Fourier growth to distinguish between low-degree polynomials arising from different types of algorithms traces back to the landmark oracle separation of BQP and PH [RT22]. The core challenge in that setting was that both models admit low-degree polynomial approximations, and Fourier growth was used precisely to tell these polynomials apart.

1.4 Applications

We study the complexity of the Forrelation problem and its variants in the DQC_k and $\frac{1}{2}\text{BQP}$ models. Combining our Fourier growth bound (Theorem 1.5) with the results of [RT22, CHLT19] (see Theorem 2.4) and the upper bounds on 2-FORRELATION from [Aar10, AA15], we immediately obtain the following corollary.

Corollary 1.8. *For any $k \in \mathbb{N}$, the 2-FORRELATION problem on 2^k -bit inputs can be solved by a DQC_k algorithm with success probability at least $2/3$ by making one quantum query, however, any DQC_{k-t} algorithm that makes d quantum queries has success probability at most $\frac{1}{2} + \tilde{O}(d^3) \cdot 2^{-t/2}$.*

In particular, any DQC_{k-t} algorithm that succeeds with probability at least $2/3$ must make at least $\tilde{\Omega}(2^{t/6})$ queries. Setting $k = \log N$, we obtain the following corollary.

Corollary 1.9. *The 2-FORRELATION problem on N -bit inputs, which can be solved with $\log N$ clean qubits and one quantum query, requires $\tilde{\Omega}(N^{c/6})$ queries in the $\text{DQC}_{(1-c)\log N}$ model for all constants $c < 1$. In particular, any DQC_1 algorithm for 2-FORRELATION must make $\tilde{\Omega}(N^{1/6})$ queries.*

We remark that [Corollary 1.9](#) holds even if the algorithm is allowed to make $\tilde{\Omega}(N^{c/6})$ classical pre-processing queries in advance (using clean bits). We derive the following implications of [Corollary 1.9](#).

A Hierarchy Theorem for DQC_k . In this work, we quantify the power that each additional clean qubit gives to quantum algorithms. It is not too difficult to show that any DQC_k algorithm can be simulated by a DQC_{k-t} algorithm without additional queries but with a loss of $2^{\Theta(t)}$ in the advantage ([Claim A.8](#)). [Corollary 1.8](#) shows that this is tight, up to a constant in the exponent. This shows that the number of clean qubits in a quantum algorithm cannot be efficiently reduced, even with a large amount of classical pre-processing on clean bits.

The First Oracle separation between $\frac{1}{2}\text{BQP}$ and DQC_1 . We give the first oracle separation between $\frac{1}{2}\text{BQP}$ and DQC_1 , resolving a conjecture of [\[JM24\]](#). In particular, Jacobs and Mehraban showed that 2-FORRELATION on N -bit inputs is solvable in the $\frac{1}{2}\text{BQP}$ model with two quantum queries and conjectured that it requires $N^{\Omega(1)}$ queries in the DQC_1 model (see open question #1 on page 8 [\[JM24\]](#)). Our work ([Corollary 1.8](#)) proves this conjecture.

A New Oracle separation between BQP and $\frac{1}{2}\text{BQP}$. Jacobs and Mehraban conjectured (see open question #5 on page 8 [\[JM24\]](#)) that 3-FORRELATION is not in $\frac{1}{2}\text{BQP}$ and our work ([Corollary 1.10](#)) resolves this. By combining our Fourier growth bound ([Theorem 1.6](#)) with the results of [\[BS21\]](#) ([Theorem 2.6](#)), we immediately obtain the following corollary.

Corollary 1.10. *The 3-FORRELATION problem on $3N$ -bit inputs, which can be solved by a BQP algorithm with two quantum queries, requires $\tilde{\Omega}(N^{\Omega(1)})$ queries in the $\frac{1}{2}\text{BQP}$ model.*

We remark that while [Corollary 1.10](#) is not the first oracle separation between BQP and $\frac{1}{2}\text{BQP}$, there are some advantages to this new separation. The prior separation (in [\[JM24\]](#)) is as follows: given any oracle O of length 2^n separating BQP and BPP, we can embed it into a larger oracle O' of length 2^{2n} whose first diagonal block is O and all other diagonal entries are 1. It is not too difficult to show that O' separates BQP and models like $\frac{1}{2}\text{BQP}$, DQC_1 , NISQ, and this was formalized in [\[CCHL23, JM24\]](#). The key intuition is that these intermediate models operate on highly mixed states and therefore assign only a vanishingly small weight to the relevant part of O' , namely O itself. Embeddings of the Forrelation problem can thus separate BQP and $\frac{1}{2}\text{BQP}$, but such separations are somewhat unsatisfactory since they do not establish the hardness of the original problem and apply uniformly to all models like $\frac{1}{2}\text{BQP}$, DQC_1 , NISQ. Our lower bound technique circumvents this limitation by directly proving a lower bound for the original 3-FORRELATION problem in the $\frac{1}{2}\text{BQP}$ model.

1.5 Technical Highlight: Matrix Decomposition Lemma

The main recurring technique in our paper is the use of a matrix decomposition lemma (see [Lemma 3.1](#)). This lemma offers a way to encode information about the indices involved in a matrix product and arises naturally in the context of quantum algorithms, as it allows us to encode information about the Fourier coefficients within a sequence of matrix products. We think it might be of independent interest.

Firstly, we observe that the acceptance probability of quantum algorithms can be expressed as a product of matrices with bounded operator norms. To give some intuition, fix $i_1, i_{d+1} \in [N]$. Consider a sequence of unitary matrices U_1, \dots, U_d and let $U[i|j]$ denote the (i, j) th-entry of U .

Consider a BQP algorithm that starts with the initial state $|i_1\rangle$, evolves it according to the unitary operators U_1, \dots, U_d , interleaved with phase oracles O_x and finally measures the qubits and accepts if the outcome is $|i_{d+1}\rangle$. The acceptance probability of this algorithm is given by $|f(x)|^2$ where

$$\begin{aligned} f(x) &:= \langle i_1 | U_1 \cdot O_x \cdot U_2 \cdot O_x \cdots O_x \cdot U_d | i_{d+1} \rangle \\ &= \sum_{i_2, \dots, i_d} \left(\prod_{t \in [d]} U_t[i_t | i_{t+1}] \right) \cdot \left(\prod_{t \in [d] \setminus \{1\}} x_{i_t} \right) \end{aligned}$$

More generally, by allowing the matrices U_1, \dots, U_d to be arbitrary matrices with spectral norm at most 1 and by adding workspace, we can produce a similar expression for $f(x)$ which *equals* the acceptance probability of an arbitrary $\lfloor d/2 \rfloor$ -query BQP algorithm (see [Claim 2.8](#)). There are other expressions for capturing the acceptance probability of DQC_k and $\frac{1}{2}\text{BQP}$ algorithms using matrix products (see [Claim 2.10](#) and [Claim 2.12](#)). Now that we have an expression for the acceptance probability, we need to compute the Fourier coefficients. Observe that for all $S \subseteq [N]$,

$$\hat{f}(S) = \sum_{i_2, \dots, i_d} \prod_{t \in [d]} U_t[i_t | i_{t+1}] \cdot \mathbb{1}[S = \{i_2\} \oplus \dots \oplus \{i_d\}].$$

Our main idea is to try and encode information about the Fourier coefficients inside a product of matrices with bounded norms. The hope is that since $f(x)$ itself is a product of matrices with bounded norms, so are its Fourier coefficients. To illuminate the main idea, say we wish to multiply the matrices U_1, \dots, U_d to get a matrix U where

$$U[i_1 | i_{d+1}] = \sum_{i_2, \dots, i_d} \prod_{t \in [d]} U_t[i_t | i_{t+1}]$$

but additionally, we wish to retain information about the symmetric difference of the intermediate indices $\{i_2\}, \dots, \{i_d\}$ until the very end. More formally, we wish to design a matrix \tilde{U} whose rows are indexed by i_1 and columns by $i_{d+1} S_{d+1}$ such that

$$\tilde{U}[i_1 | i_{d+1} S_{d+1}] = \sum_{i_2, \dots, i_d} \prod_{t \in [d]} U_t[i_t | i_{t+1}] \cdot \mathbb{1}[S_{d+1} = \{i_2\} \oplus \dots \oplus \{i_d\}].$$

Here, the indicator function ensures that for each S_{d+1} , the corresponding entry of the final matrix only involves contributions from indices that satisfy the parity condition with respect to S_{d+1} . The reason we want to do this is clear; the entry $\tilde{U}[i_1 | i_{d+1} S]$ precisely equals the Fourier coefficient $\hat{f}(S)$. Thus, by reading off the entries of matrix \tilde{U} restricted to rows corresponding to i_1 and columns corresponding to i_{d+1} , we would obtain the list of all Fourier coefficients. The challenge lies in constructing such a matrix \tilde{U} with bounded norms and this is precisely achieved by [Lemma 3.1](#). It embeds the required combinatorial information about the indices within a matrix product while maintaining control over the norms of \tilde{U} . We also show an improved matrix decomposition lemma ([Lemma 3.2](#)) that allows slightly more complex predicates of the indices being summed over – in particular, we allow the imposition of inequality constraints over indices being summed over.

We remark that [\[GSTW24\]](#) implicitly proves another matrix decomposition lemma with a few key differences – their bounds are for algorithms with a small fixed number of rounds but can handle parallel queries, and they only require bounds on the spectral norms of the underlying matrices. In our work, to handle DQC_k and $\frac{1}{2}\text{BQP}$ algorithms that can have a large number of rounds, we need a different kind of matrix decomposition and crucially, we require bounds on the Frobenius norms of the matrices in the decomposition, as well as the ability to impose inequality constraints over indices being summed over. This part is fundamentally new and requires additional work to prove.

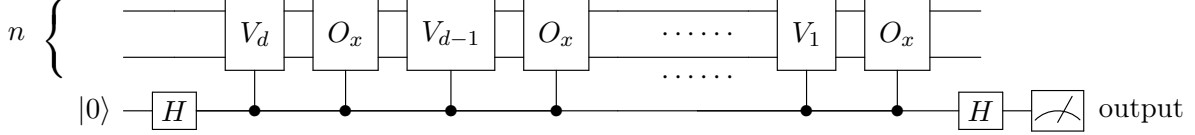


Figure 4: A simple example of a d -query DQC_1 algorithm. The initial state on the first n qubits is maximally mixed.

1.6 Proof Sketch

In general, proving Fourier growth bounds is quite challenging and technically involved. A major challenge arises from the need to incorporate the signs $\alpha_S \in [-1, 1]$ into the matrix product given by the matrix decomposition lemma, and also from the need to sum over all sets S of size ℓ . Introducing the signs in a naive fashion often blows up the operator norms of the underlying matrices, making it difficult to maintain control over the Fourier growth. The heart of our proof involves techniques to incorporate these signs while keeping the operator norms bounded. This step turns out to be especially challenging for $\frac{1}{2}\text{BQP}$ algorithms and we are unable to handle arbitrary signs α_S . However, we are able to successfully encode the signs that arise from the 3-FORRELATION problem.

In this section, we present the simplest part of our proof: using the matrix decomposition lemma (Lemma 3.1) to establish Fourier growth bounds for DQC_1 algorithms. We will make some simplifications: we only focus on level $\ell = 2$; we will assume that there is no restriction ρ on the inputs; and we will only consider algorithms with one clean qubit of a special form in Figure 4. These simplifications are only for the proof sketch and still give enough intuition for the general case.

Firstly, it is not too difficult to derive an expression for acceptance probability of the algorithm in Figure 4. This is given by $\frac{1}{2} + \frac{1}{2}f(x)$ where

$$\begin{aligned}
 f(x) &:= \frac{1}{N} \text{Tr} (O_x \cdot V_1 \cdot O_x \cdot V_2 \cdots O_x \cdot V_d) \\
 &= \frac{1}{N} \sum_{i_1, \dots, i_d \in [N]} \left(\prod_{t \in [d]} V_t[i_t | i_{t+1}] \right) \cdot \left(\prod_{t \in [d]} x_{i_t} \right)
 \end{aligned} \tag{1}$$

where $V_1 \dots, V_d$ are the $N \times N$ unitary matrices applied by the algorithm and we use the convention that $i_{d+1} = i_1$. One can derive a similar expression for the acceptance probability of an arbitrary DQC_k algorithm (see Claim 2.10 for more details). Let us now compute the Fourier coefficients of the acceptance probability, which equals (up to a factor of $1/2$) the Fourier coefficients of $f(x)$, which are easy to read off of Equation (1). For any $S \subseteq [N]$, the S -th Fourier coefficient of f is given by

$$\widehat{f}(S) = \frac{1}{N} \sum_{i_1, \dots, i_d \in [N]} \left(\prod_{t \in [d]} V_t[i_t | i_{t+1}] \right) \cdot \mathbb{1}[\{i_1\} \oplus \dots \oplus \{i_d\} = S]. \tag{2}$$

The quantity we wish to bound is the level-2 Fourier growth of f , i.e., $L_{1,2}(f) = \max_{\alpha} L_{1,2}^{\alpha}(f)$, where

$$L_{1,2}^{\alpha}(f) \triangleq \sum_{|S|=2} \alpha_S \cdot \widehat{f}(S) \tag{3}$$

for signs $\alpha_S \in [-1, 1]$ for $S \subseteq [N]$ of size 2. Fix any such signs α . Substituting the expression for Fourier coefficients $\widehat{f}(S)$ (Equation (2)) in the expression for $L_{1,2}^{\alpha}(f)$ (Equation (3)), we see that

our goal is to upper bound

$$L_{1,2}^\alpha(f) = \sum_{|S|=2} \alpha_S \cdot \frac{1}{N} \sum_{i_1, \dots, i_d \in [N]} \left(\prod_{t \in [d]} V_t[i_t | i_{t+1}] \right) \cdot \mathbb{1}[\{i_1\} \oplus \dots \oplus \{i_d\} = S]. \quad (4)$$

Decomposing $L_{1,2}^\alpha$ into a few terms. First, we will group the terms in Equation (4) into a few terms. We will express $L_{1,2}(f)^\alpha$ as a sum over pairs (t_1, t_2) such that $t_1 \neq t_2 \in [d]$ of a quantity Δ_{t_1, t_2}^α . We describe this below.

Observe that for a term to contribute to Equation (4), the symmetric difference of i_1, \dots, i_d has size 2. In this case, there must exist a pair of indices $t_1 < t_2 \in [d]$ such that i_{t_1} and i_{t_2} are distinct and the symmetric difference of the rest of the i_t is the empty set. More precisely, if $\{i_1\} \oplus \dots \oplus \{i_d\} = S$ for a set S of size 2, then

$$\exists t_1 < t_2 \in [d] \text{ such that } \{i_{t_1}, i_{t_2}\} = S \text{ and } \bigoplus_{t \in [d] \setminus \{t_1, t_2\}} \{i_t\} = \emptyset.$$

Conversely, any such $t_1, t_2 \in [d]$ and i_1, \dots, i_d satisfying the above equation defines a unique $S = \{i_{t_1}, i_{t_2}\}$. For any pair of indices $t_1 < t_2 \in [d]$, let Δ_{t_1, t_2}^α be the contribution of the corresponding terms to $L_{1,2}^\alpha(f)$, i.e.,

$$\Delta_{t_1, t_2}^\alpha := \frac{1}{N} \sum_{i_{t_1} \neq i_{t_2} \in [N]} \alpha_{\{i_{t_1}, i_{t_2}\}} \cdot \sum_{\substack{i_{t_1+1}, \dots, i_{t_2-1} \in [N] \\ i_{t_2+1}, \dots, i_{t_1-1} \in [N]}} \left(\prod_{t \in [d]} V_t[i_t | i_{t+1}] \right) \cdot \mathbb{1} \left[\bigoplus_{t \in [d] \setminus \{t_1, t_2\}} \{i_t\} = \emptyset \right]. \quad (5)$$

Ideally, we would like to say that $L_{1,2}^\alpha(f) = \sum_{t_1 < t_2 \in [d]} \Delta_{t_1, t_2}^\alpha$ and to bound the latter quantity, we observe that there are $O(d^2)$ choices of $t_1 < t_2 \in [d]$ and for any such choice, we will show in the second step that $\Delta_{t_1, t_2}^\alpha \leq 1$, obtaining $L_{1,2}^\alpha(f) \leq O(d^2)$ as desired. Unfortunately, it is not true that $L_{1,2}^\alpha(f) = \sum_{t_1 < t_2 \in [d]} \Delta_{t_1, t_2}^\alpha$, as t_1, t_2 are not uniquely defined for a given set of indices i_1, \dots, i_d .³ In order to address this, we take two different approaches for DQC₁ and $\frac{1}{2}$ BQP algorithms. In the former case, we let t_1 denote the first time an element of S is seen, and let t_2 denote the next time an element of S is seen. As a result, we will have to modify the definition of Δ_{t_1, t_2}^α as follows.

$$\begin{aligned} \Delta_{t_1, t_2}^\alpha := & \frac{1}{N} \sum_{i_{t_1} \neq i_{t_2} \in [N]} \alpha_{\{i_{t_1}, i_{t_2}\}} \cdot \sum_{\substack{i_{t_1+1}, \dots, i_{t_2-1} \in [N] \\ i_{t_2+1}, \dots, i_{t_1-1} \in [N]}} \left(\prod_{t \in [d]} V_t[i_t | i_{t+1}] \right) \cdot \mathbb{1} \left[\bigoplus_{t \in [d] \setminus \{t_1, t_2\}} \{i_t\} = \emptyset \right] \\ & \cdot \prod_{t \in [1, t_1)} \mathbb{1}[i_t \neq i_{t_1}] \cdot \prod_{t \in [1, t_2)} \mathbb{1}[i_t \neq i_{t_2}]. \end{aligned}$$

For the $\frac{1}{2}$ BQP bound on the other hand, we take an alternate approach where we don't uniquely identify t_1, t_2 , instead, we indeed sum over all Δ_{t_1, t_2} , each occurring with a coefficient that precisely cancels out to give an expression for the $L_{1,2}^\alpha(f)$. For the rest of the proof overview, we ignore these subtleties and imagine for now that $L_{1,2}^\alpha(f) = \sum_{t_1 < t_2 \in [d]} \Delta_{t_1, t_2}^\alpha$ for Δ_{t_1, t_2}^α as in Equation (5) and proceed.

³We thank Francisco Escudero Gutierrez and Miquel Saucedo Cuesta for pointing this out.

Showing that $\Delta_{t_1, t_2}^\alpha \leq 1$. This is where we will use the matrix decomposition lemma ([Lemma 3.1](#)). We will group the terms $t \in [d]$ into circular intervals $[t_1, t_2)$ and $[t_2, t_1)$ ⁴. We will apply the matrix decomposition lemma on $V_{t_1}, \dots, V_{t_2-1}$ to remember the symmetric difference of $\{i_t\}$ for $t \in (t_1, t_2)$ and similarly on the matrices $V_{t_2}, \dots, V_{t_1-1}$ to remember the symmetric difference of $\{i_t\}$ for $t \in (t_2, t_1)$ and then enforce equality between these sets. More precisely, apply [Lemma 3.1](#) (with $T = \emptyset$) on the matrices $V_{t_1}^\rho, \dots, V_{t_2-1}^\rho$ to obtain $\tilde{V}_{[t_1, t_2)}$ and to $V_{t_2}^\rho, \dots, V_{t_1-1}^\rho$ backwards to obtain $\tilde{V}_{[t_2, t_1)}$ such that for all $i_{t_1}, i_{t_2} \in [N], S_{t_2} \subseteq [N]$,

$$\tilde{V}_{[t_1, t_2)}[i_{t_1} | i_{t_2} S_{t_2}] = \sum_{i_t \in [N] \text{ for } t \in (t_1, t_2)} \left(\prod_{t \in [t_1, t_2)} V_t[i_t | i_{t+1}] \right) \cdot \mathbb{1} \left[\bigoplus_{t \in (t_1, t_2)} \{i_t\} = S_{t_2} \right], \quad (6)$$

$$\tilde{V}_{[t_2, t_1)}[i_{t_1} | i_{t_2} S_{t_2}] = \sum_{i_t \in [N] \text{ for } t \in (t_2, t_1)} \left(\prod_{t \in [t_2, t_1)} V_t[i_t | i_{t+1}] \right) \cdot \mathbb{1} \left[\bigoplus_{t \in (t_2, t_1)} \{i_t\} = S_{t_2} \right]. \quad (7)$$

Substituting [Equations \(6\) and \(7\)](#) in [Equation \(5\)](#), we see that

$$\begin{aligned} \Delta_{t_1, t_2}^\alpha &\triangleq \frac{1}{N} \sum_{i_{t_1} \neq i_{t_2} \in [N]} \alpha_{\{i_{t_1}, i_{t_2}\}} \sum_{S_{t_2} \subseteq [N]} \tilde{V}_{[t_1, t_2)}[i_{t_1} | i_{t_2} S_{t_2}] \cdot \tilde{V}_{[t_2, t_1)}[i_{t_1} | i_{t_2} S_{t_2}] \\ &\leq \frac{1}{N} \sum_{i_{t_1} \neq i_{t_2} \in [N]} \sum_{S_{t_2} \subseteq [N]} \left| \tilde{V}_{[t_1, t_2)}[i_{t_1} | i_{t_2} S_{t_2}] \right| \cdot \left| \tilde{V}_{[t_2, t_1)}[i_{t_1} | i_{t_2} S_{t_2}] \right| \quad (\text{since } \alpha_{\{i_{t_1}, i_{t_2}\}} \in [-1, 1]) \\ &\leq \frac{1}{N} \cdot \left\| \tilde{V}_{[t_1, t_2)} \right\|_{\text{frob}} \cdot \left\| \tilde{V}_{[t_2, t_1)} \right\|_{\text{frob}} \quad (\text{Fact 2.3}) \end{aligned}$$

Firstly, observe that

$$\max \left(\left\| \tilde{V}_{[t_1, t_2)} \right\|_{\text{frob}}, \left\| \tilde{V}_{[t_2, t_1)} \right\|_{\text{frob}} \right) \leq \sqrt{N}.$$

This is because both matrices have operator norm at most one and either have at most N rows or N columns. This implies that $\Delta_{t_1, t_2}^\alpha \leq N^{-1} \cdot N \leq 1$. This completes the proof sketch.

We now describe some of the additional ideas involved in generalizing this proof.

Generalizing to higher levels. We provide a general bootstrapping argument that proves higher level Fourier growth bounds assuming bounds for lower levels. This is applicable to any restriction-closed family of boolean functions and allows us to prove improved bounds on the Fourier growth of BQP algorithms, as well as higher-level and level-6 bounds for DQC_k and $\frac{1}{2}$ BQP algorithms respectively.

$\frac{1}{2}$ BQP algorithms. It is not too hard to show that the expression for the acceptance probability of a d -query $\frac{1}{2}$ BQP algorithm is quite similar to [Equation \(1\)](#), except, there are $2d + 2$ matrices V_1, \dots, V_{2d+2} , and more importantly, there is an extra term of the form $F_{i_1, i_{d+1}} \in \{0, 1\}$ inside the summation, which corresponds to the post-processing of the measurement outcomes of the initial and final states. (See [Equation \(55\)](#) and [Claim 2.12](#) for a formal expression.) This additional term

⁴We arrange $1, \dots, d$ in a clock-wise circle and define the intervals clock-wise. For instance, the interval $[d-2, 2]$ refers to the set $\{d-2, d-1, d, 1, 2\}$. The intervals (t_1, t_2) and (t_2, t_1) are well-defined but would be empty if $t_2 = t_1 \pm 1$ modulo d . In each of these cases, it is understood that the summation over $i_{t_1+1}, \dots, i_{t_2-1}$ and $i_{t_2+1}, \dots, i_{t_1-1}$ respectively is to be ignored.

$F_{i_1, i_{d+1}}$ is challenging to incorporate while keeping the norms bounded. As a result, proving bounds for $\frac{1}{2}$ BQP algorithms turns out to be more technically involved. We need to use an improved matrix decomposition lemma ([Lemma 3.6](#)).

Furthermore, we are only able to prove level-3 and level-6 Fourier growth bounds for a particular family of signs as in [Definition 2.5](#). The reason why the signs $\alpha(\gamma)$ and $\beta(\gamma)$ in [Definition 2.5](#) are easier to deal with than general signs, is that once we fix i_2 , $\alpha(\gamma)_{i_1, i_2, i_3}$ becomes a product of three terms, the first depending only on i_1 , the second on i_3 and the third on γ in a product fashion. Similarly, once we fix i_2, i_5 , then $\beta(\gamma)_{i_1, \dots, i_6}$ becomes a product of five terms, the first depending only on i_1 , the second on i_4 , the third on i_3 , the fourth on i_6 , and the fifth on γ in a product fashion. These kinds of signs that are products across the indices are much easier to handle than general families of signs and often exhibit a Fourier growth that is much smaller than the Fourier growth for arbitrary signs⁵. We then show that summing over the i_2 , or over the i_2, i_5 doesn't blow up the Fourier growth by much. (See [Section 5](#) for more details.)

1.7 Outlook & Future Directions

Broadly, our results suggest that Fourier growth provides a powerful analytic lens to separate models of quantum computation. Several natural next steps emerge in this direction and we highlight some open questions in this section.

1. **Fourier Growth of NISQ.** Researchers have attempted to model NISQ (noisy intermediate scale quantum) algorithms through the lens of query complexity, in the hopes of understanding the computational power of near-term quantum devices [[CCHL23](#), [CHHK24](#)]. There has been recent interest in using 2-FORRELATION to show quantum advantages in near-term experiments [[Geo25](#), [Shu25](#)] and this prompts the natural question, can we solve 2-FORRELATION in NISQ? If not, can we prove bounds on the Fourier growth of NISQ?
2. **The Power of DQC_1 .** Where does DQC_1 fit within the landscape of classical complexity, and in particular, is it contained in PH? The differences between the Fourier growth of DQC_1 and PH are quite stark, but it is not clear how to leverage this into an oracle separation, as existing approaches rely on the Forrelation problem, which is hard for DQC_1 . Developing new techniques here would not only clarify the power of DQC_1 , but also expand the toolkit for proving lower bounds on classical computation.
3. **The Power of IQP.** Another intriguing intermediate model is IQP, whose power derives from its ability to perform Fourier sampling. How does this model compare to DQC_1 and $\frac{1}{2}$ BQP? Understanding the relationship between these models would help chart the intermediate landscape between BPP and BQP and reveal the relative power of various quantum capabilities like Fourier sampling and trace estimation. It was shown by [[JM24](#)] that IQP can be simulated by $\frac{1}{2}$ BQP and they conjectured that this containment is strict. Is 2-FORRELATION solvable in IQP⁶ and if not, can we prove Fourier growth bounds?
4. **Tight Bounds on the Fourier Growth of Quantum Algorithms.** Finally, many of our upper bounds on the Fourier growth are not known to be tight. Are the dependencies on d and ℓ tight in [Theorems 1.5](#) to [1.7](#)? What is the Fourier growth of $\frac{1}{2}$ BQP with respect to arbitrary families of signs? Tight bounds on Fourier growth could provide a precise handle for

⁵Indeed, for general bounded degree- d polynomials, the level- ℓ Fourier growth with arbitrary signs can be as large as $N^{\Omega(\ell)}$, whereas for signs that are a product across the indices, the Fourier growth is at most $d^{O(\ell)}$ [[IRR⁺21](#)].

⁶This question has since been resolved affirmatively by [[BC26](#)].

quantum computational power, and help map the landscape between classical, intermediate, and fully quantum models.

1.8 Organization.

Section 2 consists of preliminaries, where we formally describe the various models of computation and state the results we need from prior works on Forrelation. In **Section 3**, we describe and prove some of the basic matrix decomposition lemmas (**Lemmas 3.1** and **3.2**). We prove our Fourier growth bounds for DQC_k in **Section 4** (proof of **Theorem 1.5**), $\frac{1}{2}\text{BQP}$ in **Section 5** (proof of **Theorem 1.6**) and BQP in **Section 6** (proof of **Theorem 1.7**).

2 Preliminaries & Notation

Restrictions. For a restriction $\rho \in \{-1, 1, *\}^N$ and a vector $x \in \{-1, 1\}^N$, the i -th coordinate of the restricted vector $\rho(x) \in \{-1, 1\}^N$ is ρ_i if $\rho_i \in \{-1, 1\}$ and x_i if $\rho_i = *$ for $i \in [N]$. For a boolean function $f : \{-1, 1\}^N \rightarrow \mathbb{R}$, and a restriction $\rho \in \{-1, 1, *\}^N$, we use $f|_\rho$ to denote the restricted function which maps x to $f(\rho(x))$ for $x \in \{-1, 1\}^N$.

Sets. For $x \in \mathbb{R}^N$ and $S \subseteq [N]$, we use $\chi_S(x)$ or x_S to denote $\prod_{i \in S} x_i$. For indices $i_1, \dots, i_k \in [N]$, we use $\{i_1\} \oplus \dots \oplus \{i_k\}$ to denote the symmetric difference $\oplus_{t \in [k]} \{i_t\}$ and similarly $S_1 \oplus S_2$ denotes the symmetric difference of the sets S_1 and S_2 .

We will often use uppercase letters to denote 2 to the power of lowercase letters, in particular, $N = 2^n, W = 2^w, K = 2^k$ and $M = 2^m$.

Circular Intervals. For $i, j \in [n]$, we use $[i, j]$ to denote the clockwise sequence of points from i to j when $1, \dots, n$ are arranged clock-wise in a circle. For example, $[n, 2] = \{n, 1, 2\}$ and $[1, 3] = \{1, 2, 3\}$. We use $(,]$ and $[,)$ and $(,)$ to denote half-open or open intervals.

Vectors and Inner Products. We identify the space $\{0, 1\}^n$ with $[N]$ under the natural correspondence $(a_1, \dots, a_n) \rightarrow 1 + \sum_i a_i 2^{i-1}$. We also identify $\{0, 1\}^n$ with $\{-1, 1\}^n$ under the correspondence that maps 0 to 1 and 1 to -1 . For $u, v \in [N]$, we use $\langle u, v \rangle_2 := \sum_{i \in [n]} u_i v_i \pmod 2$ to denote the inner product over \mathbb{F}_2 under the aforementioned correspondence. For $u \in \mathbb{C}^N$ and $U \in \mathbb{C}^{N \times N}$, we use u^\dagger, U^\dagger to denote the conjugate-transpose. For complex vectors $u, v \in \mathbb{C}^N$, we use $\langle u | v \rangle, v^\dagger u$, and $\langle u, v \rangle$ to denote $\sum_i u_i \bar{v}_i$, the complex inner product.

Matrices. We use \mathbb{I} to denote the identity matrix, where the dimensions are clear from context. We will often encounter matrices whose rows and columns are indexed by (i, w) for $i \in [N], w \in [W]$, or by (i, w, k) for $i \in [N], w \in [W], k \in [K]$. For ease of notation, we use I as a shorthand for (i, w) or (i, w, k) , where the distinction will be clear from the context. For $I_t, I_{t+1} \in [M]$, we use either $U_t[I_t | I_{t+1}]$ or $U_t[I_t, I_{t+1}]$ to denote the (I_t, I_{t+1}) -the entry of U_t . For matrices U_1, \dots, U_d , we use $U_{[t_1, t_2]}$ to denote the product $\prod_{t \in [t_1, t_2]} U_t = U_{t_1} \cdots U_{t_2}$ of the matrices in the circular interval $[t_1, t_2]$ in clockwise order. We define $U_{[t_1, t_2)}, U_{(t_1, t_2]}, U_{(t_1, t_2)}$ analogously.

Definition 2.1 (Hadamard Matrix). *For $N = 2^n$, the Hadamard matrix H_N is defined to be*

$$H_N = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}^{\otimes n}.$$

Matrix Norms & Inequalities. Let $\|\cdot\|_{\text{op}}$ and $\|\cdot\|_{\text{frob}}$ denote the spectral and Frobenius norm, or equivalently, the Schatten- ∞ and Schatten-2 norms. The following basic fact follows from Holder's Inequality for Schatten norms.

Fact 2.2. *Let A, B, C be rectangular matrices with $A = BC$. Then,*

$$\|A\|_{\text{frob}} \leq \min(\|B\|_{\text{op}} \cdot \|C\|_{\text{frob}}, \|B\|_{\text{frob}} \cdot \|C\|_{\text{op}})$$

Proof. $\|A\|_{\text{frob}}^2 \triangleq \text{Tr}(AA^\dagger) = \text{Tr}(BC \cdot C^\dagger B^\dagger) = \text{Tr}(B^\dagger B \cdot CC^\dagger) \leq \|C\|_{\text{op}}^2 \cdot \text{Tr}(B^\dagger B \cdot \mathbb{I}) = \|C\|_{\text{op}}^2 \cdot \|B\|_{\text{frob}}^2$. Applying the same argument on A^\dagger gives the other inequality. \square

The Cauchy-Schwarz inequality implies the following fact.

Fact 2.3. *For rectangular matrices A, B , and any subset T of indices, we have*

$$\sum_{(i,j) \in T} |A[i|j]| \cdot |B[i|j]| \leq \|A\|_{\text{frob}} \cdot \|B\|_{\text{frob}}.$$

2.1 Fourier Growth

Recall the definition of the Fourier growth as in [Definition 1.1](#) and [Definition 1.2](#). For a family of functions \mathcal{F} , we use $L_{1,\ell}(\mathcal{F})$ to denote $\max_{f \in \mathcal{F}} L_{1,\ell}(f)$.

Lower Bounds for Forrelation from Fourier Growth. The results of [\[RT22, CHLT19\]](#) imply that to show lower bounds on the 2-FORRELATION problem, it suffices to prove Fourier growth bounds for level 2.

Theorem 2.4 ([\[RT22, CHLT19\]](#)). *Let \mathcal{F} be any family of $2N$ -variate boolean functions closed under restrictions. Then, the maximum advantage with which \mathcal{F} solves 2-FORRELATION is at most*

$$O\left(\frac{L_{1,2}(\mathcal{F})}{\sqrt{N}}\right).$$

The results of [\[BS21\]](#) imply that to show lower bounds on the 3-FORRELATION problem, it suffices to prove signed-Fourier growth bounds for level 3 and 6, for the following family of signs.

Definition 2.5. *Partition $[3N]$ into $A := [N], B := (N, 2N], C := (2N, 3N]$ and fix any ordering $<$ of the elements in A, B, C . There is a natural correspondence $B \leftrightarrow [N]$ given by $b \leftrightarrow b - N$ for all $b \in B$ and a similar correspondence $C \leftrightarrow [N]$ given by $c \leftrightarrow c - 2N$ for all $c \in C$. Let $\gamma \in [-1, 1]^{3N}, \gamma' \in [-1, 1]^{3N}$. Define $\alpha(\gamma) \in [-1, 1]^{\binom{3N}{3}}$ and $\beta(\gamma, \gamma') \in [-1, 1]^{\binom{3N}{6}}$ as follows. Let $\overline{H} \in \{-1, 1\}^{N \times N}$ be the matrix whose (i, j) -th entry is $(-1)^{(i,j)_2} = \text{sign}(H_N[i|j])$ for $i, j \in [N]$. For $i_1, i_2, i_3 \in [3N]$, let*

$$\alpha(\gamma)_{i_1, i_2, i_3} := \begin{cases} \overline{H}(i_2, i_1) \cdot \overline{H}(i_2, i_3) \cdot \left(\prod_{t \in [3]} \gamma_{i_t}\right) & \text{if } i_1 \in A, i_2 \in B, i_3 \in C \\ 0 & \text{otherwise.} \end{cases}$$

For $i_1, \dots, i_6 \in [3N]$, let

$$\beta(\gamma, \gamma')_{i_1, \dots, i_6} := \begin{cases} \alpha(\gamma)_{i_1, i_2, i_3} \cdot \alpha(\gamma')_{i_4, i_5, i_6} & \text{if } i_1 < i_4 \in A, i_2 < i_5 \in B, i_3 < i_6 \in C \\ 0 & \text{otherwise.} \end{cases}$$

The following theorem is implicit in [BS21].⁷

Theorem 2.6 (Implicit in [BS21]). *Let \mathcal{F} be any family of $3N$ -variate boolean functions that is closed under restrictions. Let $\gamma, \gamma' \in [-1, 1]^{3N}$ and $\alpha(\gamma) \in [-1, 1]^{\binom{3N}{3}}, \beta(\gamma, \gamma') \in [-1, 1]^{\binom{3N}{6}}$ be as in Definition 2.5. Then, the maximum advantage with which \mathcal{F} solves 3-FORRELATION is at most*

$$\max_{\gamma, \gamma' \in [-1, 1]^{3N}} O \left(\frac{L_{1,3}^{\alpha(\gamma)}(\mathcal{F})}{N} + \frac{L_{1,6}^{\beta(\gamma, \gamma')}(\mathcal{F})}{N^2} \right).$$

2.2 Quantum Query Complexity

In the setting of quantum query complexity, the input is accessed by an oracle. This oracle is typically an operator \tilde{O}_x for $x \in \{0, 1\}^N$ which maps $|b\rangle |i\rangle \rightarrow |b \oplus x_i\rangle |i\rangle$ for $b \in \{0, 1\}, i \in [N]$. One can alternatively define an oracle O_x for $x \in \{-1, 1\}^N$ which maps $|b\rangle |i\rangle$ to itself if $b = 0$ and to $|b\rangle |i\rangle x_i$ if $b = 1$ and $i \in [N]$. It is not too difficult to show that these two definitions are equivalent, up to a Hadamard gate on the first qubit. We will work with the oracle O_x and later introduce some additional simplifications.

The most general model of a quantum query algorithm is the BQP model defined below. For the following definition, we interpret n as the number of qubits on which the oracle acts and w as the number of qubits of extra workspace. As mentioned before, we use \mathbb{I} to denote the identity matrix, where the dimension is implicit.

Definition 2.7 (BQP Algorithm with d Queries). *Let $n, w \in \mathbb{N}$, $N = 2^n, W = 2^w$ and $M = NW$. A BQP algorithm acts on $n + w$ qubits initialized to $|0, \dots, 0\rangle$. Let $U_0, U_1, \dots, U_d \in \mathbb{C}^{M \times M}$ be $M \times M$ unitary matrices. The algorithm applies the unitary operators U_0, \dots, U_d interleaved with the oracle $O_x \otimes \mathbb{I}$ and measures all the qubits at the end to obtain an outcome I_{d+1} . The algorithm accepts iff $I_{d+1} \in \mathcal{F}$ where $\mathcal{F} \subseteq [M]$ is a subset. (See Figure 5 for a depiction.)*

The following claim expresses the acceptance probability of a d -query BQP algorithm and is not too difficult to prove.

Claim 2.8. *The acceptance probability of a d -query BQP algorithm can be expressed as*

$$f(x) := \langle v | O \cdot V_1 \cdots V_{2d-1} \cdot O | v \rangle$$

where $O = O_x \otimes \mathbb{I}$, $V_1, \dots, V_{2d-1} \in \mathbb{C}^{M \times M}$ are matrices with $\|V_t\|_{\text{op}} \leq 1$ for all $t \in [2d-1]$ and $v \in \mathbb{C}^M$ is a unit vector.

In the following sections, we will define DQC_k and $\frac{1}{2}\text{BQP}$ algorithms.

2.3 DQC_k algorithms

We interpret n as the number of qubits on which the oracle acts, k as the number of clean qubits, and w as the number of qubits of extra workspace.

⁷In particular, see equation (5.7) and the equation above in [BS21] for the level-3 contribution and equation (5.13) and the preceding paragraph for the level-6 contribution.

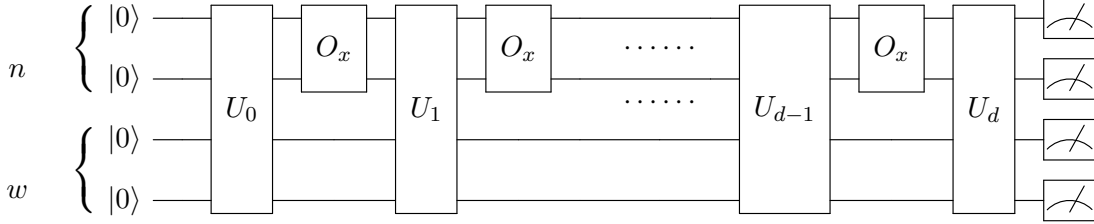


Figure 5: A d -query BQP algorithm.

Definition 2.9 (DQC $_k$ Algorithm with d Queries). *Let $n, w, k \in \mathbb{N}$ and $N = 2^n, W = 2^w, K = 2^k$ and $M = NWK$. A DQC $_k$ algorithm acts on k clean qubits initialized to the $|0 \dots 0\rangle$ state and $n + w$ maximally noisy qubits which consist of n qubits on which the oracle acts and w qubits of workspace. Let $U_1, \dots, U_{d+1} \in \mathbb{C}^{M \times M}$ be $M \times M$ unitary matrices. Let $\mathcal{S} = [NW] \times \{1\}$ be the set of all possible starting basis states of the algorithm and $\mathcal{F} \subseteq [NWK]$ be the subset of final basis states that is accepted by the algorithm. The algorithm starts with a uniformly random basis state sampled from \mathcal{S} , applies the unitary operators U_1, \dots, U_{d+1} , interleaved with the oracle $O_x \otimes \mathbb{I}$, measures all the qubits at the end and accepts if the outcome is in \mathcal{F} . (See [Figure 1](#) for a depiction.)*

Remark. In our model, the oracles are not allowed to directly act on the clean qubits, nevertheless, we can effectively implement this type of operation by swapping the clean qubits with the noisy qubits, applying the oracle on those noisy qubits and swapping them back with the clean qubits. While this transformation does require the use of k extra (potentially noisy) qubits to do the swap operation, our formalism has the advantage that we can talk about oracle separations where k , the number of clean qubits is significantly smaller than n , where the length of the input is 2^n . This is important, since when $k \gg n$, many problems become solvable with a few quantum queries with $O(k)$ clean qubits.

We will now provide an expression for the acceptance probability of a DQC $_k$ algorithm, which we will prove in the appendix ([Appendix A.5](#)). As mentioned before, estimating the trace of a unitary matrix described by a quantum circuit is known to be complete for the class DQC $_1$ [[KL98](#)] and a similar statement is true in query complexity as well.

Claim 2.10. *The acceptance probability of a d -query DQC $_k$ algorithm can be expressed as*

$$f(x) = (NW)^{-1} \cdot \text{Tr}(O \cdot V_1 \cdots O \cdot V_{2d})$$

where $O = O_x \otimes \mathbb{I}$, $V_1, \dots, V_{2d} \in \mathbb{C}^{M \times M}$ satisfy $\|V_t\|_{\text{op}} \leq 1$ for $t \in [2d]$, furthermore, $\|V_1\|_{\text{frob}} \leq \sqrt{NW}$.

2.4 $\frac{1}{2}$ BQP algorithms

We interpret n as the number of qubits on which the oracle acts and w as the number of qubits of extra workspace.

Definition 2.11 ($\frac{1}{2}$ BQP Algorithm with d Queries). *Let $n, w \in \mathbb{N}$, $N = 2^n, W = 2^w$ and $M = NW$. A $\frac{1}{2}$ BQP algorithm acts on $n + w$ qubits initialized to $|I_1\rangle$ for a uniformly random $I_1 \sim [M]$. The algorithm does not have knowledge of I_1 . Let $U_1, \dots, U_{d+1} \in \mathbb{C}^{M \times M}$ be $M \times M$ unitary matrices. The algorithm applies the unitary operators U_1, \dots, U_{d+1} interleaved with the oracle $O_x \otimes \mathbb{I}$ and measures all the qubits at the end to obtain an outcome I_{d+2} . Finally, the algorithm then learns I_1 . The algorithm accepts iff $(I_1, I_{d+2}) \in \mathcal{F}$ where $\mathcal{F} \subseteq [M] \times [M]$ is a subset. (See [Figure 2](#) for a depiction.)*

We provide an expression for the acceptance probability of a d -query $\frac{1}{2}$ BQP algorithm, which is proved in [Appendix A.5](#).

Claim 2.12. *The acceptance probability of a d -query $\frac{1}{2}$ BQP algorithm can be expressed as*

$$f(x) := M^{-1} \sum_{I_1, I_{d+2} \in [M]} F_{I_1, I_{d+2}} \cdot \langle I_1 | U_1^\dagger \cdot O \cdots O \cdot U_{d+1}^\dagger | I_{d+2} \rangle \langle I_{d+2} | U_{d+1} \cdot O \cdots O \cdot U_1 | I_1 \rangle$$

where $O = O_x \otimes \mathbb{I}$, and $U_1, \dots, U_{d+1} \in \mathbb{C}^{M \times M}$ are matrices with $\|U_t\|_{\text{op}} \leq 1$ for all $t \in [d+1]$.

Some Remarks.

- While our way of defining DQC_k and $\frac{1}{2}$ BQP doesn't clearly subsume BPP, there is a simple way to fix this. We can define variants of these models where the algorithm is allowed to make up to d classical pre-processing queries on clean bits, and based on the query outcomes, choose a d -query quantum algorithm to run. When defined this way, these models immediately subsume BPP, since we can implement any BPP algorithm in the pre-processing part. Interestingly, many of the results in our paper, especially the lower bounds hold even for algorithms with a large amount of classical pre-processing. See [Appendix A.3](#) for more details.
- Unlike [\[GSTW24\]](#), our model does not allow parallel queries. This is without loss of generality, as our model has unrestricted depth and we can simulate k parallel queries by k adaptive queries. If we allow parallel queries but limit the depth, we suspect that it might lead improved Fourier growth bounds in terms of the depth of the algorithm, but we leave this to future work.
- In the rest of this paper, we will work with the oracle O'_x which maps $|i\rangle$ to $|i\rangle x_i$ for all $i \in [N]$ where x is of length N . Note that the aforementioned oracle O_x is the controlled version of O'_x and generally offers more functionality than O'_x . However, in all our proofs, it suffices to work with the oracle O'_x since we allow restrictions $\rho \in \{-1, 1, *\}^N$ to act on our input. In particular, if we consider O'_x for bit-strings of length $2N$ and apply the restriction which fixes the first N coordinates to 1, we obtain the oracle O_x on bit-strings of length N as desired. Since all our Fourier growth bounds work even under restrictions of the input, it suffices to work with oracles of the form O'_x and all our Fourier growth bounds will carry over to oracles of the form O_x if N is replaced by $2N$. Henceforth, we will refer to the oracle O'_x as O_x and work with this oracle.

3 Main Technical Tool: Matrix Decomposition Lemma

The following matrix decomposition lemma is a recurring tool in this paper. It allows us to encode information about the indices in a matrix multiplication by embedding them inside a larger matrix multiplication. In this lemma, we have matrices U_1, \dots, U_d where the rows and columns of U_t are indexed by I_t and I_{t+1} respectively. Here, I is a shorthand for either (i, w, k) or (i, w) where $i \in [N]$ corresponds to indices we want to remember information about and $w \in [W], k \in [K]$ corresponds to auxiliary workspace indices. The set T corresponds to the complement of matrices whose index information we want to retain, i.e., we don't care about the matrices in T . The set L indicates that we do not store parity information for indices i_t with $i_t \notin L$ and the set S_{d+1} corresponds to the information aggregated after multiplying the matrices.

Lemma 3.1. Let U_1, \dots, U_d be $M \times M$ matrices with $\|U_t\|_{\text{op}} \leq 1$ for $t \in [d]$ and let $T \subseteq [d]$ and $L \subseteq [N]$. Define a matrix \tilde{U} such that for all $I_1, I_{d+1} \in [M], S_{d+1} \subseteq [N]$,

$$\tilde{U}[I_1|I_{d+1}S_{d+1}] = \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \mathbb{1} \left[S_{d+1} = \bigoplus_{\substack{t \in [1, d] \setminus T \\ i_t \in L}} \{i_t\} \right].$$

Then, $\|\tilde{U}\|_{\text{frob}} \leq \min_{t \in [d]} \|U_t\|_{\text{frob}}$.

Proof of Lemma 3.1. Define a diagonal matrix O_x whose i -th entry is a variable x_i if $i \in L$ and is 1 otherwise. Define matrices $V_i(x) = U_i$ if $i \in T$ and $V_i(x) = O_x \cdot U_i$ otherwise. Observe that $\|V_i\|_{\text{op}} \leq 1$ for all $i \in [d]$. Consider the matrix-valued function $M(x) := V_1(x) \cdots V_d(x)$. Observe that

$$M(x)[I_1|I_{d+1}] = \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \prod_{\substack{t \in [1, d] \setminus T \\ i_t \in L}} x_{i_t}$$

Hence, for all $S \subseteq [N]$, we have $\widehat{M}(S)[I_1|I_{d+1}] = \tilde{U}[I_1|I_{d+1}S_{d+1}]$. This implies that

$$\begin{aligned} \|\tilde{U}\|_{\text{frob}}^2 &= \sum_{S \subseteq [N]} \sum_{I_1, I_{d+1} \in [M]} \widehat{M}(S)[I_1|I_{d+1}]^2 \\ &= \sum_{S \subseteq [N]} \|\widehat{M}(S)\|_{\text{frob}}^2 = \mathbb{E}_x [\|M(x)\|_{\text{frob}}^2] \end{aligned} \quad (\text{by Parseval's.})$$

Finally, let $t \in [d]$ and $x \in \{\pm 1\}^N$. Observe that $M(x) = V_{<t}(x) \cdot V_t(x) \cdot V_{>t}(x)$ where $V_{<t}(x) := \prod_{t' < t} V_{t'}(x)$ and $V_{>t}(x) := \prod_{t' > t} V_{t'}(x)$ have operator norm at most 1, hence $\|M(x)\|_{\text{frob}} \leq \|V_t(x)\|_{\text{frob}} = \|U_t\|_{\text{frob}}$. This completes the proof. \square

We will need a slightly more complicated version of this lemma that incorporates inequality constraints over the indices being summed over.

Define functions $g_1, g_2 : [N] \times [N] \rightarrow \mathbb{R}$ as follows. $g_1(i, j) = \mathbb{1}[i \neq j]$ and $g_2(i, j) = (-1)^{\mathbb{1}[i=j]}$.

Lemma 3.2. Let U_1, \dots, U_d be $M \times M$ matrices with $\|U_t\|_{\text{op}} \leq 1$ for $t \in [d]$ and let $t_1 \in [d]$. For all $\gamma \in [2]$, define a matrix U_γ such that for all $I_1, I_{d+1} \in [M]$

$$U_\gamma[I_1|I_{d+1}] = \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \prod_{t \in (t_1, d]} g_\gamma(i_t, i_{t_1}).$$

Then, for all $\gamma \in [2]$, $\|U_\gamma\|_{\text{frob}} \leq O(d) \cdot \min_{t \in [d]} \|U_t\|_{\text{frob}}$ and $\|U_\gamma\|_{\text{op}} \leq O(d)$. Furthermore, $\|U_\gamma\|_{\text{frob}} \leq \min_{t \in [t_1, d]} (O(t) \cdot \|U_t\|_{\text{frob}}, \sqrt{M})$.

Corollary 3.3. Under the same hypothesis as Lemma 3.2, let $T \subseteq [d]$ and $L \subseteq [N]$. Define a matrix \tilde{U} such that for all $I_1, I_{d+1} \in [M], S_{d+1} \subseteq [N]$,

$$\tilde{U}_\gamma[I_1|I_{d+1}S_{d+1}] = \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \mathbb{1} \left[S_{d+1} = \bigoplus_{\substack{t \in [1, d] \setminus T \\ i_t \in L}} \{i_t\} \right] \cdot \prod_{t \in (t_1, d]} g_\gamma(i_t, i_{t_1}).$$

Then, $\|\tilde{U}_\gamma\|_{\text{frob}} \leq O(d) \cdot \min_t \|U_t\|_{\text{frob}}$ and $\|\tilde{U}_\gamma\|_{\text{frob}} \leq O(\sqrt{M})$.

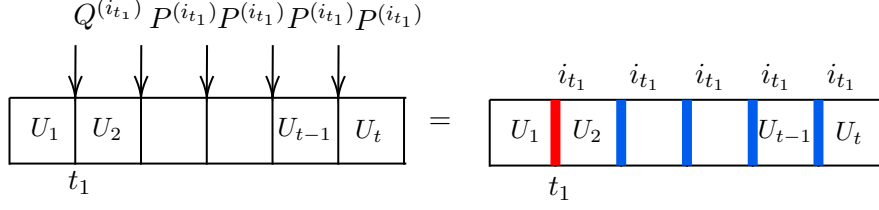


Figure 6: We use red and blue lines indexed by i_{t_1} to depict the insertion of $Q^{(i_{t_1})}$ and $P^{(i_{t_1})}$ respectively, where we omit the superscript γ .

Proof of Corollary 3.3 from Lemma 3.2. Define a diagonal matrix O_x whose i -th entry is a variable x_i if $i \in L$ and is 1 otherwise, and define matrices $V_t = V_t(x) = U_t$ if $t \in T$ and $V_t = V_t(x) = (O_x \otimes \mathbb{I}) \cdot U_t$ otherwise. Apply Lemma 3.2 to $V_1(x), \dots, V_d(x)$ to obtain a matrix $U_\gamma(x)$ with

$$\|U_\gamma(x)\|_{\text{frob}} \leq O(d) \cdot \min_t (\|V_t(x)\|_{\text{frob}}) = O(d) \cdot \min_t (\|U_t\|_{\text{frob}})$$

such that for all $I_1, I_{d+1} \in [M]$,

$$\begin{aligned} U_\gamma(x)[I_1|I_{d+1}] &= \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} V_t(x)[I_t|I_{t+1}] \right) \cdot \prod_{t \in (t_1, d)} g_\gamma(i_t, i_1) \\ &= \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \prod_{\substack{t \in [1, d] \setminus T \\ i_t \in L}} x_{i_t} \cdot \prod_{t \in (t_1, d)} g_\gamma(i_t, i_1). \end{aligned}$$

Observe that for all $S \subseteq [N]$, we have $\widehat{U}(S)[I_1|I_{d+1}] = \widetilde{U}[I_1|I_{d+1}S]$. By Parseval's,

$$\|\widetilde{U}\|_{\text{frob}}^2 = \sum_{S \subseteq [N]} \sum_{I_1, I_{d+1} \in [M]} \left| \widehat{U}(S)[I_1|I_{d+1}] \right|^2 \triangleq \sum_{S \subseteq [N]} \|\widehat{U}(S)\|_{\text{frob}}^2 = \mathbb{E}_x [\|U(x)\|_{\text{frob}}^2].$$

This completes the proof. \square

Proof of Lemma 3.2. For $i \in [N]$, and $\gamma \in [2]$, define matrices $P^{(i, \gamma)}$ as follows. Define $M \times M$ matrices $P^{(i, 1)}, P^{(i, 2)}, Q^{(i)}$ as follows.

$$\begin{aligned} P^{(i, 1)} &:= (\mathbb{I} - |i\rangle \langle i|) \otimes \mathbb{I} \\ P^{(i, 2)} &:= (\mathbb{I} - 2|i\rangle \langle i|) \otimes \mathbb{I} \\ Q^{(i)} &:= |i\rangle \langle i| \otimes \mathbb{I}. \end{aligned}$$

Observe that $\|Q^{(i)}\|_{\text{op}}, \|P^{(i, 1)}\|, \|P^{(i, 2)}\|_{\text{op}} \leq 1$, furthermore,

$$Q^{(i)} = \mathbb{I} - P^{(i, 1)} = \frac{1}{2} (\mathbb{I} - P^{(i, 2)}). \quad (8)$$

Secondly, the $Q^{(i)}$ are orthogonal, i.e., for all $i, i' \in [N]$ we have

$$Q^{(i')} \cdot Q^{(i)\dagger} = Q^{(i)\dagger} \cdot Q^{(i')} = 0 \quad \text{if } i \neq i'. \quad (9)$$

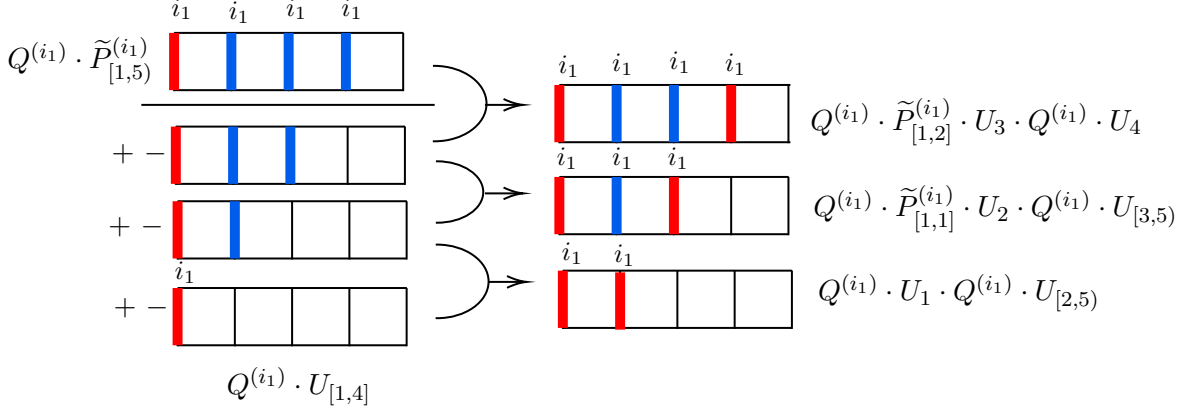


Figure 7: Our goal is to bound the operator norm of matrices like the one in the top left corner of this figure. We instead add and subtract the three matrices below. Using the fact that $Q^{(i)} = \gamma \cdot (\mathbb{I} - P^{(i,\gamma)})$ as in [Equation \(8\)](#), we see that the difference of every successive pair is given by the matrices on the right, up to a factor of γ .

For $\gamma \in [2]$, define $\tilde{P}_t^{(i,\gamma)} = U_t \cdot P^{(i,\gamma)}$ for all $i \in [N], t \in [t_1, d-1]$ and let $\tilde{P}_d^{(i,\gamma)} = U_d$. Observe that for all $\gamma \in [2]$,

$$U_\gamma = U_{[1,t_1]} \cdot \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{t_1}^{(i_{t_1},\gamma)} \dots \tilde{P}_d^{(i_{t_1},\gamma)} \triangleq U_{[1,t_1]} \cdot \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,d]}^{(i_{t_1},\gamma)}.$$

An example of this is depicted in [Figure 6](#). Fix any $t \in [t_1, d]$ and $\gamma \in [2]$. We have

$$\begin{aligned} \|U_\gamma\|_{\text{frob}}^2 &\triangleq \left\| U_{[1,t_1]} \cdot \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,d]}^{(i_{t_1},\gamma)} \right\|_{\text{frob}}^2 \leq \left\| \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,d]}^{(i_{t_1},\gamma)} \right\|_{\text{frob}}^2 \quad (\text{since } \|U_{[1,t_1]}\|_{\text{op}} \leq 1) \\ &= \sum_{i_{t_1} \in [N]} \left\| Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,d]}^{(i_{t_1},\gamma)} \right\|_{\text{frob}}^2 \quad (\text{by Equation (9)}) \\ &\leq \sum_{i_{t_1} \in [N]} \left\| Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,t]}^{(i_{t_1},\gamma)} \cdot U_t \right\|_{\text{frob}}^2 \quad (\text{by Fact 2.2 and since other terms have } \|\cdot\|_{\text{op}} \leq 1) \\ &= \left\| \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,t]}^{(i_{t_1},\gamma)} \cdot U_t \right\|_{\text{frob}}^2 \quad (\text{by Equation (9)}) \\ &\leq \|U_t\|_{\text{frob}}^2 \cdot \left\| \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,t]}^{(i_{t_1},\gamma)} \right\|_{\text{op}}^2 \quad (\text{by Fact 2.2}) \end{aligned}$$

Setting $t = t_1$ and recalling that $\sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} = \mathbb{I}$, we get $\|U_\gamma\|_{\text{frob}} \leq \|U_{t_1}\|_{\text{frob}} \leq \sqrt{M}$. We now show that $\left\| \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1,t]}^{i_{t_1}} \right\|_{\text{op}} \leq O(t)$. By the telescoping sum, we have

$$Q^{(i_1)} \cdot U_{[t_1,t]} - Q^{(i_1)} \cdot \tilde{P}_{[t_1,t]}^{(i_1,\gamma)} = \sum_{t' \in [t_1, t-1]} Q^{(i_1)} \cdot \tilde{P}_{[t_1,t']}^{(i_1,\gamma)} \cdot (U_{t'} - \tilde{P}_{t'}^{(i_1,\gamma)}) \cdot U_{(t',t)}$$

$$\begin{aligned}
&= \sum_{t' \in [t_1, t-1]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot \left(\mathbb{I} - \tilde{P}^{(i_1, \gamma)} \right) \cdot U_{(t', t)} \quad (\text{by definition}) \\
&= \sum_{t' \in [t_1, t-1]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot \gamma \cdot Q^{(i_1)} \cdot U_{(t', t)}. \quad (\text{by Equation (8)})
\end{aligned}$$

An example of this telescoping sum is depicted in **Figure 7**. Summing over $i_1 \in [N]$ and using the fact that $\sum_{i_1 \in [N]} Q^{(i_1)} = \mathbb{I}$, we see that

$$\sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t]}^{(i_1)} = U_{[t_1, t]} - \gamma \sum_{t' \in [t_1, t-1]} \sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot Q^{(i_1)} \cdot U_{(t', t)}.$$

Hence, using the fact that $\|U_{t'}\|_{\text{op}} \leq 1$ for all $t' \in [1, d]$, we get

$$\left\| \sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t]}^{(i_1, \gamma)} \right\|_{\text{op}} \leq 1 + (t-1) \cdot \gamma \cdot \max_{t' \in [t_1, t-1]} \left\| \sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot Q^{(i_1)} \right\|_{\text{op}}.$$

We observe that $\gamma \leq 2$ for all $\gamma \in [2]$. Finally, we observe that each of the terms inside the maximum is at most 1, due to the following:

$$\begin{aligned}
&\left(\sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot Q^{(i_1)} \right) \cdot \left(\sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot Q^{(i_1)} \right)^\dagger \\
&= \sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot Q^{(i_1)} \cdot Q^{(i_1)\dagger} \cdot U_{t'}^\dagger \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)\dagger} \cdot Q^{(i_1)\dagger} \quad (\text{by Equation (9)}) \\
&\preceq \sum_{i_1 \in [N]} Q^{(i_1)} \cdot Q^{(i_1)\dagger} = \mathbb{I} \quad (\text{since } \left\| \tilde{P}_{[1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot Q^{(i_1)} \right\|_{\text{op}} \leq 1)
\end{aligned}$$

The desired operator norm is thus at most $O(t)$, and we have argued that $\|U_\gamma\|_{\text{frob}} \leq O(d) \cdot \min_{t \in [t_1, d]} \|U_t\|_{\text{frob}}$. The same argument as above implies that

$$\|U_\gamma\|_{\text{op}} \leq \left\| \sum_{i \in [N]} Q^{(i)} \cdot \tilde{P}_{[t_1, d]}^{(i, \gamma)} \right\|_{\text{op}} \leq 1 + 2d \cdot \max_{t' \in [t_1, d]} \left\| \sum_{i_1 \in [N]} Q^{(i_1)} \cdot \tilde{P}_{[t_1, t']}^{(i_1, \gamma)} \cdot U_{t'} \cdot Q^{(i_1)} \right\|_{\text{op}} \leq O(d).$$

Finally, we see that

$$\begin{aligned}
\|U_\gamma\|_{\text{frob}}^2 &\triangleq \left\| U_{[1, t_1]} \cdot \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, d]}^{(i_{t_1}, \gamma)} \right\|_{\text{frob}}^2 \\
&\leq \|U_{[1, t_1]}\|_{\text{frob}}^2 \cdot \left\| \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, d]}^{(i_{t_1}, \gamma)} \right\|_{\text{op}}^2 \quad (\text{by Fact 2.2}) \\
&\leq \min_{t \in [1, t_1]} \|U_t\|_{\text{frob}}^2 \cdot O(d^2).
\end{aligned}$$

Altogether, we have $\|U_\gamma\|_{\text{frob}} \leq O(d) \cdot \min_{t \in [d]} \|U_t\|_{\text{frob}}$. This completes the proof. \square

3.1 Matrix Decomposition Lemma for DQC_1

Lemma 3.4. Let U_1, \dots, U_d be $M \times M$ matrices with $\|U_t\|_{\text{op}} \leq 1$ for $t \in [d]$. Let $t^* \in (1, d]$. Define a matrix U such that for all $I_1, I_{d+1} \in [M]$,

$$\tilde{U}[I_1|I_{d+1}] = \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \prod_{t \in [t^*, d]} \mathbb{1}[i_t \neq i_1, i_{d+1}].$$

Then, $\|\tilde{U}\|_{\text{frob}} \leq O(d) \cdot \|U_{t^*}\|_{\text{frob}}$.

Corollary 3.5. Under the same hypothesis as [Lemma 3.4](#), let $T \subseteq [d]$ and $L \subseteq [N]$. Define a matrix \tilde{U} such that for all $I_1, I_{d+1} \in [M], S_{d+1} \subseteq [N]$,

$$\tilde{U}[I_1|I_{d+1}S_{d+1}] = \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \mathbb{1} \left[S_{d+1} = \bigoplus_{\substack{t \in [1, d] \setminus T \\ i_t \in L}} \{i_t\} \right] \cdot \prod_{t \in [t^*, d]} \mathbb{1}[i_t \neq i_1, i_{d+1}].$$

Then, $\|\tilde{U}\|_{\text{frob}} \leq O(d) \cdot \|U_{t^*}\|_{\text{frob}}$.

The proof of [Corollary 3.5](#) from [Lemma 3.4](#) is identical to that of [Corollary 3.3](#) from [Lemma 3.2](#) and is omitted. The proof of [Lemma 3.4](#) is deferred to [Appendix A.1](#).

3.2 Matrix Decomposition Lemma for $\frac{1}{2}\text{BQP}$

Let $g : [N] \times [N] \rightarrow \mathbb{R}$ where $g(i, j) = (-1)^{\mathbb{1}[i=j]}$ for $i, j \in [N]$.

Lemma 3.6. Let U_1, \dots, U_d be $M \times M$ matrices with $\|U_t\|_{\text{op}} \leq 1$ for $t \in [d]$. Let $t_1 < t_2 \in [2, d]$. Define a matrix U such that for all $I_1, I_{d+1} \in [M]$,

$$U[I_1|I_{d+1}] = \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \prod_{t \in (t_1, d]} g(i_t, i_{t_1}) \cdot \prod_{t \in (t_2, d]} g(i_t, i_{t_2}).$$

Then, $\|U\|_{\text{frob}} \leq O(d^4) \cdot \min_{t \in [d]} (\|U_t\|_{\text{frob}})$.

Corollary 3.7. Under the same hypothesis as [Lemma 3.6](#), let $T \subseteq [d]$ and $L \subseteq [N]$. Define a matrix \tilde{U} such that for all $I_1, I_{d+1} \in [M], S_{d+1} \subseteq [N]$,

$$\begin{aligned} \tilde{U}[I_1|I_{d+1}S_{d+1}] &= \sum_{I_2, \dots, I_d \in [M]} \left(\prod_{t \in [1, d]} U_t[I_t|I_{t+1}] \right) \cdot \mathbb{1} \left[S_{d+1} = \bigoplus_{\substack{t \in [1, d] \setminus T \\ i_t \in L}} \{i_t\} \right] \\ &\cdot \prod_{t \in (t_1, d]} g(i_t, i_{t_1}) \cdot \prod_{t \in (t_2, d]} g(i_t, i_{t_2}). \end{aligned}$$

Then, $\|\tilde{U}\|_{\text{frob}} \leq O(d^4) \cdot \min_{t \in [d]} (\|U_t\|_{\text{frob}})$.

The proof of [Corollary 3.7](#) from [Lemma 3.6](#) is identical to that of [Corollary 3.3](#) from [Lemma 3.2](#) and is omitted. The proof of [Lemma 3.6](#) is deferred to [Appendix A.2](#).

4 Fourier Growth of DQC_k : Level-Two

In this section, we will show the $\ell = 2$ case of [Theorem 1.5](#), i.e., the level-two Fourier growth of a d -query DQC_k algorithm is at most $O(d^3\sqrt{K})$. Since DQC_k algorithms are a sub-class of BQP algorithms, the bounds from [Theorem 1.7](#) immediately apply to DQC_k algorithms and complete the proof when $\min(2^{k/2}, \sqrt{N}) = \sqrt{N}$. It suffices to handle the other case, i.e., $\min(2^{k/2}, \sqrt{N}) = 2^{k/2}$ which will be the focus of this section.

Throughout this section, to simplify notation, we use the shorthand I_t to denote (i_t, w_t, k_t) where $i_t \in [N], w_t \in [W], k_t \in [K]$ for $N = 2^n, W = 2^w, K = 2^k$. We use \mathbb{I} to denote the identity matrix, where the dimension is implicit.

Let $f(x)$ be the acceptance probability of a d -query DQC_k algorithm and ρ be any restriction of the input variables. We will now derive an expression for the Fourier coefficients of $f|_\rho(x)$. Let ρ keep the variables in $L \subseteq [N]$ alive and fix the rest. Thus, only Fourier coefficients corresponding to $S \subseteq [L]$ are non-zero and are described by the following claim.

Claim 4.1. *Let $f(x)$ be the acceptance probability of a d -query DQC_k algorithm and let $\rho \in \{-1, 1, *\}^N$ be any restriction that leaves $L \subseteq [N]$ unfixed and fixes the rest. Then, there exist matrices $V_1^\rho, \dots, V_{2d}^\rho$ such that for all $S \subseteq [L]$,*

$$\widehat{f|_\rho}(S) = (NW)^{-1} \sum_{I_1, \dots, I_{2d} \in [M]} \left(\prod_{t \in [2d]} V_t^\rho[I_t | I_{t+1}] \right) \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in [2d] \\ \text{with } i_t \in L}} \{i_t\} = S \right].$$

where $V_1^\rho, \dots, V_{2d}^\rho \in \mathbb{C}^{M \times M}$ satisfy $\|V_t^\rho\|_{\text{op}} \leq 1$ for $t \in [2d]$ and $\|V_1^\rho\|_{\text{frob}} \leq \sqrt{M/K}$

The proof of this is fairly simple and is deferred to [Appendix A.6](#). We will now establish $L_{1,2}$ bounds for DQC_k algorithms. The goal of this section is to upper bound

$$L_{1,2}(f|_\rho) \triangleq \max_{\alpha \in [-1, 1]^{\binom{N}{2}}} L_{1,2}^\alpha(f|_\rho) = \max_{\alpha \in [-1, 1]^{\binom{N}{2}}} \sum_{S \in \binom{[N]}{2}} \alpha_S \cdot \widehat{f|_\rho}(S). \quad (10)$$

Fix any $\alpha_S \in [-1, 1]$ for each $S \in \binom{[L]}{2}$. From [Equation \(10\)](#) and [Claim 4.1](#), we see that our goal is to upper bound

$$L_{1,2}^\alpha(f|_\rho) = \sum_{\substack{S \subseteq L \\ |S|=2}} (NW)^{-1} \sum_{I_1, \dots, I_{2d} \in [M]} \left(\prod_{t \in [2d]} V_t^\rho[I_t | I_{t+1}] \right) \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in [2d] \\ \text{with } i_t \in L}} \{i_t\} = S \right] \cdot \alpha_S \quad (11)$$

Observe that if $\bigoplus_{t \in [2d], i_t \in L} \{i_t\}$ has size 2, then there must exist a subset $T = \{t_1, t_2\} \subseteq [2d]$ for $t_1 \neq t_2$ such that $\{i_{t_1}, i_{t_2}\}$ has two distinct elements in L and $\bigoplus_{\substack{t \in [2d] \setminus T \\ \text{with } i_t \in L}} \{i_t\} = \emptyset$. For any fixed S , there may be many such T , but to uniquely identify one, we define T to correspond to the first times that variables in S appear. More precisely, let t_1 be the first time t for which $i_t \in S$, and let t_2 be the first time $t > t_1$ for which $i_t \in S \setminus \{i_{t_1}\}$. This says that i_{t_1} is the first element of S to appear, i_{t_2} is the next unseen element of S to appear. Conversely, for any T and $\{i_t\}_{t \in T}$ satisfying the above conditions, it defines a unique $S = \{i_t : t \in T\}$.

Fix $T \subseteq [2d]$ of size 2 (this can be done in $\binom{2d}{2}$ ways) and let $T = \{t_1, t_2\}$ for $t_1 < t_2$. Define

$$\begin{aligned} \Delta_T := & \sum_{I_1, \dots, I_{2d} \in [M]} \left(\prod_{t \in [2d]} V_t^\rho[I_t | I_{t+1}] \right) \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in [2d] \setminus T \\ \text{with } i_t \in L}} \{i_t\} = \emptyset \right] \\ & \cdot \prod_{t \in [1, t_1)} \mathbb{1}[i_t \neq i_{t_1}] \cdot \prod_{t \in [1, t_2)} \mathbb{1}[i_t \neq i_{t_2}] \cdot \mathbb{1}[i_{t_1} \neq i_{t_2} \in L] \cdot \alpha_{\{i_{t_1}, i_{t_2}\}}. \end{aligned} \quad (12)$$

From the above paragraph, it follows that

$$L_{1,2}^\alpha(f|\rho) = (NW)^{-1} \sum_{T \in \binom{[2d]}{2}} \Delta_T \leq \binom{2d}{2} \cdot (NW)^{-1} \cdot \max_{T \in \binom{[2d]}{2}} |\Delta_T|.$$

We will now show that for all $T \in \binom{[2d]}{2}$, we have $|\Delta_T| \leq O(d) \cdot M \cdot K^{-1/2}$. This, along with the above equation (and the fact that $M = KNW$) would imply that $L_{1,2}^\alpha(f|\rho) \leq \sqrt{K} \cdot O(d^3)$ as desired. We now show the desired bound of $\Delta_T \leq O(d) \cdot M \cdot K^{-1/2}$.

We will group the terms $t \in [2d]$ into circular intervals $[t_1, t_2), [t_2, t_1)$. Define matrices $\tilde{V}_{[t_1, t_2)}$ and $\tilde{V}_{[t_2, t_1)}$ such that for all $I_{t_1}, I_{t_2} \in [M], S_{t_1} \subseteq [N]$, we have

$$\begin{aligned} \tilde{V}_{[t_1, t_2)}[I_{t_1} S_{t_1} | I_{t_2}] = & \sum_{I_{t_1+1}, \dots, I_{t_2-1} \in [M]} \left(\prod_{t \in [t_1, t_2)} V_t^\rho[I_t | I_{t+1}] \right) \cdot \mathbb{1} \left[S_{t_1} = \bigoplus_{\substack{t \in (t_1, t_2) \\ i_t \in L}} \{i_t\} \right] \\ & \cdot \prod_{t \in (t_1, t_2)} \mathbb{1}[i_t \neq i_{t_2}], \text{ and} \end{aligned} \quad (13)$$

$$\begin{aligned} \tilde{V}_{[t_2, t_1)}[I_{t_2} | I_{t_1} S_{t_1}] = & \sum_{I_{t_2+1}, \dots, I_{t_1-1} \in [M]} \left(\prod_{t \in [t_2, t_1)} V_t^\rho[I_t | I_{t+1}] \right) \cdot \mathbb{1} \left[S_{t_1} = \bigoplus_{\substack{t \in (t_2, t_1) \\ i_t \in L}} \{i_t\} \right] \\ & \cdot \prod_{t \in [1, t_1)} \mathbb{1}[i_t \neq i_{t_2}, i_{t_1}], \end{aligned} \quad (14)$$

Define $\alpha'_{\{i_{t_1}, i_{t_2}\}} = \alpha_{\{i_{t_1}, i_{t_2}\}} \cdot \mathbb{1}[i_{t_1}, i_{t_2} \in L]$ for all $i_{t_1}, i_{t_2} \in [N]$. Combining [Equation \(12\)](#) with [Equations \(13\)](#) and [\(14\)](#), and using the fact that

$$\prod_{t \in [1, t_1)} \mathbb{1}[i_t \neq i_{t_1}] \cdot \prod_{t \in [1, t_2)} \mathbb{1}[i_t \neq i_{t_2}] = \mathbb{1}[i_{t_1} \neq i_{t_2}] \cdot \prod_{t \in [1, t_1)} \mathbb{1}[i_t \neq i_{t_2}, i_{t_1}] \cdot \prod_{t \in (t_1, t_2)} \mathbb{1}[i_t \neq i_{t_2}]$$

we obtain that

$$\begin{aligned} \Delta_T = & \sum_{\substack{I_{t_1}, I_{t_2} \in [M] \\ S_{t_2} \subseteq [N]}} \tilde{V}_{[t_1, t_2)}[I_{t_1} S_{t_1} | I_{t_2}] \cdot \tilde{V}_{[t_2, t_1)}[I_{t_2} | I_{t_1} S_{t_1}] \cdot \alpha'_{\{i_{t_1}, i_{t_2}\}} \cdot \mathbb{1}[i_{t_1} \neq i_{t_2}] \\ |\Delta_T| \leq & \|\tilde{V}_{[t_1, t_2)}\|_{\text{frob}} \cdot \|\tilde{V}_{[t_2, t_1)}\|_{\text{frob}} \end{aligned} \quad (\text{by [Fact 2.3](#)})$$

It suffices to upper bound $\|\tilde{V}_{[t_1, t_2)}\|_{\text{frob}} \cdot \|\tilde{V}_{[t_2, t_1)}\|_{\text{frob}}$. We consider two cases:

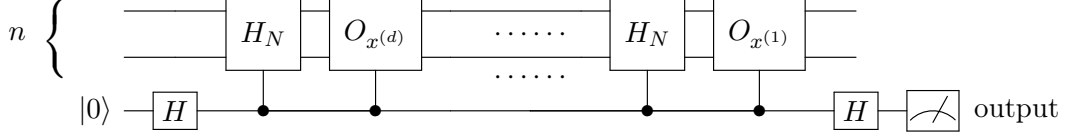


Figure 8: A d -query DQC_1 algorithm with n maximally mixed qubits.

Case 1: $t_1 = 1$. In this case, observe that $1 \in [t_1, t_2]$. We observe that the matrix $\tilde{V}_{[t_1, t_2]}$ in Equation (13) is obtained by applying Corollary 3.3 on the matrices $V_{t_1}^\rho, \dots, V_{t_2-1}^\rho$ (with omitted set T) in reverse order with transposes and hence, $\|\tilde{V}_{[t_1, t_2]}\|_{\text{frob}} \leq O(d) \cdot \|V_1^\rho\|_{\text{frob}} \leq O(d) \cdot \sqrt{M/K}$. Since $[1, t_1] = \emptyset$, the matrix $\tilde{V}_{[t_2, t_1]}$ is obtained by applying Lemma 3.1 on matrices $V_{t_2}^\rho, \dots, V_{t_1-1}^\rho$ (with omitted set T) in this order and hence, $\|\tilde{V}_{[t_2, t_1]}\|_{\text{frob}} \leq \sqrt{M}$. In this case,

$$\|\tilde{V}_{[t_1, t_2]}\|_{\text{frob}} \cdot \|\tilde{V}_{[t_2, t_1]}\|_{\text{frob}} \leq O(d) \cdot \sqrt{M/K} \cdot \sqrt{M} \leq O(d) \cdot M/\sqrt{K}.$$

Case 2: $t_1 > 1$. In this case, observe that $1 \in (t_2, t_1)$. Observe that the matrix $\tilde{V}_{[t_2, t_1]}$ in Equation (14) is obtained by applying Corollary 3.5 on matrices $V_{t_2}^\rho, \dots, V_{t_1-1}^\rho$ (with omitted set T and $t^* = 1$, which is well-defined) in forward order, and hence, $\|\tilde{V}_{[t_2, t_1]}\|_{\text{frob}} \leq O(d) \cdot \|V_1^\rho\|_{\text{frob}} \leq O(d) \cdot \sqrt{M/K}$. As before, the matrix $\tilde{V}_{[t_1, t_2]}$ in Equation (13) is obtained by applying Corollary 3.3 on the matrices $V_{t_1}^\rho, \dots, V_{t_2-1}^\rho$ (with omitted set T) in reverse order with transposes and hence, $\|\tilde{V}_{[t_1, t_2]}\|_{\text{frob}} \leq \sqrt{M}$. In this case,

$$\|\tilde{V}_{[t_1, t_2]}\|_{\text{frob}} \cdot \|\tilde{V}_{[t_2, t_1]}\|_{\text{frob}} \leq O(d) \cdot \sqrt{M} \cdot \sqrt{M/K} \leq O(d) \cdot M/\sqrt{K}.$$

4.1 Tightness of our Bounds for DQC_1

In this section, we will show that the dependence on k and N is tight in Theorem 1.5.

Dependence on N . First, we consider the case $k = 1$ and show that DQC_1 algorithms can indeed achieve level- ℓ Fourier growth of roughly $N^{(\ell-2)/2}$, i.e., the dependence on N is tight in Theorem 1.5. We will do so by producing an algorithm on inputs of length dN which makes d oracle queries and whose level- ℓ Fourier growth for $\ell = d$ is $\Omega(N^{(\ell-2)/2})$.

Let H_N be the Hadamard matrix as in Definition 2.1 and view this matrix as an n -qubit unitary operator. For $t \in [d]$, let N_t denote the interval $((t-1)N, tN]$ so that $N_1 \sqcup \dots \sqcup N_d = [dN]$. We view the input $x \in \{-1, 1\}^{N^d}$ as comprising of d input strings $x^{(1)}, \dots, x^{(d)}$ of length N each such that $x^{(t)}$ is supported on N_t . Instead of the oracle O_x , we will consider d oracles $O_{x^{(1)}}, \dots, O_{x^{(d)}}$. Consider the d -query DQC_1 algorithm as in Figure 8.⁸ As we saw in Equation (1), it is not too difficult to show the bias of this algorithm is precisely

$$f(x) = \frac{1}{2^N} \text{Tr}(O_{x^{(1)}} \cdot H_N \cdots O_{x^{(d)}} \cdot H_N).$$

⁸Typically, we express DQC_1 in terms of a single oracle O_x , as opposed to d smaller oracles $O_{x^{(1)}}, \dots, O_{x^{(d)}}$, nevertheless, it is easy to embed the circuit in Figure 8 into a larger one consisting only of O_x oracle calls for $x = (x^{(1)}, \dots, x^{(d)})$ by applying the following sequence of operators d times: $H_N \otimes \mathbb{I}$, followed O_x , followed by the permutation matrix Π that maps $|i\rangle \rightarrow |i - N \pmod{Nd}\rangle$ for all computational basis states $i \in [Nd]$.

We observe the Fourier coefficients of f correspond to subsets $S \subseteq [Nd]$ that pick exactly one element from each N_t . There are N^d such non-zero Fourier coefficients and they are given by

$$\widehat{f}(S) = \frac{1}{2N} \sum_{\substack{i_t \in N_t \\ \text{for } t \in [d]}} (-1)^{\langle i_1, i_2 \rangle + \dots + \langle i_d, i_1 \rangle} \cdot \frac{1}{N^{d/2}} \cdot \mathbb{1}[S = \{i_1, \dots, i_d\}].$$

Each such S uniquely identifies $i_1 \in N_1, \dots, i_d \in N_d$ and we set $\alpha_S := (-1)^{\langle i_1, i_2 \rangle + \dots + \langle i_d, i_1 \rangle}$. Thus, we obtain that the level- d Fourier growth is at least

$$N^d \cdot \frac{1}{2N} \cdot \frac{1}{N^{d/2}} \geq \Omega\left(N^{(d-2)/2}\right).$$

This completes the proof.

Dependence on k . It is clear to see that a DQC_k algorithm can solve the Forrelation problem on inputs of length 2^k , since we can run the k -qubit Forrelation circuit on the clean qubits. As the Forrelation function on 2^k -bit inputs has level-two Fourier growth of $2^{k/2}$, this saturates the bound from [Theorem 1.5](#) for level two.

5 Fourier Growth of $\frac{1}{2}$ BQP: Level-Three

In this section, we will show the $\ell = 3$ case of [Theorem 1.6](#), i.e., the level-three Fourier growth of a d -query $\frac{1}{2}$ BQP algorithm is at most $O(d^7 \sqrt{N})$. Throughout this section, to simplify notation, we use the shorthand I_t to denote (i_t, w_t) where $i \in [N], w \in [W]$ for $N = 2^n, W = 2^w$. We use \mathbb{I} to denote the identity matrix, where the dimension is implicit.

Given the expression for the acceptance probability of a d -query $\frac{1}{2}$ BQP algorithm ([Claim 2.12](#)), it is not too difficult to derive an expression for the Fourier coefficients under any restriction – this part is similar to the proof of [Claim 4.1](#) from [Claim 2.10](#). We obtain the following claim, whose proof is deferred to [Appendix A.6](#).

Claim 5.1. *Let $f(x)$ be the acceptance probability of a d -query $\frac{1}{2}$ BQP algorithm and $\rho \in \{-1, 1, *\}^N$ be any restriction that leaves the coordinates in $L \subseteq [N]$ free and fixes the rest. Then, there exist matrices $V_1^\rho, \dots, V_{2d+2}^\rho \in \mathbb{C}^{M \times M}$ such that for all $S \subseteq L$,*

$$\widehat{f|_\rho}(S) = M^{-1} \sum_{I_1, \dots, I_{2d+2} \in [M]} F_{I_1, I_{d+2}} \cdot \prod_{t \in [2d+2]} V_t^\rho[I_t | I_{t+1}] \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in [2d+2] \setminus \{1, d+2\} \\ i_t \in L}} \{i_t\} = S \right] \quad (15)$$

where $\|V_t^\rho\|_{\text{op}} \leq 1$ for all $t \in [2d+2]$.

Now that we have an expression for the Fourier coefficients, we turn our attention to proving Fourier growth bounds. Let $\tilde{A} = A \cap L, \tilde{B} = B \cap L, \tilde{C} = C \cap L$, where A, B, C are as in [Definition 2.5](#). Let $D = [2d+2] \setminus \{1, d+2\}$. We use I to denote I_1, \dots, I_{2d+2} . As mentioned before, we will only be able to bound $L_{1,3}^{\alpha(\gamma)}(f|_\rho)$ where $\gamma \in [-1, 1]^{3N}$ and $\alpha(\gamma)$ is as in [Definition 2.5](#). Fix any such $\alpha(\gamma)$. Recalling [Definition 2.5](#), for $\alpha(\gamma)_S$ to be non-zero, we must have $S = \{a, b, c\}$ where $a \in \tilde{A}, b \in \tilde{B}, c \in \tilde{C}$. Fix any such S . For $\bigoplus_{\substack{t \in D \\ i_t \in L}} \{i_t\} = S$, there exist distinct $t_1, t_2, t_3 \in D$ such that

$$i_{t_1} = a, i_{t_2} = b, i_{t_3} = c \text{ and } \bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ i_t \in L}} \{i_t\} = \emptyset,$$

however, these t_1, t_2, t_3 may not be unique. Instead of defining them uniquely, we add each of them but with a multiplier function that cancels out redundancies.

Let $g(i_1, i_2) = (-1)^{\mathbb{1}[i_1=i_2]}$ for $i_1, i_2 \in [N]$ be the function as before. For $q \in [3]$, let

$$r_{t_q}[I] := (-1)^{\#\{t < t_q, t \in D | i_t = i_{t_q}\}} = \prod_{t < t_q, t \in D} g(i_t, i_{t_q}). \quad (16)$$

$$s_{t_q}[I] := (-1)^{\#\{t > t_q, t \in D | i_t = i_{t_q}\}} = \prod_{t > t_q, t \in D} g(i_t, i_{t_q}). \quad (17)$$

Claim 5.2. Consider $\Phi(I)$ defined as follows.

$$\Phi(I) := \sum_{\substack{t_1, t_2, t_3 \in D \\ \text{distinct}}} r_{t_1}[I] \cdot \mathbb{1}[i_{t_1} = a] \cdot r_{t_2}[I] \cdot \mathbb{1}[i_{t_2} = b] \cdot r_{t_3}[I] \cdot \mathbb{1}[i_{t_3} = c]$$

Whenever $\bigoplus_{\substack{t \in D \\ i_t \in L}} \{i_t\} = \{a, b, c\}$, we have $\Phi(I) = 1$.

Proof of Claim 5.2. Consider

$$\begin{aligned} \Phi(I) &:= \sum_{\substack{t_1, t_2, t_3 \in D \\ \text{distinct}}} r_{t_1}[I] \cdot \mathbb{1}[i_{t_1} = a] \cdot r_{t_2}[I] \cdot \mathbb{1}[i_{t_2} = b] \cdot r_{t_3}[I] \cdot \mathbb{1}[i_{t_3} = c] \\ &= \sum_{\substack{t_2, t_3 \in D \\ \text{distinct}}} r_{t_2}[I] \cdot \mathbb{1}[i_{t_2} = b] \cdot r_{t_3}[I] \cdot \mathbb{1}[i_{t_3} = c] \cdot \sum_{\substack{t_1 \in D \\ t_1 \neq t_2, t_3}} r_{t_1}[I] \cdot \mathbb{1}[i_{t_1} = a] \\ &= \sum_{\substack{t_2, t_3 \in D \\ \text{distinct}}} r_{t_2}[I] \cdot \mathbb{1}[i_{t_2} = b] \cdot r_{t_3}[I] \cdot \mathbb{1}[i_{t_3} = c] \cdot 1 \end{aligned}$$

where the last equality follows because for any fixed t_2, t_3 , we have $i_{t_1} = a$ for an odd number of t_1 (because of the condition $\bigoplus_{\substack{t \in D \\ i_t \in L}} \{i_t\} = \{a, b, c\}$) and any such t_1 is distinct from t_2, t_3 (since $b, c \neq a$) and as t_1 varies over these possible times, the quantity $r_{t_1}[I]$ alternates between 1 and -1 (at the first time it is $+1$, the second time it is -1 , and so on). We can now repeat the same argument to obtain

$$\Phi(I) := \sum_{t_2 \in D} r_{t_2}[I] \cdot \mathbb{1}[i_{t_2} = b] \cdot \left(\sum_{\substack{t_3 \in D \\ t_3 \neq t_2}} r_{t_3}[I] \cdot \mathbb{1}[i_{t_3} = c] \right) = \sum_{t_2 \in D} r_{t_2}[I] \cdot \mathbb{1}[i_{t_2} = b] = 1.$$

□

Therefore, for any $a \in \tilde{A}, b \in \tilde{B}, c \in \tilde{C}$ we have

$$\begin{aligned} \mathbb{1} \left[\bigoplus_{\substack{t \in D \\ i_t \in L}} \{i_t\} = \{a, b, c\} \right] &= \mathbb{1} \left[\bigoplus_{\substack{t \in D \\ i_t \in L}} \{i_t\} = \{a, b, c\} \right] \cdot \Phi(I) \\ &= \sum_{\substack{t_1, t_2, t_3 \in D \\ \text{distinct}}} \mathbb{1} \left[\bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ i_t \in L}} \{i_t\} = \emptyset \right] \end{aligned}$$

$$\cdot \mathbb{1}[i_{t_1} = a] \cdot \mathbb{1}[i_{t_2} = b] \cdot \mathbb{1}[i_{t_3} = c] \cdot r_{t_1}(I) \cdot r_{t_2}(I) \cdot r_{t_3}(I)$$

Now, we multiply both sides by $\alpha(\gamma)_{a,b,c}$ and sum over all possible $a \in \tilde{A}, b \in \tilde{B}, c \in \tilde{C}$ to obtain

$$\begin{aligned} L_{1,3}^{\alpha(\gamma)}[f|_\rho] &\triangleq \sum_{a \in \tilde{A}, b \in \tilde{B}, c \in \tilde{C}} \alpha(\gamma)_{a,b,c} \cdot \widehat{f|_\rho}(\{a, b, c\}) \\ &= M^{-1} \sum_{\substack{a \in \tilde{A}, b \in \tilde{B}, c \in \tilde{C} \\ I_1, \dots, I_{2d+2} \in [M]}} F_{I_1, I_{d+2}} \cdot \prod_{t \in [2d+2]} V_t^\rho[I_t | I_{t+1}] \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in D \\ i_t \in L}} \{i_t\} = \{a, b, c\} \right] \cdot \alpha(\gamma)_{a,b,c} \\ &\quad \text{(by Claim 5.1)} \\ &= M^{-1} \sum_{\substack{a \in \tilde{A}, b \in \tilde{B}, c \in \tilde{C} \\ I_1, \dots, I_{2d+2} \in [M]}} F_{I_1, I_{d+2}} \cdot \prod_{t \in [2d+2]} V_t^\rho[I_t | I_{t+1}] \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ i_t \in L}} \{i_t\} = \emptyset \right] \\ &\quad \cdot \sum_{\substack{t_1, t_2, t_3 \in D \\ \text{distinct}}} \mathbb{1}[i_{t_1} = a] \cdot \mathbb{1}[i_{t_2} = b] \cdot \mathbb{1}[i_{t_3} = c] \cdot r_{t_1}(I) \cdot r_{t_2}(I) \cdot r_{t_3}(I) \cdot \alpha(\gamma)_{i_{t_1}, i_{t_2}, i_{t_3}} \\ &= M^{-1} \sum_{\substack{I_1, \dots, I_{2d+2} \in [M] \\ t_1, t_2, t_3 \in D \text{ distinct}}} F_{I_1, I_{d+2}} \cdot \prod_{t \in [2d+2]} V_t^\rho[I_t | I_{t+1}] \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ i_t \in L}} \{i_t\} = \emptyset \right] \\ &\quad \cdot \mathbb{1}[i_{t_1} \in \tilde{A}] \cdot \mathbb{1}[i_{t_2} \in \tilde{B}] \cdot \mathbb{1}[i_{t_3} \in \tilde{C}] \cdot r_{t_1}(I) \cdot r_{t_2}(I) \cdot r_{t_3}(I) \cdot \alpha(\gamma)_{i_{t_1}, i_{t_2}, i_{t_3}} \end{aligned}$$

where the last line follows from the fact that $\sum_{a \in \tilde{A}} \mathbb{1}[i_{t_1} = a] = \mathbb{1}[i_{t_1} \in \tilde{A}]$. Fix any $i_{t_2}^* \in \tilde{B}$ and distinct $t_1, t_2, t_3 \in D$. (There are at most N possibilities for such $i_{t_2}^*$ and $O(d^3)$ possibilities for (t_1, t_2, t_3) .) Define

$$\begin{aligned} \Delta_{t_1, t_2, t_3, i_{t_2}^*}^\gamma &:= \sum_{I_1, \dots, I_{2d+2} \in [M]} F_{I_1, I_{d+2}} \cdot \prod_{t \in [2d+2]} V_t^\rho[I_t | I_{t+1}] \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ \text{with } i_t \in L}} \{i_t\} = \emptyset \right] \\ &\quad \cdot \mathbb{1}[i_{t_1} \in \tilde{A}] \cdot \mathbb{1}[i_{t_2} = i_{t_2}^*] \cdot \mathbb{1}[i_{t_3} \in \tilde{C}] \cdot r_{t_1}(I) \cdot r_{t_2}(I) \cdot r_{t_3}(I) \cdot \alpha(\gamma)_{i_{t_1}, i_{t_2}^*, i_{t_3}} \end{aligned} \quad (18)$$

Substituting this in the expression for the Fourier growth, we have

$$\begin{aligned} L_{1,3}^{\alpha(\gamma)} &= M^{-1} \sum_{\substack{\text{distinct } t_1, t_2, t_3 \\ \text{in } [2d+2] \setminus \{1, d+2\}, \\ i_{t_2}^* \in \tilde{B}}} \Delta_{t_1, t_2, t_3, i_{t_2}^*}^\gamma \quad \text{(from Equation (18) and Claim 5.1)} \\ &\leq M^{-1} \cdot O(d^3 N) \cdot \max_{\substack{\text{distinct } t_1, t_2, t_3 \\ \text{in } [2d+2] \setminus \{1, d+2\}, \\ i_{t_2}^* \in \tilde{B}}} \left| \Delta_{t_1, t_2, t_3, i_{t_2}^*}^\gamma \right|. \end{aligned}$$

We will show that for each distinct $t_1, t_2, t_3 \in [2d+2] \setminus \{1, d+2\}$ and $i_{t_2}^* \in \tilde{B}$, we have $\left| \Delta_{t_1, t_2, t_3, i_{t_2}^*}^\gamma \right| \leq O(d^4) \cdot \sqrt{MW}$. Substituting this above, we would get

$$L_{1,3}^{\alpha(\gamma)}(f|_\rho) \leq M^{-1} \cdot O(d^7 N) \cdot \sqrt{MW} = O(d^7) \cdot \sqrt{N} \cdot \sqrt{NMW} \cdot M^{-1} \leq O(d^7) \cdot \sqrt{N},$$

where we used the fact that $M = NW$. It now suffices prove the bound $\Delta_{t_1, t_2, t_3, i_{t_2}^*}^\gamma \leq O(d^4) \cdot \sqrt{MW}$.

Fix any distinct $t_1, t_2, t_3 \in D$ and $i_{t_2}^* \in \tilde{B}$. We will now use [Definition 2.5](#) to get

$$\alpha(\gamma)_{i_{t_1}, i_{t_2}, i_{t_3}} = \overline{H}(i_{t_1}, i_{t_2}^*) \cdot \overline{H}(i_{t_3}, i_{t_2}^*) \cdot \gamma_{i_{t_1}} \cdot \gamma_{i_{t_2}^*} \cdot \gamma_{i_{t_3}}$$

We will use this to encode the action of multiplication by $\alpha(\gamma)_{i_{t_1}, i_{t_2}, i_{t_3}}$ using a matrix product with diagonal matrices. Define matrices P_t as follows. Let $\Gamma_{t_1}, \Gamma_{t_2}, \Gamma_{t_3}$ be $M \times M$ diagonal matrices with $[-1, 1]$ -valued entries defined as follows. For $I_{t_1}, I_{t_2}, I_{t_3} \in [M]$, let

$$\Gamma_{t_1}[I_{t_1}|I_{t_1}] = \begin{cases} \gamma_{i_{t_1}} \cdot \overline{H}(i_{t_1}, i_{t_2}^*) & \text{if } i_{t_1} \in \tilde{A} \\ 0 & \text{otherwise.} \end{cases} \quad (19)$$

$$\Gamma_{t_2}[I_{t_2}|I_{t_2}] = \begin{cases} \gamma_{i_{t_2}^*} & \text{if } i_{t_2} = i_{t_2}^* \\ 0 & \text{otherwise.} \end{cases} \quad (20)$$

$$\Gamma_{t_3}[I_{t_3}|I_{t_3}] = \begin{cases} \gamma_{i_{t_3}} \cdot \overline{H}(i_{t_3}, i_{t_2}^*) & \text{if } i_{t_3} \in \tilde{C} \\ 0 & \text{otherwise.} \end{cases} \quad (21)$$

Let $P'_{t_1} := \Gamma_{t_1} \cdot V_{t_1}^\rho$, $P'_{t_2} := \Gamma_{t_2} \cdot V_{t_2}^\rho$, and $P'_{t_3} := \Gamma_{t_3} \cdot V_{t_3}^\rho$ and for all $t \neq t_1, t_2, t_3$, set $P'_t = V_t^\rho$. Finally, define a diagonal matrix $D^{(i_{t_2}^*)} = (\mathbb{I} - 2|i_{t_2}^*\rangle\langle i_{t_2}^*|) \otimes \mathbb{I}$, or equivalently, for all $I \in [M]$, let

$$D^{(i_{t_2}^*)}[I, I] := \begin{cases} -1 & \text{if } i = i_{t_2}^* \\ 1 & \text{otherwise.} \end{cases}$$

For $t \in D, t < t_2$, set $P_t = D^{(i_{t_2}^*)} \cdot P'_t$ and for all other t , set $P_t = P'_t$. Firstly, observe that

$$\|P_t\|_{\text{op}} \leq 1 \text{ for all } t \in [d], \quad (22)$$

since $\gamma \in [-1, 1]^{3N}$ and $\|D^{(i_{t_2}^*)}\|_{\text{op}}, \|\Gamma_t\|_{\text{op}}, \|V_t^\rho\|_{\text{op}} \leq 1$. Secondly, observe that

$$\|P_{t_2}\|_{\text{frob}} \leq \sqrt{W}, \quad (23)$$

since $P_{t_2} = \Gamma_{t_2} \cdot V_{t_2}^\rho$ and multiplying by the matrix Γ_{t_2} has the effect of zeroing out all but W rows (only the rows indexed by $i_{t_2}^*$ survive), and each row of $V_{t_2}^\rho$ has norm at most one. Furthermore, recalling the definition of $r_{t_2}(I)$ in [Equation \(16\)](#), we see that multiplication by $D^{(i_{t_2}^*)}$ in the appropriate locations has the effect of multiplying by $r_{t_2}(I)$. This construction allows us to simplify [Equation \(18\)](#) as

$$\Delta_{t_1, t_2, t_3, i_{t_2}^*}^\gamma = \sum_{I_1, \dots, I_{2d+2} \in [M]} F_{I_1, I_{d+2}} \cdot \prod_{t \in [2d+2]} P_t[I_t|I_{t+1}] \cdot \mathbb{1} \left[\bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ \text{with } i_t \in L}} \{i_t\} = \emptyset \right] \cdot r_{t_1}(I) \cdot r_{t_3}(I). \quad (24)$$

Because we are imposing the constraint $\bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ \text{with } i_t \in L}} \{i_t\} = \emptyset$, recalling [Equations \(16\)](#) and [\(17\)](#), we must have

$$r_{t_1}(I) = s_{t_1}(I) \quad \text{and} \quad r_{t_3}(I) = s_{t_3}(I).$$

This is because for all $q \in \{1, 3\}$, the number of times i_{t_q} occurs prior to t_q plus the number of times i_{t_q} occurs after t_q must be even, otherwise, the set $\bigoplus_{\substack{t \in D \setminus \{t_1, t_2, t_3\} \\ \text{with } i_t \in L}} \{i_t\}$ would contain i_{t_q} .

Let $D_L = D \cap [1, d+1]$ and $D_R = D \cap [d+2, 2d+2]$. For $q \in \{1, 3\}$, let $\eta(q) = D_L$ if $t_q \in D_L$ and $\eta(q) = D_R$ if $t_q \in D_R$. Define matrices $\tilde{P}_{[1, d+1]}$ and $\tilde{P}_{[d+2, 2d+2]}$ such that for all $I_1, I_{d+2} \in [M], S_{d+2} \subseteq [N]$, we have

$$\begin{aligned} \tilde{P}_{[1, d+1]}[I_1 S_1 | I_{d+2}] &= \sum_{I_2, \dots, I_{d+1} \in [M]} \left(\prod_{t \in [1, d+1]} P_t[I_t | I_{t+1}] \right) \cdot \mathbb{1} \left[S_1 = \bigoplus_{\substack{t \in D_L \setminus \{t_1, t_2, t_3\} \\ \text{with } i_t \in L}} \{i_t\} \right] \\ &\cdot \prod_{q \in \{1, 3\}: \eta(q) = D_L} r_{t_q}(I) \end{aligned} \quad (25)$$

$$\begin{aligned} \tilde{P}_{[d+2, 2d+2]}[I_{d+2} | I_1 S_1] &= \sum_{I_{d+3}, \dots, I_{2d+2} \in [M]} \left(\prod_{t \in [d+2, 2d+2]} P_t[I_t | I_{t+1}] \right) \cdot \mathbb{1} \left[S_1 = \bigoplus_{\substack{t \in D_R \setminus \{t_1, t_2, t_3\} \\ \text{with } i_t \in L}} \{i_t\} \right] \\ &\cdot \prod_{q \in \{1, 3\}: \eta(q) = D_R} s_{t_q}(I) \end{aligned} \quad (26)$$

Recall that $D_L \cup D_R = D$ and $[1, d+1] \cup [d+2, 2d+2] = [1, 2d+2]$. Plugging in [Equations \(25\)](#) and [\(26\)](#) into [Equation \(24\)](#), we have

$$\begin{aligned} \Delta_{t_1, t_2, t_3, i_{t_2}}^\gamma &= \sum_{\substack{I_1, I_{d+2} \in [M] \\ S_{d+2} \subseteq [N]}} F_{I_1, I_{d+2}} \cdot \tilde{P}_{[1, d+1]}[I_1 S_1 | I_{d+2}] \cdot \tilde{P}_{[d+2, 2d+2]}[I_{d+2} | I_1 S_1] \\ &\leq \sum_{\substack{I_1, I_{d+2} \in [M] \\ S_{d+2} \subseteq [N]}} \left| \tilde{P}_{[1, d+1]}[I_1 S_1 | I_{d+2}] \right| \cdot \left| \tilde{P}_{[d+2, 2d+2]}[I_{d+2} | I_1 S_1] \right| \quad (\text{since } F_{I_1, I_{d+2}} \in \{0, 1\}) \\ &\leq \|\tilde{P}_{[1, d+1]}\|_{\text{frob}} \cdot \|\tilde{P}_{[d+2, 2d+2]}\|_{\text{frob}}. \quad (\text{by [Fact 2.3](#)}) \end{aligned}$$

It suffices to show upper bounds on $\|\tilde{P}_{[1, d+1]}\|_{\text{frob}} \cdot \|\tilde{P}_{[d+2, 2d+2]}\|_{\text{frob}}$. We consider two cases separately.

Case 1: $\eta(1) \neq \eta(3)$. In this case, either $t_1 \in D_L$ and $t_3 \in D_R$ or $t_1 \in D_R$ and $t_3 \in D_L$. For simplicity, consider the case that $t_1 \in D_L, t_3 \in D_R$ and the analysis for the other case is identical. In this case, we see that $\{t \in D, t > t_3\} = (t_3, 2d+2]$. Thus, $\tilde{P}_{[d+2, 2d+2]}$ is the matrix one would obtain by applying [Corollary 3.3](#) to the matrices P_{d+2}, \dots, P_{2d+2} (with parameters $\gamma = 2$ and $t_1 \leftarrow t_3$ and omitted set $\{t_1, t_2, t_3, 1\}$). This implies that

$$\|\tilde{P}_{[d+2, 2d+2]}\|_{\text{frob}} \leq O(d) \cdot \min_{t \in [d+2, 2d+2]} \|P_t\|_{\text{frob}}.$$

Similarly, we see that $\{t \in D, t < t_1\} = [2, t_1)$ and hence, the matrix $\tilde{P}_{[1, d+1]}$ is the matrix one would obtain by applying [Corollary 3.3](#) in reverse order to the transpose of the matrices P_1, \dots, P_{d+1} (with parameters $\gamma = 2$ and $t_1 \leftarrow t_1$ and omitted set $\{t_1, t_2, t_3, d+2\}$). This implies that

$$\|\tilde{P}_{[1, d+1]}\|_{\text{frob}} \leq O(d) \cdot \min_{t \in [1, d+1]} \|P_t\|_{\text{frob}}.$$

Altogether, since $\|P_{t_2^*}\|_{\text{frob}} = \sqrt{W}$ and $\|P_t\|_{\text{frob}} \leq \sqrt{M}$ for all $t \in [2d+2]$, we have

$$\|\tilde{P}_{[d+2,2d+2]}\|_{\text{frob}} \cdot \|\tilde{P}_{[1,d+1]}\|_{\text{frob}} \leq O(d^2) \cdot \sqrt{MW}.$$

Case 2: $\eta(1) = \eta(3)$. In this case, either $t_1, t_3 \in D_L$ or $t_1, t_3 \in D_R$. For simplicity, consider the case that $t_1, t_3 \in D_R$ and the analysis for the other case is identical. In this case, observe that $\{t > t_1, t \in D\} = (t_1, 2d+2]$ and $\{t > t_3, t \in D\} = (t_3, 2d+2]$. Hence, $\tilde{P}_{[d+2,2d+2]}$ is precisely the matrix obtained by applying [Corollary 3.7](#) to the matrices P_{d+2}, \dots, P_{2d+2} (with parameters $t_1 \leftarrow t_1, t_2 \leftarrow t_3$ and omitted set $\{t_1, t_2, t_3, d+2\}$). This implies that

$$\|\tilde{P}_{[d+2,2d+2]}\|_{\text{frob}} \leq O(d^4) \cdot \min_{t \in [d+2,2d+2]} \|P_t\|_{\text{frob}}$$

Similarly, observe that $\tilde{P}_{[1,d+1]}$ is precisely the matrix obtained by applying [Lemma 3.1](#) to the matrices P_1, \dots, P_{d+1} in reverse order (with omitted set $\{t_1, t_2, t_3, d+2\}$). This implies that

$$\|\tilde{P}_{[1,d+1]}\|_{\text{frob}} \leq \min_{t \in [1,d+1]} \|P_t\|_{\text{frob}}.$$

Altogether, since $\|P_{t_2^*}\|_{\text{frob}} = \sqrt{W}$ and $\|P_t\|_{\text{frob}} \leq \sqrt{M}$ for all $t \in [2d+2]$, we have

$$\|\tilde{P}_{[d+2,2d+2]}\|_{\text{frob}} \cdot \|\tilde{P}_{[1,d+1]}\|_{\text{frob}} \leq O(d^4) \cdot \sqrt{MW}.$$

This completes the proof.

6 Higher Level Fourier Growth from Lower Levels

In this section, we will present the bootstrapping argument that allows us to prove higher level Fourier growth bounds from lower levels. A version of this argument originally appeared in [\[CHLT19\]](#).

Theorem 6.1. *Let \mathcal{F} be any restriction-closed family of boolean functions on $\{-1, 1\}^N$. Then, for all $\ell \in [N]$, we have $L_{1,\ell+1}(\mathcal{F}) \leq L_{1,\ell}(\mathcal{F}) \cdot O\left(\sqrt{N \log \binom{N}{\ell} \log N}\right)$.*

Using the fact that $L_{1,1}$ of d -query bounded low-degree polynomials is at most $O(d)$ [\[IRR⁺21\]](#), we obtain the following corollary, which as a special case, includes $d/2$ -query BQP algorithms.

Corollary 6.2 ([Theorem 1.7](#)). *Let f be a degree- d polynomial bounded on $\{\pm 1\}^N$ by $[-1, 1]$ and ρ be any restriction, then $L_{1,\ell}(f|_\rho) \leq c^\ell \cdot d \cdot N^{(\ell-1)/2} \cdot \log^{\ell-1}(N) \cdot \sqrt{\ell!}$ for some constant $c > 0$.*

Using the fact that $L_{1,2}$ of d -query DQC_k algorithms is at most $O(d^3)$ as proved in [Section 4](#), we obtain the following corollary.

Corollary 6.3 ([Theorem 1.5](#)). *Let f be a d -query DQC_k algorithm on oracle of length N and ρ be any restriction, then $L_{1,\ell}(f|_\rho) \leq \sqrt{K} \cdot c^\ell \cdot d^3 \cdot N^{(\ell-2)/2} \cdot \log^{\ell-2}(N) \cdot \sqrt{\ell!}$ for some constant $c > 0$.*

For $\frac{1}{2}$ BQP algorithms, combining [Theorem 6.1](#) and the level-3 bound will only give a $\text{poly}(d) \cdot N^2$ bound on the level-6 bound, which does not suffice to prove lower bounds for 3-FORRELATION. We will require a more refined analysis described in [Section 6.1](#).

Proof of Theorem 6.1. Fix any $f \in \mathcal{F}$ and any signs $\alpha_S \in [-1, 1]$ for $|S| = \ell + 1$. We wish to bound $L_{1,\ell+1}^\alpha(f) = \sum_{|S|=\ell+1} \alpha_S \hat{f}(S)$. We randomly partition $[N]$ into $X \sqcup Y$ as follows. For each $i \in [N]$, independently include it in X with probability $1/(\ell + 1)$. Define

$$L(X) := \sum_{|S|=\ell+1} \alpha_S \cdot \hat{f}(S) \cdot \mathbb{1}[|S \cap X| = 1] = \sum_{\substack{i \in X \\ T \subseteq [Y], |T|=\ell}} \alpha_{T \cup \{i\}} \hat{f}(T \cup \{i\}). \quad (27)$$

Observe that for any set $S \subseteq [N]$ of size $\ell + 1$, the probability that $|S \cap X| = 1$ is precisely $c(\ell) = (\ell + 1) \cdot \frac{1}{\ell+1} \cdot \left(\frac{\ell}{\ell+1}\right)^\ell \geq \Omega(1)$. This implies that

$$L_{1,\ell+1}^\alpha(f) = \mathbb{E}_X[L(X)] \cdot \frac{1}{c(\ell)}. \quad (28)$$

Since $c(\ell) \geq \Omega(1)$, it suffices to upper bound $\mathbb{E}_X[L(X)]$ and we will in fact bound $L(X)$ for any partition X . To do this, we express $\hat{f}(T \cup \{i\})$ as $\mathbb{E}_{\substack{x \sim \{\pm 1\}^X \\ y \sim \{\pm 1\}^Y}} [f(x, y) \cdot x_i \cdot \chi_T(y)]$ where the underlying distributions are uniform – this follows from the definition of Fourier coefficients. Substituting this in Equation (27), we obtain

$$\begin{aligned} L(X) &= \sum_{\substack{i \in X \\ T \subseteq [Y], |T|=\ell}} \alpha_{T \cup \{i\}} \cdot \mathbb{E}_{\substack{x \sim \{\pm 1\}^X \\ y \sim \{\pm 1\}^Y}} [f(x, y) \cdot x_i \cdot \chi_T(y)] \\ &= \mathbb{E}_{\substack{x \sim \{\pm 1\}^X \\ y \sim \{\pm 1\}^Y}} \left[f(x, y) \cdot \sum_{T \subseteq [Y], |T|=\ell} \chi_T(y) \cdot \sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}} \right]. \end{aligned} \quad (29)$$

Consider the function f_x where $f_x(y) := f(x, y)$ for all $y \in \{\pm 1\}^Y$. By definition, we have $\hat{f}_x(T) = \mathbb{E}_{Y \sim \{\pm 1\}^Y} [f(x, y) \cdot \chi_T(y)]$. Substituting this above, we see that

$$\begin{aligned} L(X) &= \mathbb{E}_{x \sim \{\pm 1\}^X} \left[\sum_{T \subseteq [Y], |T|=\ell} \hat{f}_x(T) \cdot \sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}} \right] \\ &\leq \mathbb{E}_{x \sim \{\pm 1\}^X} \left[\sum_{T \subseteq [Y], |T|=\ell} |\hat{f}_x(T)| \cdot \max_{T \subseteq [Y], |T|=\ell} \left| \sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}} \right| \right]. \end{aligned} \quad (30)$$

We now use the fact that $f_x \in \mathcal{F}$ due the restriction-closed property and that $\sum_{T \subseteq [Y], |T|=\ell} |\hat{f}_x(T)| = L_{1,\ell}(f_x) \leq L_{1,\ell}(\mathcal{F})$ to conclude that

$$L(X) \leq L_{1,\ell}(\mathcal{F}) \cdot \mathbb{E}_{x \sim \{\pm 1\}^X} \max_{T \subseteq [Y], |T|=\ell} \left[\left| \sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}} \right| \right].$$

Fix any $T \subseteq [Y]$ with $|T| = \ell$. We now observe that when $x \sim \{\pm 1\}^X$, the random variable $\sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}}$ is a sum of $|X|$ many independent $[-1, 1]$ -random variables where $|X| \leq N$, hence,

$$\Pr \left[\left| \sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}} \right| \geq t\sqrt{N} \right] \leq \exp(-\Omega(t^2)).$$

We now set $t = \Theta(\sqrt{\log N} \cdot \sqrt{\log \binom{N}{\ell}})$ so that we can union bound over all $T \subseteq [Y]$ with $|T| = \ell$, to conclude that with probability at least $1 - 1/N$ over $x \sim \{\pm 1\}^X$, for all $T \subseteq [Y]$ with $|T| = \ell$,

$$\left| \sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}} \right| \leq O \left(\sqrt{N} \cdot \sqrt{\log \binom{N}{\ell}} \cdot \sqrt{\log N} \right).$$

For all $x \in \{\pm 1\}^X$, we have that the L.H.S. above is at most N . This, along with the above implies that

$$\begin{aligned} \mathbb{E}_{x \sim \{\pm 1\}^X} \left[\max_{T \subseteq [Y], |T| = \ell} \left| \sum_{i \in X} x_i \cdot \alpha_{T \cup \{i\}} \right| \right] &\leq \frac{1}{N} \cdot N + \sqrt{N} \cdot \sqrt{\log \binom{N}{\ell}} \cdot \sqrt{\log N} \\ &\leq O \left(\sqrt{N \log \binom{N}{\ell} \log N} \right) + 1. \end{aligned}$$

Finally, we observe that if $\ell = N$, then $L_{1, \ell+1}(\mathcal{F}) = 0$ and if $\ell < N$, then $\log \binom{N}{\ell} \geq 1$, and hence we can ignore the $+1$ factor in the R.H.S. by absorbing into the $O(\cdot)$. This completes the proof. \square

6.1 Improved Bootstrapping for $\frac{1}{2}$ BQP algorithms

Theorem 6.4. *Let \mathcal{F} be a restriction-closed family of boolean functions of degree at most d on $\{\pm 1\}^{3N}$. Let $\gamma, \gamma' \in [-1, 1]^{3N}$ and $\alpha(\gamma), \beta(\gamma, \gamma')$ be signs as in [Definition 2.5](#). Then, for any restriction ρ on $[3N]$, $L_{1,6}^{\beta(\gamma, \gamma')}(f|_\rho) \leq \max_{\gamma''} (L_{1,3}^{\alpha(\gamma'')}(f)) \cdot O(d^3 N)$, where the maximum is over $\gamma'' \in [-1, 1]^{3N}$.*

Using the bound on $L_{1,3}^{\alpha(\gamma)}$ for $\frac{1}{2}$ BQP algorithms, we obtain the following corollary.

Corollary 6.5 ([Theorem 1.6](#)). *Let $\gamma, \gamma' \in [-1, 1]^{3N}$ and $\beta(\gamma, \gamma')$ be signs as in [Definition 2.5](#). Let f be a d -query $\frac{1}{2}$ BQP algorithm on oracle of length N . Then, $L_{1,6}^{\beta(\gamma, \gamma')}(f) \leq d^{10} \cdot N^{3/2}$.*

Proof of [Theorem 6.4](#). Our goal is to upper bound

$$L_{1,6}^{\beta(\gamma, \gamma')}(f) = \sum_{\substack{i_1 < i_4 \in A \\ i_2 < i_5 \in B \\ i_3 < i_6 \in C}} \beta(\gamma, \gamma')_{i_1, \dots, i_6} \cdot \widehat{f}(\{i_1, \dots, i_6\})$$

Let $A = [N], B = (N, 2N], C = (2N, 3N]$ as in [Definition 2.5](#). Partition these sets as $A = X_A \sqcup Y_A, B = X_B \sqcup Y_B, C = X_C \sqcup Y_C$, where for each element, we independently include it in the corresponding X with probability $1/2$. Let $X = X_A \cup X_B \cup X_C$. Define

$$L(X) := \sum_{\substack{i_1 \in X_A, i_2 \in X_B, i_3 \in X_C \\ i_4 \in Y_A, i_5 \in Y_B, i_6 \in Y_C}} \beta(\gamma, \gamma')_{i_1, \dots, i_6} \widehat{f}(\{i_1, \dots, i_6\}). \quad (31)$$

Observe that for any $i_1 < i_4 \in A, i_2 < i_5 \in B, i_3 < i_6 \in C$, the probability over X that this term contributes to $L(X)$ is precisely the probability that $\{i_1, i_2, i_3\} \subseteq X$ and $\{i_4, i_5, i_6\} \subseteq Y$ which is exactly 2^{-6} . Therefore,

$$L_{1,6}^{\beta(\gamma, \gamma')}(f|_\rho) = \mathbb{E}_X [L(X)] \cdot \frac{1}{2^6}.$$

It thus suffices to upper bound $L(X)$ for any partition X . As before, we will express $\widehat{f}(\{i_1, \dots, i_6\})$ as $\mathbb{E}_{\substack{x \sim \{\pm 1\}^X \\ y \sim \{\pm 1\}^Y}} \left[f(x, y) \prod_{t \in [6]} x_t \right]$, where the underlying distributions are uniform – this follows from the definition of Fourier coefficients. Substituting this in [Equation \(31\)](#), we see that

$$\begin{aligned} L(X) &= \sum_{\substack{i_1 \in X_A, i_2 \in X_B, i_3 \in X_C \\ i_4 \in Y_A, i_5 \in Y_B, i_6 \in Y_C}} \beta(\gamma, \gamma')_{i_1, \dots, i_6} \mathbb{E}_{\substack{x \sim \{\pm 1\}^X \\ y \sim \{\pm 1\}^Y}} \left[f(x, y) \prod_{t \in [6]} x_t \right] \\ &= \mathbb{E}_{x \sim \{\pm 1\}^X} \left[\sum_{\substack{i_1 \in X_A \\ i_2 \in X_B \\ i_3 \in X_C}} \alpha(\gamma)_{i_1, i_2, i_3} \cdot x_{i_1} x_{i_2} x_{i_3} \cdot \mathbb{E}_{y \sim \{\pm 1\}^Y} \left[\sum_{\substack{i_4 \in Y_A \\ i_5 \in Y_B \\ i_6 \in Y_C}} [f(x, y) \cdot \alpha(\gamma')_{i_4, i_5, i_6} \cdot y_{i_4} y_{i_5} y_{i_6}] \right] \right] \end{aligned} \quad (32)$$

where in the last line, we used the fact that $\beta(\gamma, \gamma')_{i_1, \dots, i_6} = \alpha(\gamma)_{i_1, i_2, i_3} \cdot \alpha(\gamma')_{i_4, i_5, i_6}$. As before, we define a function f_x such that $f_x(y) := f(x, y)$ for all $y \in \{\pm 1\}^Y$. Define a function $g(x)$ at $x \in \{\pm 1\}^X$ by

$$g(x) := \mathbb{E}_{y \sim \{\pm 1\}^Y} \left[\sum_{\substack{i_4 \in Y_A \\ i_5 \in Y_B \\ i_6 \in Y_C}} [f(x, y) \cdot \alpha(\gamma')_{i_4, i_5, i_6} \cdot y_{i_4} y_{i_5} y_{i_6}] \right] \triangleq \sum_{\substack{i_4 \in Y_A \\ i_5 \in Y_B \\ i_6 \in Y_C}} \alpha(\gamma')_{i_4, i_5, i_6} \cdot \widehat{f}_x(\{i_4, i_5, i_6\}) \quad (33)$$

where the last equality follows by the definition of Fourier coefficients. Substituting [Equation \(33\)](#) in [Equation \(32\)](#), we obtain

$$L(X) = \mathbb{E}_{x \sim \{\pm 1\}^X} \left[\sum_{\substack{i_1 \in X_A \\ i_2 \in X_B \\ i_3 \in X_C}} \alpha(\gamma)_{i_1, i_2, i_3} \cdot x_{i_1} x_{i_2} x_{i_3} \cdot g(x) \right] = \sum_{\substack{i_1 \in X_A \\ i_2 \in X_B \\ i_3 \in X_C}} \alpha(\gamma)_{i_1, i_2, i_3} \cdot \widehat{g}(\{i_1, i_2, i_3\}) \quad (34)$$

where again, the last equality follows by the definition of Fourier coefficients. Furthermore, by defining γ'' to be equal to γ in X and 0 otherwise, we see that the R.H.S. of [Equation \(34\)](#) equals $L_{1,3}^{\alpha(\gamma'')}(g)$. Observe from [Equation \(33\)](#) that $g(x)$ is a polynomial in x of degree at most d , since $f(x, y)$ is of degree at most d . We now use the level-3 bounds for bounded low-degree polynomials from [\[IRR⁺21\]](#) to conclude that

$$L_{1,3}^{\alpha(\gamma'')}(g) \leq O(d^3 N) \cdot \max_{x \in \{\pm 1\}^X} |g(x)|.$$

We will now show that $|g(x)|$ is bounded for all $x \in \{\pm 1\}^X$. Since \mathcal{F} is restriction closed, we have $f_x \in \mathcal{F}$. Furthermore, setting γ'' to be equal to γ' on Y and 0 otherwise, and recalling [Equation \(33\)](#), we obtain that

$$|g(x)| = \left| \sum_{\substack{i_4 \in Y_A \\ i_5 \in Y_B \\ i_6 \in Y_C}} \alpha(\gamma')_{i_4, i_5, i_6} \cdot \widehat{f}_x(\{i_4, i_5, i_6\}) \right| = \max(L_{1,3}^{\alpha(\gamma'')}(f_x), L_{1,3}^{-\alpha(\gamma'')}(f_x)) \leq \max_{\gamma''} (L_{1,3}^{\alpha(\gamma'')}(\mathcal{F})).$$

This completes the proof. \square

7 Acknowledgements

The author would like to thank Vishnu Iyer, Dale Jacobs, Natalie Parham, Makrand Sinha, Kewen Wu and Henry Yuen for very helpful comments and discussions. The author is also grateful to Francisco Escudero Gutierrez and Miquel Saucedo Cuesta for pointing out a critical bug in an earlier version of this paper that has since been corrected.

References

- [AA11] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing, STOC '11*, page 333–342, New York, NY, USA, 2011. Association for Computing Machinery. [3](#)
- [AA15] Scott Aaronson and Andris Ambainis. Forrelation: A problem that optimally separates quantum from classical computing. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*, pages 307–316, 2015. [3](#), [6](#), [9](#)
- [Aar10] Scott Aaronson. BQP and the polynomial hierarchy. In *Proceedings of the 61st IEEE Annual Symposium on Foundations of Computer Science*, pages 141–150, 2010. [3](#), [6](#), [9](#)
- [ABKM17] Scott Aaronson, Adam Bouland, Greg Kuperberg, and Saeed Mehraban. The computational complexity of ball permutations. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 317–327. ACM, 2017. [3](#), [4](#)
- [AG23] Srinivasan Arunachalam and Uma Girish. Trade-offs between entanglement and communication. In *Proceedings of the 38th Computational Complexity Conference, CCC '23*, Dagstuhl, DEU, 2023. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. [8](#)
- [AGL23] Srinivasan Arunachalam, Uma Girish, and Noam Lifshitz. One Clean Qubit Suffices for Quantum Communication Advantage. 10 2023. [4](#)
- [BC26] Quentin Buzet and André Chailloux. Iqp circuits for 2-forrelation. *arXiv preprint arXiv:2604.15248*, 2026. [15](#)
- [BS21] Nikhil Bansal and Makrand Sinha. k -forrelation optimally separates quantum and classical query complexity. In Samir Khuller and Virginia Vassilevska Williams, editors, *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1303–1316, 2021. [6](#), [7](#), [8](#), [10](#), [17](#), [18](#)
- [BV97] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997. [3](#)
- [CCHL23] Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li. The complexity of nisq. *Nature Communications*, 14(1):6001, 2023. [3](#), [10](#), [15](#)
- [CHHK24] Nai-Hui Chia, Min-Hsiu Hsieh, Shih-Han Hung, and En-Jui Kuo. Oracle separation between noisy quantum polynomial time and the polynomial hierarchy, 2024. [15](#)

- [CHLT19] Eshan Chattopadhyay, Pooya Hatami, Shachar Lovett, and Avishay Tal. Pseudorandom generators from the second fourier level and applications to AC0 with parity gates. In Avrim Blum, editor, *Proceedings of the 10th Innovations in Theoretical Computer Science Conference*, volume 124 of *LIPICs*, pages 22:1–22:15, 2019. 6, 7, 9, 17, 34
- [CSS21] Anirban N. Chowdhury, Rolando D. Somma, and Yigit Subasi. Computing partition functions in the one-clean-qubit model. *Phys. Rev. A*, 103:032422, Mar 2021. 4
- [DFC05] Animesh Datta, Steven T. Flammia, and Carlton M. Caves. Entanglement and the power of one qubit. *Phys. Rev. A*, 72:042316, Oct 2005. 4
- [DJ92] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439:553–558, 1992. 3
- [FKM⁺15] Keisuke Fujii, Hiro tada Kobayashi, Tomoyuki Morimae, Harumichi Nishimura, Shuhei Tamate, and Seiichiro Tani. Power of quantum computation with few clean qubits. *arXiv preprint arXiv:1509.07276*, 2015. 4
- [FKM⁺18] Keisuke Fujii, Hiro tada Kobayashi, Tomoyuki Morimae, Harumichi Nishimura, Shuhei Tamate, and Seiichiro Tani. Impossibility of classically simulating one-clean-qubit model with multiplicative error. *Physical review letters*, 120(20):200502, 2018. 4
- [Geo25] Alexandru Georghiu. Verifiable quantum advantage: old and new ideas, 2025. Link to recording of talk: <https://youtu.be/7NqAcaSwKf8?si=2ZBkEkbiVaUMSzMJ>. 15
- [GSTW24] Uma Girish, Makrand Sinha, Avishay Tal, and Kewen Wu. The power of adaptivity in quantum query algorithms. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 1488–1497, 2024. 8, 9, 11, 20
- [GTW21] Uma Girish, Avishay Tal, and Kewen Wu. Fourier growth of parity decision trees. In *36th Computational Complexity Conference (CCC 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021. 6
- [IRR⁺21] Siddharth Iyer, Anup Rao, Victor Reis, Thomas Rothvoss, and Amir Yehudayoff. Tight bounds on the fourier growth of bounded functions on the hypercube. *CoRR*, abs/2107.06309, 2021. 6, 7, 8, 9, 15, 34, 37
- [JM24] Dale Jacobs and Saeed Mehraban. Space just above one clean qubit, 2024. 3, 4, 5, 10, 15
- [KL98] E. Knill and R. Laflamme. Power of one bit of quantum information. *Phys. Rev. Lett.*, 81:5672–5675, Dec 1998. 3, 4, 19
- [Man95] Yishay Mansour. An $O(n^{\log \log n})$ learning algorithm for DNF under the uniform distribution. *J. Comput. Syst. Sci.*, 50(3):543–550, 1995. Appeared in COLT, 1992. 6
- [MFF14] Tomoyuki Morimae, Keisuke Fujii, and Joseph F Fitzsimons. Hardness of classically simulating the one-clean-qubit model. *Physical review letters*, 112(13):130502, 2014. 3, 4
- [Mor17] Tomoyuki Morimae. Hardness of classically sampling the one-clean-qubit model with constant total variation distance error. *Physical Review A*, 96(4):040302, 2017. 4

- [RT22] Ran Raz and Avishay Tal. Oracle separation of BQP and PH. *J. ACM*, 69(4):30:1–30:21, 2022. [6](#), [7](#), [9](#), [17](#)
- [SB08] Dan J. Shepherd and Michael J. Bremner. Temporally unstructured quantum computation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 465:1413 – 1439, 2008. [3](#)
- [She10] Daniel James Shepherd. Quantum complexity: restrictions on algorithms and architectures. *CoRR*, abs/1005.1425, 2010. [4](#)
- [Shu25] Noah Shatty. Simons institute summer cluster on quantum computing: Lightning talks, 2025. Link to recording of talk: <https://www.youtube.com/live/7F5LBNGDRmk?si=NhPVNL25qGTtcngP&t=2651>. [15](#)
- [Sim97] Daniel R. Simon. On the power of quantum computation. *SIAM J. Comput.*, 26(5):1474–1483, October 1997. [3](#)
- [SJ08] Peter W. Shor and Stephen P. Jordan. Estimating jones polynomials is a complete problem for one clean qubit. *Quantum Info. Comput.*, 8(8):681–714, September 2008. [4](#)
- [SSW23] Alexander A. Sherstov, Andrey A. Storozhenko, and Pei Wu. An optimal separation of randomized and quantum query complexity. *SIAM J. Comput.*, 52(2):525–567, 2023. [6](#), [7](#), [9](#)
- [Tal17] Avishay Tal. Tight bounds on the Fourier spectrum of AC0. In *Computational Complexity Conference*, volume 79 of *LIPICs*, pages 15:1–15:31. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017. [6](#)
- [Tal20] Avishay Tal. Towards optimal separations between quantum and randomized query complexities. In *Proceedings of the 61st IEEE Annual Symposium on Foundations of Computer Science*, pages 228–239, 2020. [6](#), [7](#), [9](#)

A Appendix

A.1 Proof of [Lemma 3.4](#)

Proof of [Lemma 3.4](#). For $i_1 \in [N]$, let $P^{(i_1)}$ be the $M \times M$ projection matrix $(\mathbb{I} - |i_1\rangle\langle i_1|) \otimes \mathbb{I}$ for $i_1 \in [N]$ and let $Q^{(i_1)} = \mathbb{I} - P^{(i_1)} = |i_1\rangle\langle i_1| \otimes \mathbb{I}$. Define $\tilde{P}_t^{(i_1)} = U_t \cdot P^{(i_1)}$ for all $i_1 \in [N], t \in [t^* - 1, d - 1]$ and $\tilde{P}_d = U_d$. For $i_1, i_{d+1} \in [N]$, let $P^{(i_1, i_{d+1})} = P^{(i_1)} \cdot P^{(i_{d+1})}$, and let $\tilde{P}_t^{(i_1, i_{d+1})} = U_t \cdot P^{(i_1, i_{d+1})}$ for $t \in [t^* - 1, d - 1]$. As before, we have

$$Q^{(i')\dagger} \cdot Q^{(i)} = Q^{(i)\dagger} \cdot Q^{(i')} = 0 \text{ if } i \neq i', \quad (35)$$

$$\sum_i Q^{(i)} = \sum_i Q^{(i)\dagger} \cdot Q^{(i)} = \sum_i Q^{(i)} \cdot Q^{(i)\dagger} = \mathbb{I}, \quad (36)$$

$$Q^{(i)} = \mathbb{I} - P^{(i)}. \quad (37)$$

Observe that

$$U = \sum_{i_1, i_{d+1} \in [N]} Q^{(i_1)} \cdot U_{[1, t^* - 1]} \cdot \tilde{P}_{[t^* - 1, d - 1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d \cdot Q^{(i_{d+1})}.$$

Define

$$\begin{aligned}
U' &= \sum_{i_1, i_{d+1} \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, d-1]}^{(i_1)} \cdot \tilde{P}_d \cdot Q^{(i_{d+1})} \\
&= \sum_{i_1 \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, d-1]}^{(i_1)} \cdot \tilde{P}_d \quad (\text{since } \sum_{i_{d+1} \in [N]} Q^{(i_{d+1})} = \mathbb{I}.)
\end{aligned}$$

Observe that U' is precisely the matrix we would get by an application of [Lemma 3.2](#) to the matrices $U_{[1, t^*-1]}, U_{t^*-1}, U_{t^*}, U_{t^*+1}, \dots, U_d$ with $\gamma = 1$. By [Lemma 3.2](#), we have

$$\|U'\|_{\text{frob}} \leq O(d) \cdot \|U_{t^*}\|_{\text{frob}}.$$

We now compare U and U' . We observe that the only difference between U and U' comes $\tilde{P}_{[t^*-1, d-1]}^{(i_1, i_{d+1})}$ versus $\tilde{P}_{[t^*-1, d-1]}^{(i_1)}$. As before, we use the telescoping expansion to obtain

$$\begin{aligned}
&U' - U \\
&= \sum_{t' \in [t^*, d]} \sum_{i_1, i_{d+1} \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, t'-1]}^{(i_1)} \cdot \left(\tilde{P}_{t'-1}^{(i_1)} - \tilde{P}_{t'-1}^{(i_1, i_{d+1})} \right) \cdot \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})} \\
&= \sum_{t' \in [t^*, d]} \sum_{i_1, i_{d+1} \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, t'-1]}^{(i_1)} \cdot \left(\tilde{P}_{t'-1}^{(i_1)} \cdot Q^{(i_{d+1})} \right) \cdot \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})} \\
&\hspace{20em} (\text{by definition and [Equation \(37\)](#)}) \\
&= \sum_{t' \in [t^*, d]} W_{t'}, \text{ where} \\
&W_{t'} = \sum_{i_1 \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, t'-1]}^{(i_1)} \cdot \underbrace{\sum_{i_{d+1} \in [N]} Q^{(i_{d+1})} \cdot \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})}}_{Z_{t'}^{(i_1)}}.
\end{aligned}$$

We will show that for all $t' \in [t^*, d]$, we have $\|W_{t'}\|_{\text{frob}} \leq O(1) \cdot \|U_{t^*}\|_{\text{frob}}$ and this would complete the proof. Suppose $t' = t^*$, then the corresponding term is

$$W_{t^*} := \sum_{i_1 \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{t^*-1}^{(i_1)} \cdot \sum_{i_{d+1} \in [N]} Q^{(i_{d+1})} \cdot \tilde{P}_{[t^*, d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})}$$

Observe that

$$\begin{aligned}
\|W_{t'}\|_{\text{frob}}^2 &= \sum_{i_1, i_{d+1} \in [N]} \left\| Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{t^*-1}^{(i_1)} \cdot Q^{(i_{d+1})} \cdot \tilde{P}_{[t^*, d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})} \right\|_{\text{frob}}^2 \\
&\hspace{15em} (\text{Equation (35) applied to } Q^{(i_1)}, Q^{(i_{d+1})}) \\
&\leq \sum_{\substack{i_1 \in [N] \\ i_{d+1} \in [N]}} \left\| Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{t^*-1}^{(i_1)} \cdot Q^{(i_{d+1})} \cdot U_{t^*} \right\|_{\text{frob}}^2 \\
&\hspace{10em} (\text{by [Fact 2.2](#) and since the rest of the terms have } \|\cdot\|_{\text{op}} \leq 1) \\
&= \sum_{i_{d+1} \in [N]} \left\| \sum_{i_1 \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{t^*-1}^{(i_1)} \cdot Q^{(i_{d+1})} \cdot U_{t^*} \right\|_{\text{frob}}^2 \\
&\hspace{15em} (\text{Equation (35) applied to } Q^{(i_1)})
\end{aligned}$$

$$\begin{aligned}
&\leq \sum_{i_{d+1} \in [N]} \left\| \sum_{i_1 \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{t^*-1}^{(i_1)} \right\|_{\text{op}}^2 \cdot \left\| Q^{(i_{d+1})} \cdot U_{t^*} \right\|_{\text{frob}}^2 && \text{(by Fact 2.2)} \\
&\leq 4 \sum_{i_{d+1} \in [N]} \left\| Q^{(i_{d+1})} \cdot U_{t^*} \right\|_{\text{frob}}^2 && \text{(by Lemma 3.2 applied to } U_{[1, t^*-1]}\text{)} \\
&= 4 \sum_{i_{d+1} \in [N]} \text{Tr} \left(Q^{(i_{d+1})} \cdot U_{t^*} \cdot U_{t^*}^\dagger \cdot Q^{(i_{d+1})^\dagger} \right) \\
&= 4 \text{Tr}(U_{t^*} \cdot U_{t^*}^\dagger) = 4 \|U_{t^*}\|_{\text{frob}}^2 && \text{(by Equation (36) applied to } Q^{(i_{d+1})}\text{)}
\end{aligned}$$

We will now show that for $t' > t^*$, $\|W_{t'}\|_{\text{frob}} \leq O(d) \cdot \|U_{t^*}\|_{\text{frob}}$. To do this, we first claim that $\|Z_{t'_1}^{(i_1)}\|_{\text{op}} \leq 1$. To see this, observe that

$$\begin{aligned}
Z_{t'_1}^{(i_1)} \cdot Z_{t'_1}^{(i_1)\dagger} &\triangleq \left(\sum_{i_{d+1}} Q^{(i_{d+1})} \cdot \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})} \right) \cdot \left(\sum_{i_{d+1}} Q^{(i_{d+1})} \cdot \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})} \right)^\dagger \\
&= \sum_{i_{d+1}} Q^{(i_{d+1})} \cdot \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})} \cdot Q^{(i_{d+1})\dagger} \cdot \tilde{P}_d^{(i_1)\dagger} \cdot \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})\dagger} \cdot Q^{(i_{d+1})\dagger} \\
&\preceq \sum_{i_{d+1}} Q^{(i_{d+1})} \cdot Q^{(i_{d+1})\dagger} = \mathbb{I}. && \text{(Equation (35))} \\
&&& \text{(since } \left\| \tilde{P}_{[t', d-1]}^{(i_1, i_{d+1})} \cdot \tilde{P}_d^{(i_1)} \cdot Q^{(i_{d+1})} \right\|_{\text{op}} \leq 1\text{.)}
\end{aligned}$$

Therefore, we obtain

$$\begin{aligned}
\|W_{t'}\|_{\text{frob}}^2 &\triangleq \left\| \sum_{i_1 \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, t'_1-1]}^{(i_1)} \cdot Z_{t'_1}^{(i_1)} \right\|_{\text{frob}}^2 \\
&= \sum_{i_1 \in [N]} \left\| Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, t'_1-1]}^{(i_1)} \cdot Z_{t'_1}^{(i_1)} \right\|_{\text{frob}}^2 && \text{(Equation (35))} \\
&\leq \sum_{i_1 \in [N]} \left\| Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, t'_1-1]}^{(i_1)} \right\|_{\text{frob}}^2 && \text{(since } \|Z_{t'_1}^{(i_1)}\|_{\text{op}} \leq 1\text{)} \\
&= \left\| \sum_{i_1 \in [N]} Q^{(i_1)} \cdot U_{[1, t^*-1]} \cdot \tilde{P}_{[t^*-1, t'_1-1]}^{(i_1)} \right\|_{\text{frob}}^2 && \text{(Equation (35))} \\
&\leq O(1) \cdot \|U_{t^*}\|_{\text{frob}}^2 \\
&\text{(by an application of Lemma 3.2 to matrices } U_{[1, t^*-1]}, U_{t^*}, \dots, U_{t'_1-1} \text{ with } t_1 = 1\text{.)}
\end{aligned}$$

Thus, we get $\|U\|_{\text{frob}} \leq O(d) \cdot \|U_{t^*}\|_{\text{frob}}$. This completes the proof. \square

A.2 Proof of Lemma 3.6

Lemma A.1. For $k \in [K]$ let $X_k \in \mathbb{C}^{M \times N}$, $Y_k \in \mathbb{C}^{N \times M}$ and $C \in \mathbb{C}^{N \times N}$. Suppose for a constant $a > 0$, we have $\sum_{k \in [K]} X_k^\dagger X_k \preceq a^2 \cdot \mathbb{I}$, $\sum_{k \in [K]} Y_k^\dagger Y_k \preceq \mathbb{I}$. Then,

$$\left\| \sum_{k \in [K]} X_k C Y_k \right\|_{\text{frob}} \leq a \cdot \|C\|_{\text{frob}}.$$

Proof of Lemma A.1. Consider

$$\left\| \sum_k X_k C Y_k \right\|_{\text{frob}}^2 = \left\| [X_1 C \ \dots \ X_k C] \cdot \begin{bmatrix} Y_1 \\ \vdots \\ Y_k \end{bmatrix} \right\|_{\text{frob}}^2 \leq \left\| [X_1 C \ \dots \ X_k C] \right\|_{\text{frob}}^2 \cdot \left\| \begin{bmatrix} Y_1 \\ \vdots \\ Y_k \end{bmatrix} \right\|_{\text{op}}^2$$

where the last inequality follows by [Fact 2.2](#). Firstly, we observe that

$$\left\| \begin{bmatrix} Y_1 \\ \vdots \\ Y_k \end{bmatrix} \right\|_{\text{op}}^2 = \left\| \begin{bmatrix} Y_1^\dagger & \dots & Y_k^\dagger \end{bmatrix} \cdot \begin{bmatrix} Y_1 \\ \vdots \\ Y_k \end{bmatrix} \right\|_{\text{op}} = \left\| \sum_{k \in [K]} Y_k^\dagger Y_k \right\|_{\text{op}} \leq 1.$$

Next, we observe that

$$\left\| [X_1 C \ \dots \ X_k C] \right\|_{\text{frob}}^2 = \text{Tr} \left(\sum_{k \in [K]} X_k C C^\dagger X_k^\dagger \right) = \text{Tr} \left(\sum_{k \in [K]} X_k^\dagger X_k \cdot C C^\dagger \right) \leq a^2 \cdot \|C\|_{\text{frob}}^2.$$

□

Fact A.2. For matrices A_1, \dots, A_K we have $\left(\sum_{k=1}^K A_k \right)^\dagger \cdot \left(\sum_{k=1}^K A_k \right) \preceq K \cdot \sum_{k=1}^K A_k^\dagger A_k$.

Proof of Fact A.2. Observe that for all pairs $k < k' \in [K]$, we have

$$(A_k - A_{k'})^\dagger (A_k - A_{k'}) \succeq 0 \implies A_k^\dagger A_{k'} + A_{k'}^\dagger A_k \preceq A_k^\dagger A_k + A_{k'}^\dagger A_{k'}.$$

Summing this over all $k < k' \in [K]$ and adding $\sum_{k=1}^K A_k^\dagger A_k$ on both sides gives the desired inequality. □

Lemma A.3. For $i, j \in [N]$, let $F^{(i,j)} \in \mathbb{C}^{M \times M}$ be a matrix. Furthermore, assume that for some parameter $K \in \mathbb{N}$, there exist matrices $F_1^{(i,j)}, \dots, F_K^{(i,j)}$ such that $F^{(i,j)} = \sum_{k=1}^K F_k^{(i,j)}$ and for all $k \in [K]$, we have

$$\begin{aligned} \sum_{i,j} F_k^{(i,j)\dagger} \cdot F_k^{(i,j)} &\preceq \mathbb{I}, & \sum_i \left(\sum_j F_k^{(i,j)\dagger} \right) \cdot \left(\sum_j F_k^{(i,j)} \right) &\preceq \mathbb{I} \\ \sum_j \left(\sum_i F_k^{(i,j)\dagger} \right) \cdot \left(\sum_i F_k^{(i,j)} \right) &\preceq \mathbb{I}, & \left(\sum_{i,j} F_k^{(i,j)\dagger} \right) \cdot \left(\sum_{i,j} F_k^{(i,j)} \right) &\preceq \mathbb{I}. \end{aligned}$$

Then, the same inequalities hold for $F^{(i,j)}$ with the R.H.S. above replaced by $K^2 \cdot \mathbb{I}$.

Proof of Lemma A.3. We apply [Fact A.2](#) to the matrices $F^{(i,j)} = \sum_{k=1}^K F_k^{(i,j)}$ and then sum over i, j to obtain

$$\sum_{i,j} F^{(i,j)\dagger} \cdot F^{(i,j)} \preceq K \cdot \sum_{i,j} \sum_{k=1}^K F_k^{(i,j)\dagger} \cdot F_k^{(i,j)} \preceq K^2 \cdot \mathbb{I}$$

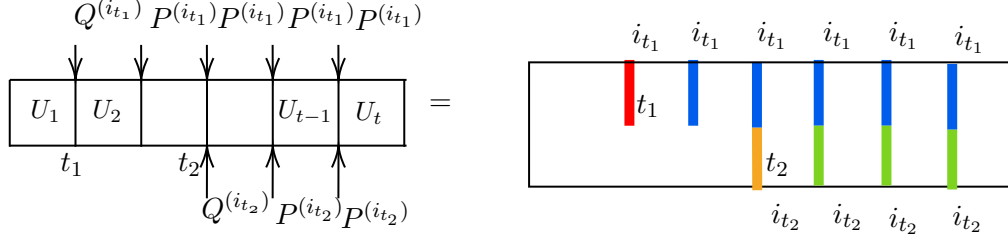


Figure 9: We use red, blue, orange and green lines to denote insertions of $Q^{(i_{t_1})}, P^{(i_{t_1})}, Q^{(i_{t_2})}$ and $P^{(i_{t_2})}$ respectively.

We apply [Fact A.2](#) to the matrices $\sum_j F^{(i,j)} = \sum_{k=1}^K (\sum_j F_k^{(i,j)})$ and then sum over i to obtain

$$\sum_i \left(\sum_j F^{(i,j)\dagger} \right) \cdot \left(\sum_j F^{(i,j)} \right) \preceq K \cdot \sum_i \sum_{k=1}^K \left(\sum_j F_k^{(i,j)\dagger} \right) \cdot \left(\sum_j F_k^{(i,j)} \right) \preceq K^2 \cdot \mathbb{I}$$

The same calculation works when the roles of i and j are swapped. We apply [Fact A.2](#) to the matrices $\sum_{i,j} F^{(i,j)} = \sum_{k=1}^K (\sum_{i,j} F_k^{(i,j)})$ to obtain

$$\left(\sum_{i,j} F^{(i,j)\dagger} \right) \cdot \left(\sum_{i,j} F^{(i,j)} \right) \preceq K \cdot \sum_{k=1}^K \left(\sum_{i,j} F_k^{(i,j)\dagger} \right) \cdot \left(\sum_{i,j} F_k^{(i,j)} \right) \preceq K^2 \cdot \mathbb{I}.$$

□

Proof of [Lemma 3.6](#). Define $M \times M$ matrices $P^{(i)}, Q^{(i)}$ as follows for $i \in [N]$.

$$P^{(i)} = (\mathbb{I} - 2|i\rangle\langle i|) \otimes \mathbb{I}$$

$$Q^{(i)} = |i\rangle\langle i| \otimes \mathbb{I}$$

Observe that $\|Q^{(i)}\|_{\text{op}}, \|P^{(i)}\|_{\text{op}} \leq 1$ for all $i \in [N]$, furthermore,

$$Q^{(i)} = \frac{1}{2} (\mathbb{I} - P^{(i)}). \quad (38)$$

Secondly, the $Q^{(i)}$ are orthogonal, i.e., for all $i, i' \in [N]$, we have

$$Q^{(i)} \cdot Q^{(i')\dagger} = Q^{(i')\dagger} \cdot Q^{(i)} = 0 \quad \text{if } i \neq i'. \quad (39)$$

Finally, we have

$$\sum_i Q^{(i)} = \sum_i Q^{(i)\dagger} \cdot Q^{(i)} = \sum_i Q^{(i)} \cdot Q^{(i)\dagger} = \mathbb{I}. \quad (40)$$

Let $P^{(i_{t_1}, i_{t_2})} = P^{(i_{t_1})} \cdot P^{(i_{t_2})}$. Let $\tilde{P}_t^{(i_{t_1})} := U_t \cdot P^{(i_{t_1})}$ if $t \in [t_1, d-1]$ and $\tilde{P}_t^{(i_{t_1}, i_{t_2})} := U_t \cdot P^{(i_{t_1}, i_{t_2})}$ for $t \in [t_2, d-1]$. Let $\tilde{P}_d^{(i_{t_1})} = \tilde{P}_d^{(i_{t_1}, i_{t_2})} = U_d$. Observe that

$$U = U_{[1, t_1]} \cdot \sum_{i_{t_1}, i_{t_2} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \cdot Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, d]}^{(i_{t_1}, i_{t_2})}$$

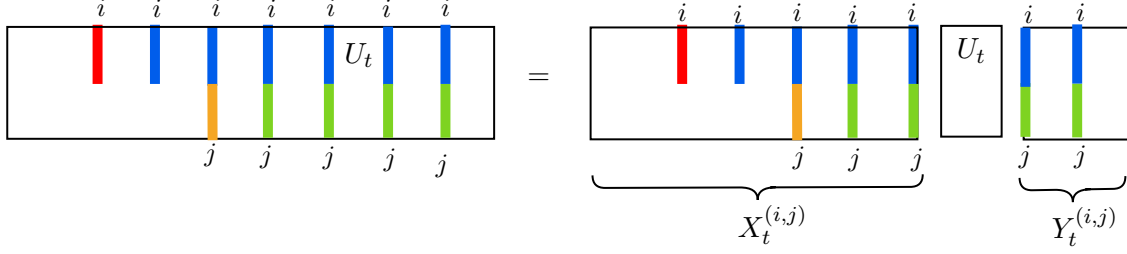


Figure 10: Our goal is to show that the Frobenius norm of the above matrix (summed over i, j) is at most the Frobenius norm of any U_t appearing in this sequence (up to $\text{poly}(d)$ factors). To do so, we split the decomposition into the part before t (call it $X^{(i,j)}$) and the part after t (call it $Y^{(i,j)}$) and our goal then becomes to show that $\left\| \sum_{i,j} X^{(i,j)} U_t Y^{(i,j)} \right\|_{\text{frob}} \leq \text{poly}(d) \cdot \|U_t\|_{\text{frob}}$.

This is depicted in [Figure 9](#). Let us fix any $t \in [t_2, d]$. We will now show that $\|U\|_{\text{frob}} \leq O(d^4) \cdot \|U_t\|_{\text{frob}}$. To do so, we will split this quantity around U_t , as depicted in [Figure 10](#). To do so, we use the fact that $\tilde{P}_t^{(i_{t_1}, i_{t_2})} = U_t \cdot P^{(i_{t_1}, i_{t_2})}$ to conclude that

$$U = U_{[1, t_1]} \cdot \sum_{i_{t_1}, i_{t_2} \in [N]} \underbrace{Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \cdot Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, t]}^{(i_{t_1}, i_{t_2})}}_{X_t^{(i_{t_1}, i_{t_2})}} \cdot U_t \cdot \underbrace{P^{(i_{t_1}, i_{t_2})} \cdot \tilde{P}_{(t, d]}^{(i_{t_1}, i_{t_2})}}_{Y_t^{(i_{t_1}, i_{t_2})}}$$

Since $\|U_{[1, t_1]}\|_{\text{op}} \leq 1$, we have

$$\|U\|_{\text{frob}} \leq \left\| \sum_{i_{t_1}, i_{t_2} \in [N]} X_t^{(i_{t_1}, i_{t_2})} \cdot U_t \cdot Y_t^{(i_{t_1}, i_{t_2})} \right\|_{\text{frob}}. \quad (41)$$

This brings us to the form where we wish to apply [Lemma A.1](#), but the various $Y_t^{(i_{t_1}, i_{t_2})}$ unfortunately don't satisfy hypothesis of [Lemma A.1](#). To get around this, we are going to use a telescoping sum to expand $Y_t^{(i_{t_1}, i_{t_2})}$ into terms that do satisfy the hypothesis, furthermore, the number of terms is small, $\text{poly}(d)$.

For ease of notation, let $\tilde{R}_{t'}^{(i,j)} := P^{(i,j)} \cdot U_{t'}$ for $t' \in (t, d]$ and let $\tilde{R}_{t'}^{(i)} := P^{(i)} \cdot U_{t'}$ for $t' \in (t, d]$ and observe that $Y_t^{(i_{t_1}, i_{t_2})} \triangleq \tilde{R}_{(t, d]}^{(i_{t_1}, i_{t_2})}$. For $r, r' \in (t, d]$, define

$$\begin{aligned} U' &:= U_{(t, d]} \\ W_r^{(i)} &:= \tilde{R}_{(t, r)}^{(i)} \cdot Q^{(i)} \cdot U_{[r, d]} \\ W_{r, r'}^{(i, j)} &:= \begin{cases} \tilde{R}_{(t, r')}^{(i, j)} \cdot Q^{(j)} \cdot \tilde{R}_{[r', r]}^{(i)} \cdot Q^{(i)} \cdot U_{[r, d]} & \text{if } r > r' \\ \tilde{R}_{(t, r)}^{(i, j)} \cdot Q^{(i)} \cdot \tilde{R}_{[r, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', d]} & \text{if } r \leq r' \end{cases} \end{aligned} \quad (42)$$

By [Equation \(38\)](#) and the telescoping argument depicted in [Figure 11](#), for all $i_{t_1}, i_{t_2} \in [N]$,

$$Y_t^{(i_{t_1}, i_{t_2})} = U' - 2 \sum_{r \in (t, d]} W_r^{(i_{t_1})} - 2 \sum_{r' \in (t, d]} W_{r'}^{(i_{t_2})} + 4 \sum_{r, r' \in (t, d]} W_{r, r'}^{(i_{t_1}, i_{t_2})}.$$

Figure 11: We express $Y_t^{(i,j)}$ by a telescoping sum using Equation (38). Firstly, we replace occurrences of $P^{(i)}$ by \mathbb{I} from right to left, producing the telescoping expansion in the top. Then, for each of the resulting terms, we replace occurrences of $P^{(j)}$ by \mathbb{I} from right to left, producing the telescoping expansion in the bottom.

Substituting the above in Equation (41), we have

$$\begin{aligned}
\|U\|_{\text{frob}} &\leq \left\| \left(\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})} \right) \cdot U_t \cdot U_{(t,d]} \right\|_{\text{frob}} + O(d) \cdot \max_{r \in (t,d]} \left\| \sum_{i_{t_1}} \left(\sum_{i_{t_2}} X_t^{(i_{t_1}, i_{t_2})} \right) \cdot U_t \cdot W_r^{(i_{t_1})} \right\|_{\text{frob}} \\
&+ O(d) \cdot \max_{r' \in (t,d]} \left\| \sum_{i_{t_2}} \left(\sum_{i_{t_1}} X_t^{(i_{t_1}, i_{t_2})} \right) \cdot U_t \cdot W_{r'}^{(i_{t_2})} \right\|_{\text{frob}} \\
&+ O(d^2) \cdot \max_{r, r' \in (t,d]} \left\| \sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})} \cdot U_t \cdot W_{r,r'}^{(i_{t_1}, i_{t_2})} \right\|_{\text{frob}}.
\end{aligned} \tag{43}$$

We will now use Lemma A.1 to control each term in the R.H.S. above.

Claim A.4. For all $r, r' \in (t, d]$, the matrices $W_r^{(i)}, W_{r,r'}^{(i,j)}$ as in Equation (42) satisfy the hypothesis of Lemma A.1, playing the role of Y' 's, i.e.,

$$\sum_i W_r^{(i)\dagger} \cdot W_r^{(i)} \preceq \mathbb{I}$$

$$\sum_{i,j} W_{r,r'}^{(i,j)\dagger} \cdot W_{r,r'}^{(i,j)} \preceq \mathbb{I}$$

Proof of Claim A.4. Recalling the definition of $W_r^{(i)}, W_{r,r'}^{(i,j)}$, we see that

$$\sum_i W_r^{(i)\dagger} \cdot W_r^{(i)} = \sum_i U_{[r,d]}^\dagger \cdot Q^{(i)\dagger} \cdot \tilde{R}_{(t,r)}^{(i)\dagger} \cdot \tilde{R}_{(t,r)}^{(i)} \cdot Q^{(i)} \cdot U_{[r,d]}$$

$$\begin{aligned}
&\preceq \sum_i U_{[r,d]}^\dagger \cdot Q^{(i)\dagger} \cdot Q^{(i)} \cdot U_{[r,d]} && \text{(since } \|\tilde{R}_{(t,r)}^{(i)}\|_{\text{op}} \leq 1) \\
&= U_{[r,d]}^\dagger \cdot U_{[r,d]} \preceq \mathbb{I}. && \text{(by Equation (40) applied to } Q^{(i)})
\end{aligned}$$

Similarly we show the desired inequality for $r \leq r'$ and the proof for $r > r'$ is identical. Consider

$$\begin{aligned}
\sum_{i,j} W_{r,r'}^{(i,j)\dagger} \cdot W_{r,r'}^{(i,j)} &= \sum_{i,j} U_{[r',d]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{R}_{[r,r']}^{(j)\dagger} \cdot Q^{(i)\dagger} \cdot \tilde{R}_{(t,r)}^{(i,j)\dagger} \cdot \tilde{R}_{(t,r)}^{(i,j)} \cdot Q^{(i)} \cdot \tilde{R}_{[r,r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r',d]} \\
&\preceq \sum_{i,j} U_{[r',d]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{R}_{[r,r']}^{(j)\dagger} \cdot Q^{(i)\dagger} \cdot Q^{(i)} \cdot \tilde{R}_{[r,r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r',d]} \\
&&& \text{(since } \|\tilde{R}_{(t,r)}^{(i,j)}\|_{\text{op}} \leq 1) \\
&\preceq \sum_j U_{[r',d]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{R}_{[r,r']}^{(j)\dagger} \cdot \tilde{R}_{[r,r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r',d]} \\
&&& \text{(by Equation (40) applied to } Q^{(i)}) \\
&\preceq \sum_j U_{[r',d]}^\dagger \cdot Q^{(j)\dagger} \cdot Q^{(j)} \cdot U_{[r',d]} && \text{(since } \|\tilde{R}_{[r,r']}^{(j)}\|_{\text{op}} \leq 1) \\
&\preceq U_{[r',d]}^\dagger \cdot U_{[r',d]} \preceq \mathbb{I}. && \text{(by Equation (40) applied to } Q^{(j)})
\end{aligned}$$

□

Claim A.5. *Let*

$$X_t^{(i_{t_1}, i_{t_2})} \triangleq Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \cdot Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, t]}^{(i_{t_1}, i_{t_2})}.$$

Then, $\{X_t^{(i_{t_1}, i_{t_2})}\}_{i_{t_1}, i_{t_2}}$, $\{\sum_{i_{t_2}} X_t^{(i_{t_1}, i_{t_2})}\}_{i_{t_1}}$, $\{\sum_{i_{t_1}} X_t^{(i_{t_1}, i_{t_2})}\}_{i_{t_2}}$, and $\{\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})}\}$ all satisfy the hypothesis of [Lemma A.1](#), playing the role of X^l 's with $a = O(d^2)$, i.e.,

$$\begin{aligned}
&\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})\dagger} \cdot X_t^{(i_{t_1}, i_{t_2})} \preceq O(d^4) \cdot \mathbb{I} \\
&\sum_{i_{t_1}} \left(\sum_{i_{t_2}} X_t^{(i_{t_1}, i_{t_2})\dagger} \right) \cdot \left(\sum_{i_{t_2}} X_t^{(i_{t_1}, i_{t_2})} \right) \preceq O(d^4) \cdot \mathbb{I} \\
&\sum_{i_{t_2}} \left(\sum_{i_{t_1}} X_t^{(i_{t_1}, i_{t_2})\dagger} \right) \cdot \left(\sum_{i_{t_1}} X_t^{(i_{t_1}, i_{t_2})} \right) \preceq O(d^4) \cdot \mathbb{I} \\
&\left(\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})\dagger} \right) \cdot \left(\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})} \right) \preceq O(d^4) \cdot \mathbb{I}
\end{aligned} \tag{44}$$

Assuming the above claim, the proof is immediate. Applying [Lemma A.1](#) on each term in [Equation \(43\)](#) and using the bounds from [Claim A.5](#) and [Claim A.4](#) would show that

$$\|U\|_{\text{frob}} \leq O(d^2) \cdot O(d^2) \cdot \min_{t \in [t_2, d]} \|U_t\|_{\text{frob}}. \tag{45}$$

Before proving [Claim A.5](#), we proceed to analyze the other cases. Now suppose $t = d$ or $t \in [1, t_1)$. Then using the fact that $\tilde{P}_d^{(i_{t_1}, i_{t_2})} = U_d$, we get

$$\|U\|_{\text{frob}} = \left\| U_{[1, t_1)} \cdot \sum_{i_{t_1}, i_{t_2} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \cdot Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, d-1]}^{(i_{t_1}, i_{t_2})} \cdot U_d \right\|_{\text{frob}}$$

$$\begin{aligned}
&\leq \left\| \sum_{i_{t_1}, i_{t_2} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \cdot Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, d-1]}^{(i_{t_1}, i_{t_2})} \right\|_{\text{op}} \cdot \min \left(\|U_{[1, t_1]}\|_{\text{frob}}, \|U_d\|_{\text{frob}} \right) \\
&\quad \text{(since } \|U_d\|_{\text{op}}, \|U_{[1, t_1]}\|_{\text{op}} \leq 1 \text{ and by Fact 2.2)} \\
&\leq \min \left(\|U_{[1, t_1]}\|_{\text{frob}}, \|U_d\|_{\text{frob}} \right) \cdot O(d^2)
\end{aligned}$$

where the last bound on the operator norm follows from a calculation similar to [Claim A.5](#) where we obtained bounds of $O(d^2)$ on the operator norms of $\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})}$. This proves that

$$\|U\|_{\text{frob}} \leq O(d^2) \cdot \min_{t \in [1, t_1] \cup \{d\}} \|U_t\|_{\text{frob}} \quad (46)$$

Finally, we consider any $t \in [t_1, t_2]$. Observe that

$$\begin{aligned}
\|U\|_{\text{frob}}^2 &\leq \left\| \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \cdot \left(\sum_{i_{t_2} \in [N]} Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, d]}^{(i_{t_1}, i_{t_2})} \right) \right\|_{\text{frob}}^2 && \text{(since } \|U_{[1, t_1]}\|_{\text{op}} \leq 1) \\
&= \sum_{i_{t_1} \in [N]} \left\| Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \cdot \left(\sum_{i_{t_2} \in [N]} Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, d]}^{(i_{t_1}, i_{t_2})} \right) \right\|_{\text{frob}}^2 && \text{(by Equation (39))} \\
&\leq \sum_{i_{t_1} \in [N]} \left\| Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \right\|_{\text{frob}}^2 \cdot \left\| \sum_{i_{t_2} \in [N]} Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, d]}^{(i_{t_1}, i_{t_2})} \right\|_{\text{op}}^2 && \text{(by Fact 2.2)}
\end{aligned}$$

For any $i_{t_1} \in [N]$, we see that the matrix $\sum_{i_{t_2} \in [N]} Q^{(i_{t_2})} \cdot \tilde{P}_{[t_2, d]}^{(i_{t_1}, i_{t_2})}$ is precisely obtained by applying [Lemma 3.2](#) with $\gamma = 2$ to the matrices $U_{t_2} \cdot P^{(i_{t_1})}, U_{t_2+1} \cdot P^{(i_{t_1})}, \dots, U_{d-1} \cdot P^{(i_{t_1})}, U_d$ and therefore, its operator norm is at most $O(d)$. Therefore, we get

$$\begin{aligned}
\|U\|_{\text{frob}}^2 &\leq O(d^2) \sum_{i_{t_1} \in [N]} \left\| Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \right\|_{\text{frob}}^2 \\
&= O(d^2) \left\| \sum_{i_{t_1} \in [N]} Q^{(i_{t_1})} \cdot \tilde{P}_{[t_1, t_2]}^{(i_{t_1})} \right\|_{\text{frob}}^2 && \text{(by Equation (39))} \\
&\leq O(d^4) \min_{t \in [t_1, t_2]} (\|U_t\|_{\text{frob}}^2) \\
&\quad \text{(by an application of Lemma 3.2 to the matrices } U_{t_1}, \dots, U_{t_2-1} \text{ with } \gamma = 2.)
\end{aligned}$$

This shows that

$$\|U\|_{\text{frob}} \leq O(d^2) \min_{t \in [t_1, t_2]} \|U_t\|_{\text{frob}} \quad (47)$$

Combining [Equations \(45\)](#) to [\(47\)](#) gives us the desired bound of

$$\|U\|_{\text{frob}} \leq O(d^4) \cdot \min_t \|U_t\|_{\text{frob}}.$$

All that remains is to prove [Claim A.5](#).

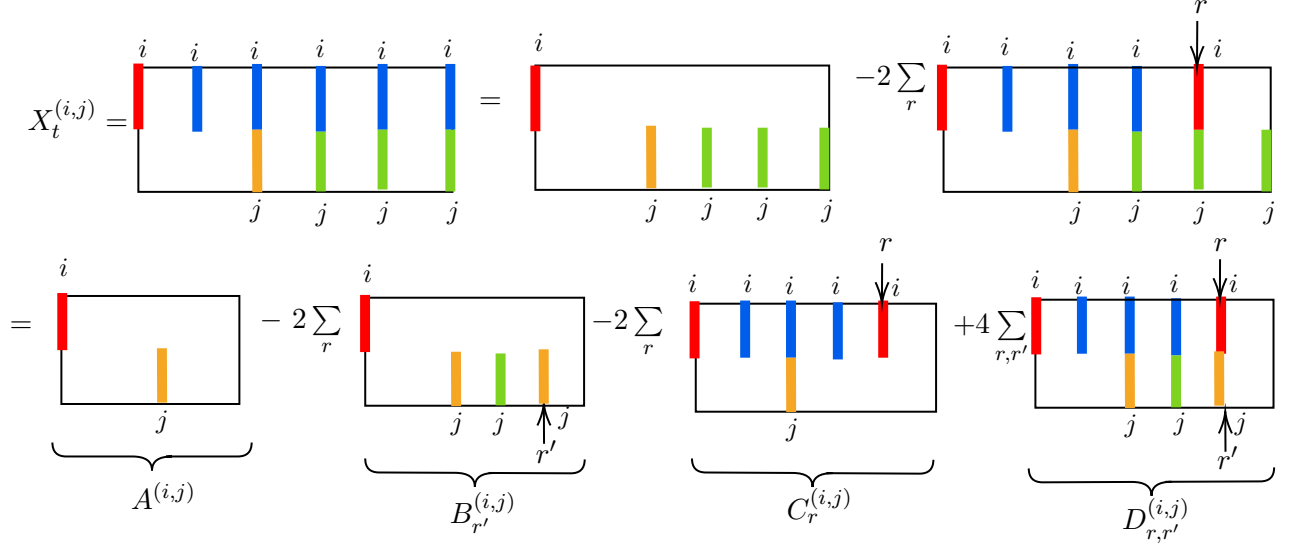


Figure 12: We express $X_t^{(i,j)}$ by a telescoping sum using Equation (38). We leave $Q^{(i)}$ and $Q^{(j)}$ untouched. Firstly, we replace occurrences of $P^{(i)}$ by \mathbb{I} from right to left, producing the telescoping expansion in the top. Then, for each of the resulting terms, we replace occurrences of $P^{(j)}$ by \mathbb{I} from right to left, producing the telescoping expansion in the bottom.

Proof of Claim A.5. While it can be shown that $\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})} \cdot X_t^{(i_{t_1}, i_{t_2})\dagger} \preceq \mathbb{I}$ using an argument similar to what we did for $W_{r,r'}^{(i_{t_1}, i_{t_2})}$, what we wish for is $\sum_{i_{t_1}, i_{t_2}} X_t^{(i_{t_1}, i_{t_2})\dagger} \cdot X_t^{(i_{t_1}, i_{t_2})} \preceq \mathbb{I}$. To show this, we are again going to use a telescoping sum to express $X_t^{(i_{t_1}, i_{t_2})}$ as a sum of $O(d^2)$ matrices which satisfy the hypothesis of Lemma A.3. We will then use Lemma A.3 to conclude that $X_t^{(i_{t_1}, i_{t_2})}$ satisfies itself the hypothesis of Lemma A.1 with $a = O(d^2)$. To express $X^{(i_{t_1}, i_{t_2})}$ as desired, we will apply the telescoping identity on the terms $\tilde{P}_{[t_1, t_2]}^{(i_{t_1})}$ and $\tilde{P}_{[t_2, t]}^{(i_{t_1}, i_{t_2})}$. Define matrices as follows. For $i, j \in [N]$, and $r \in (t_1, t), r' \in (t_2, t)$, let

$$\begin{aligned}
A^{(i,j)} &:= Q^{(i)} \cdot U_{[t_1, t_2]} \cdot Q^{(j)} \cdot U_{[t_2, t]} \\
B_{r'}^{(i,j)} &:= Q^{(i)} \cdot U_{[t_1, t_2]} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', t]} \\
C_r^{(i,j)} &:= \begin{cases} Q^{(i)} \cdot \tilde{P}_{[t_1, r]}^{(i)} \cdot Q^{(i)} \cdot U_{[r, t_2]} \cdot Q^{(j)} \cdot U_{[t_2, t]} & \text{if } r \leq t_2 \\ Q^{(i)} \cdot \tilde{P}_{[t_1, t_2]}^{(i)} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r]}^{(i)} \cdot Q^{(i)} \cdot U_{[r, t]} & \text{if } r > t_2 \end{cases} \\
D_{r,r'}^{(i,j)} &:= \begin{cases} Q^{(i)} \cdot \tilde{P}_{[t_1, r]}^{(i)} \cdot Q^{(i)} \cdot U_{[r, t_2]} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', t]} & \text{if } r \leq t_2 \\ Q^{(i)} \cdot \tilde{P}_{[t_1, t_2]}^{(i)} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r]}^{(i,j)} \cdot Q^{(i)} \cdot \tilde{P}_{[r, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', t]} & \text{if } t_2 < r \leq r' \\ Q^{(i)} \cdot \tilde{P}_{[t_1, t_2]}^{(i)} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(i,j)} \cdot Q^{(j)} \cdot \tilde{P}_{[r', r]}^{(i)} \cdot Q^{(i)} \cdot U_{[r, t]} & \text{if } r > r' \end{cases}
\end{aligned} \tag{48}$$

Again, by the telescoping expansion described in Figure 12, we have

$$X_t^{(i_{t_1}, i_{t_2})} = A^{(i_{t_1}, i_{t_2})} - 2 \sum_{r' \in (t_2, t)} B_{r'}^{(i_{t_1}, i_{t_2})} - 2 \sum_{r \in (t_1, t)} C_r^{(i_{t_1}, i_{t_2})} + 4 \sum_{\substack{r' \in (t_2, t) \\ r \in (t_1, t)}} D_{r,r'}^{(i_{t_1}, i_{t_2})}.$$

We will now show that the matrices in Equation (48) satisfy the hypothesis of Lemma A.3. This

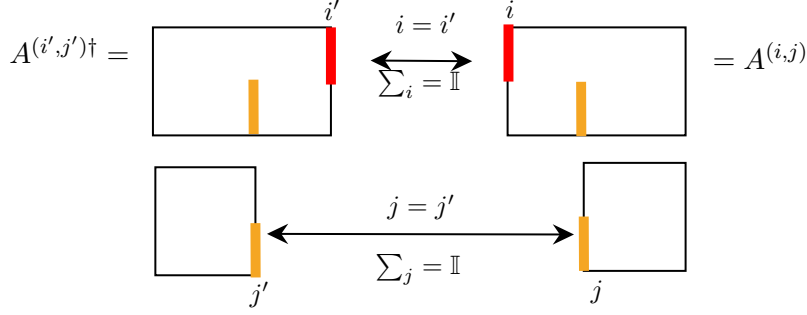


Figure 13: How to show that $A^{(i,j)}$ satisfies the hypothesis of [Lemma A.3](#).

is done by a case by case analysis. For ease of notation let $i_{t_1} = i$ and $i_{t_2} = j$. We now show the desired bound for $A^{(i,j)}$, $B_{r,r'}^{(i,j)}$, $C_r^{(i,j)}$ and $D_{r,r'}^{(i,j)}$.

Analysis for $A^{(i,j)}$. Consider $A^{(i,j)} = Q^{(i)} \cdot U_{[t_1,t_2]} \cdot Q^{(j)} \cdot U_{[t_2,t]}$. This is depicted in the top right of [Figure 13](#). We wish to take $\sum (\sum A^{(i,j)\dagger}) \cdot (\sum A^{(i,j)})$, where we intentionally omit the identity of the indices being summed over in the inner summation – it could be the empty summation, the summation over i , the summation over j , or the summation over both i and j , and the outer summation is always understood to be over the remaining variables, if any. For all the resulting expressions, we wish to show that the result is $\preceq \mathbb{I}$. In all these cases, the expression consists of a sum over possibly i', j' of the \dagger terms and a sum over possibly i, j of the regular terms. We first present the high level intuition and then the calculation. As before, we observe that the inner terms consist of $Q^{(i')\dagger} \cdot Q^{(i)}$ and by the orthogonality of the $Q^{(i)}$, the cross terms corresponding to $i \neq i'$ to vanish and we can pull the \sum_i outside. Furthermore, we can sum over i to obtain the identity matrix, as the terms before and after do not depend on i . Then, we can strip away the matrices between $Q^{(j')\dagger}$ and $Q^{(j)}$, as these only have operator norm at most 1 and do not depend on j, j' . We then use the orthogonality of the $Q^{(j)}$ to argue that the cross terms corresponding $j \neq j'$ vanish and we can pull the \sum_j outside. Furthermore, we can sum over j to obtain the identity matrix, as the terms before and after do not depend on j . This process is depicted in [Figure 13](#) and is mathematically described below.

$$\begin{aligned}
& \sum (\sum A^{(i,j)}) \cdot (\sum A^{(i,j)}) \\
&= \sum_i \underbrace{\left(\sum U_{[t_2,t]}^\dagger \cdot Q^{(j)\dagger} \cdot U_{[t_1,t_2]}^\dagger \right)}_{\text{independent of } i} \cdot \underbrace{Q^{(i)\dagger} \cdot Q^{(i)}}_{\text{sum to } \mathbb{I}} \cdot \underbrace{\left(\sum U_{[t_1,t_2]} \cdot Q^{(j)} \cdot U_{[t_2,t]} \right)}_{\text{independent of } i} \\
& \hspace{15em} \text{(Equation (39) applied to } Q^{(i)}) \\
&= \sum \left(\sum U_{[t_2,t]}^\dagger \cdot Q^{(j)\dagger} \right) \cdot U_{[t_1,t_2]}^\dagger \cdot U_{[t_1,t_2]} \cdot \left(\sum Q^{(j)} \cdot U_{[t_2,t]} \right) \\
&\preceq \sum \left(\sum U_{[t_2,t]}^\dagger \cdot Q^{(j)\dagger} \right) \cdot \left(\sum Q^{(j)} \cdot U_{[t_2,t]} \right) \\
&= \sum U_{[t_2,t]}^\dagger \cdot \underbrace{Q^{(j)\dagger} \cdot Q^{(j)}}_{\text{sum to } \mathbb{I}} \cdot U_{[t_2,t]} = U_{[t_2,t]}^\dagger \cdot U_{[t_2,t]} \preceq \mathbb{I}. \hspace{2em} \text{(Equation (39) applied to } Q^{(j)})
\end{aligned}$$

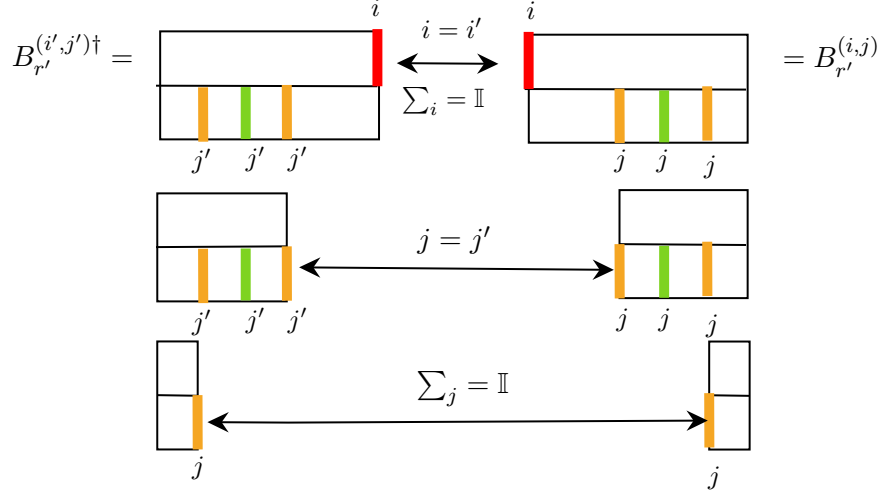


Figure 14: How to show that $B_{r'}^{(i,j)}$ satisfies the hypothesis of [Lemma A.3](#).

Analysis for $B_{r'}^{(i,j)}$. Consider $B_{r'}^{(i,j)} = Q^{(i)} \cdot U_{[t_1, t_2]} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', t]}$. This is depicted in the top right of [Figure 14](#). We wish to control $\sum \left(\sum B_{r'}^{(i,j)\dagger} \right) \cdot \left(\sum B_{r'}^{(i,j)} \right)$. The strategy is as follows. As before, we observe that the inner terms consist of $Q^{(i')\dagger} \cdot Q^{(i)}$ and by the orthogonality of the $Q^{(i)}$, the cross terms corresponding to $i \neq i'$ to vanish and we can pull the \sum_i outside. Furthermore, we can sum over i to obtain the identity matrix, as the terms before and after do not depend on i . Then, we can strip away the matrices between the closest pair of $Q^{(j')\dagger}$ and $Q^{(j)}$, as these only have operator norm at most 1 and are independent of j, j' . We then use the orthogonality of the $Q^{(j)}$ to argue that the cross terms corresponding $j \neq j'$ vanish and we can pull the \sum_j outside. Finally, we strip away all the terms except for the outermost $Q^{(j)\dagger}$ and $Q^{(j)}$, which we can do as these matrices have operator norm at most one. Finally, we can sum over j to obtain the identity matrix, as the terms before and after do not depend on j . This process is depicted in [Figure 14](#) and is mathematically described below.

$$\begin{aligned}
& \sum \left(\sum B_{r'}^{(i,j)\dagger} \right) \cdot \left(\sum B_{r'}^{(i,j)} \right) \\
&= \sum_i \underbrace{\left(\sum U_{[r', t]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{P}_{[t_2, r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \cdot U_{[t_1, t_2]}^\dagger \right)}_{\text{independent of } i} \cdot \underbrace{Q^{(i)\dagger} \cdot Q^{(i)}}_{\text{sum to } \mathbb{I}} \\
& \underbrace{\left(\sum U_{[t_1, t_2]} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', t]} \right)}_{\text{independent of } i} \quad \text{(Equation (39) applied to } Q^{(i)} \text{)} \\
&= \sum \left(\sum U_{[r', t]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{P}_{[t_2, r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \right) \cdot U_{[t_1, t_2]}^\dagger \cdot U_{[t_1, t_2]} \cdot \left(\sum Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', t]} \right) \\
& \quad \text{(Equation (40) applied to } Q^{(i)} \text{)} \\
&\preceq \sum \left(\sum U_{[r', t]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{P}_{[t_2, r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \right) \cdot \left(\sum Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r', t]} \right) \\
&= \sum_j U_{[r', t]}^\dagger \cdot Q^{(j)\dagger} \cdot \underbrace{\tilde{P}_{[t_2, r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2, r']}^{(j)}}_{\text{norm } \leq 1} \cdot Q^{(j)} \cdot U_{[r', t]} \quad \text{(Equation (39) applied to } Q^{(i)} \text{)} \\
&\preceq \sum_j U_{[r', t]}^\dagger \cdot \underbrace{Q^{(j)\dagger} \cdot Q^{(j)}}_{\text{sum to } \mathbb{I}} \cdot U_{[r', t]} \preceq \sum_j U_{[r', t]}^\dagger \cdot U_{[r', t]} \preceq \mathbb{I}. \quad \text{(Equation (40) applied to } Q^{(j)} \text{)}
\end{aligned}$$

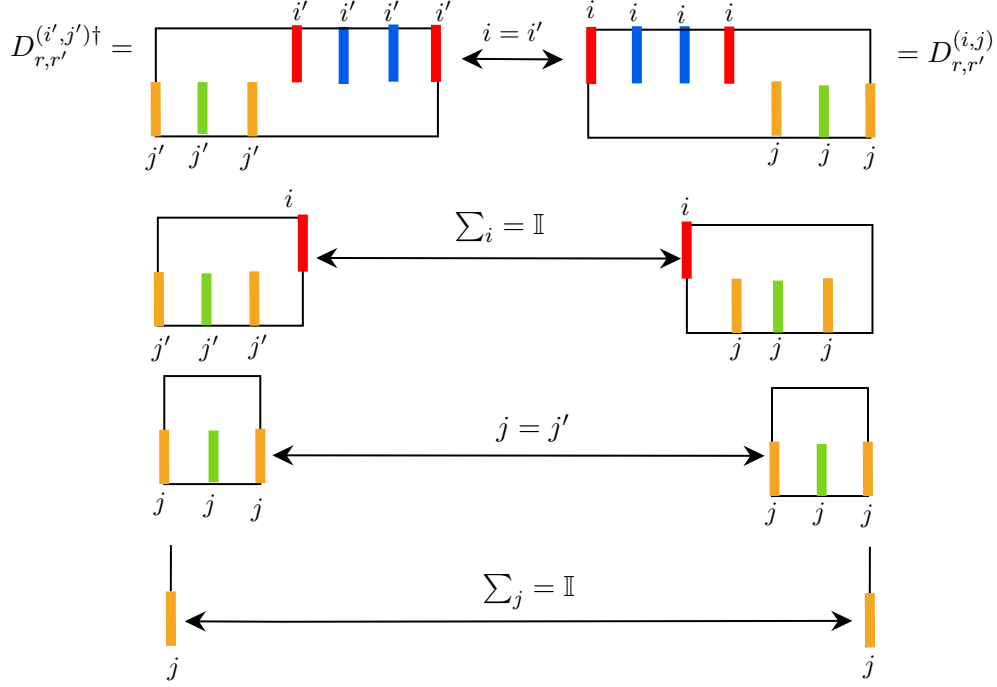


Figure 16: How to show that $D_{r,r'}^{(i,j)}$ satisfies the hypothesis of Lemma A.3 for $r \leq t_2$.

$$\begin{aligned}
& \cdot \underbrace{\left(Q^{(i)} \cdot \tilde{P}_{[t_1,r]}^{(i)} \cdot Q^{(i)} \cdot U_{[r,t_2]} \right)}_{\text{independent of } j} \cdot \left(\sum Q^{(j)} \cdot \tilde{P}_{[t_2,r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r',t]} \right) \\
& \leq \sum_i \underbrace{\left(\sum U_{[r',t]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{P}_{[t_2,r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \right)}_{\text{independent of } i} \cdot U_{[r,t_2]}^\dagger \cdot \underbrace{Q^{(i)\dagger} \cdot Q^{(i)}}_{\text{sum to } \mathbb{I}} \cdot U_{[r,t_2]} \\
& \cdot \underbrace{\left(\sum Q^{(j)} \cdot \tilde{P}_{[t_2,r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r',t]} \right)}_{\text{independent of } i} \\
& = \sum \left(\sum U_{[r',t]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{P}_{[t_2,r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \right) \cdot U_{[r,t_2]}^\dagger \cdot U_{[r,t_2]} \cdot \left(\sum Q^{(j)} \cdot \tilde{P}_{[t_2,r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r',t]} \right) \\
& \hspace{15em} \text{(Equation (40) applied to } Q^{(i)}) \\
& \leq \sum \left(\sum U_{[r',t]}^\dagger \cdot Q^{(j)\dagger} \cdot \tilde{P}_{[t_2,r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \right) \cdot \left(\sum Q^{(j)} \cdot \tilde{P}_{[t_2,r']}^{(j)} \cdot Q^{(j)} \cdot U_{[r',t]} \right) \\
& = \sum_j U_{[r',t]}^\dagger \cdot Q^{(j)\dagger} \cdot \underbrace{\tilde{P}_{[t_2,r']}^{(j)\dagger} \cdot Q^{(j)\dagger} \cdot Q^{(j)} \cdot \tilde{P}_{[t_2,r']}^{(j)}}_{\text{norm } \leq 1} \cdot Q^{(j)} \cdot U_{[r',t]} \hspace{2em} \text{(Equation (39) applied to } Q^{(j)}) \\
& \leq \sum_j U_{[r',t]}^\dagger \cdot \underbrace{Q^{(j)\dagger} \cdot Q^{(j)}}_{\text{sum to } \mathbb{I}} \cdot U_{[r',t]} \leq U_{[r',t]}^\dagger \cdot U_{[r',t]} \leq \mathbb{I}. \hspace{2em} \text{(Equation (40) applied to } Q^{(j)})
\end{aligned}$$

The analysis for the other cases is similar, and for completeness, we depict the strategy to handle $t_2 < r \leq r'$ in Figure 17 and $r > r'$ in Figure 18. This completes the proof of Claim A.5.

□

□

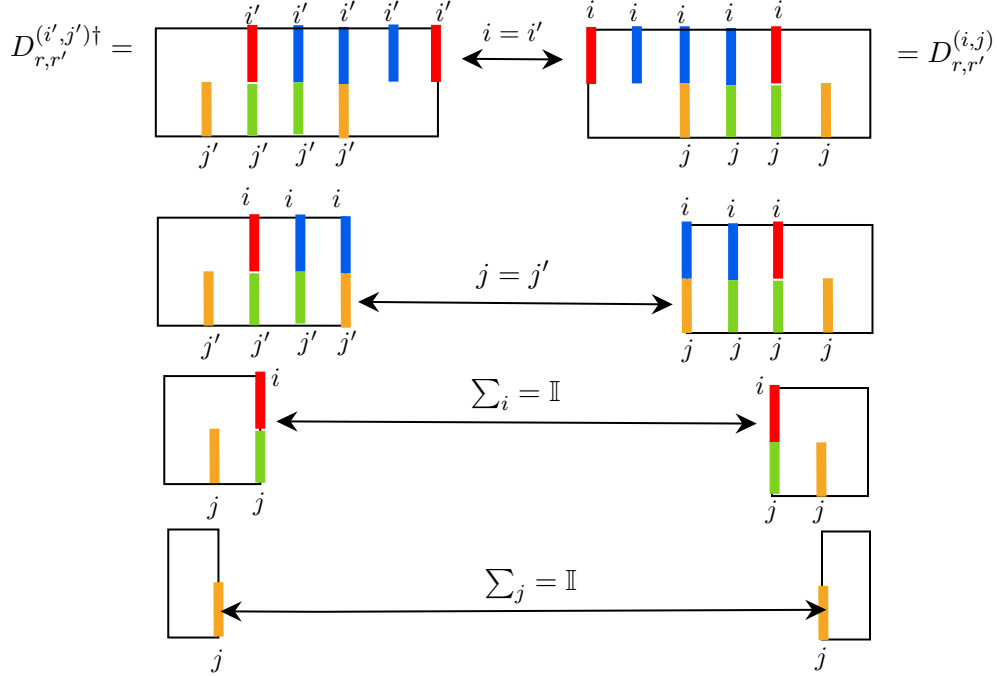


Figure 17: How to show that $D_{r,r'}^{(i,j)}$ satisfies the hypothesis of Lemma A.3 for $t_2 < r \leq r'$.

A.3 Quantum Algorithms with Classical Pre-Processing.

In this section, we prove a variant of Theorem 1.5 and Theorem 1.7 in a more general setting of algorithms that can perform classical pre-processing. We now describe this model more formally. A d -query DQC_k (respectively BQP) algorithm with classical pre-processing consists of two phases:

- CLASSICAL PHASE: The algorithm performs d classical queries on clean workspace.
- QUANTUM PHASE: Based on the results, the algorithm chooses a d -query DQC_k (respectively BQP) algorithm to run and returns the output.

Theorem A.6. *Let \mathcal{F} denote the family of acceptance probabilities of a class of algorithms without classical pre-processing. Let $f(x)$ be the acceptance probability of an algorithm with d classical pre-processing queries. Let $\rho \in \{-1, 1, *\}^N$ be any restriction and $\alpha \in [-1, 1]^{\binom{N}{\ell}}$ signs. Then, there exist $f' \in \mathcal{F}$ such that*

$$L_{1,\ell}^\alpha(f|\rho) \leq \sum_{k=0}^{\ell} \binom{d}{\ell-k} \cdot \max_{\alpha'} L_{1,k}^{\alpha'}(f'|\rho),$$

where the maximum is over α' , another family of signs.

Corollary A.7. *Analogues of Theorems 1.5 and 1.7 hold even for algorithms with classical pre-processing.*

Proof of Theorem A.6. We view the classical phase as a decision tree of depth d with 2^d leaves where each leaf y selects an algorithm f_y to run. Furthermore, we view each leaf y as a partial assignment in $\{-1, 1, *\}^N$ where the coordinates that are queried are assigned ± 1 depending on the outcome of the query, and the coordinates not queried are assigned $*$. We use $y^{-1}(*)$ to denote the coordinates of y that are alive. We know that $|y^{-1}(*)| \geq N - d$. This defines a restriction $\rho_y \in \{-1, 1, *\}^N$ of

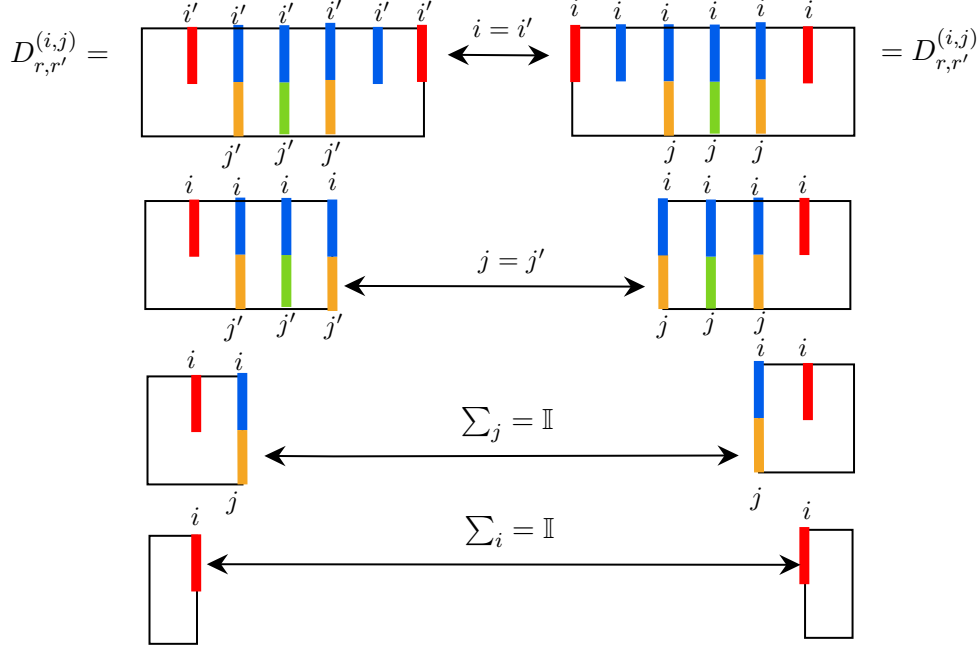


Figure 18: How to show that $D_{r,r'}^{(i,j)}$ satisfies the hypothesis of Lemma A.3 for $r' < r$.

the variables which restricts the i -th coordinate to y_i if $y_i \in \{-1, 1\}$ and leaves it alive otherwise. We can assume that any y that is ever traversed is consistent with ρ . For any such y , let $f_y(x)$ be the acceptance probability of the algorithm chosen conditioned on receiving y in the first stage. Consider:

$$\begin{aligned}
& L_{1,\ell}^\alpha(f|\rho) \\
&= \mathbb{E}_{x \sim \{-1,1\}^N} \left[\sum_{|S|=\ell} \alpha_S \cdot f|_\rho(x) \cdot \chi_S(x) \right] \\
&= \mathbb{E}_{y \text{ consistent with } \rho} \left[\mathbb{E}_{\substack{x \sim \{-1,1\}^N \\ \text{consistent with } y}} \left[\sum_{|S|=\ell} \alpha_S \cdot f_y(\rho(x)) \cdot \chi_S(x) \right] \right] \\
&= \mathbb{E}_{y \text{ consistent with } \rho} \left[\mathbb{E}_{\substack{x \sim \{-1,1\}^N \\ \text{consistent with } y}} \left[\sum_{k=0}^{\ell} \sum_{\substack{S_1 \subseteq y^{-1}(\ast) \\ S_2 \subseteq [N] \setminus y^{-1}(\ast) \\ |S_1|=k, |S_2|=\ell-k}} \alpha_{S_1 \cup S_2} \cdot f_y(\rho(x)) \cdot \chi_{S_1}(x) \cdot \chi_{S_2}(x) \right] \right].
\end{aligned}$$

Fix a leaf y that maximizes the above quantity. Since we are only taking expectations over x consistent with y , we can replace $\chi_{S_2}(x)$ by $\chi_{S_2}(y)$ in the R.H.S. above and similarly, $\rho(x)$ only depends on the variables in S_1 . Once we do this, x is completely free of y and we can replace the expectation of $x \sim \{-1, 1\}^N$ consistent with y by simply $x \sim \{-1, 1\}^N$. We obtain

$$L_{1,\ell}^\alpha(f|\rho) \leq \mathbb{E}_{x \sim \{-1,1\}^N} \left[\sum_{k=0}^{\ell} \sum_{\substack{S_1 \subseteq y^{-1}(\ast) \\ S_2 \subseteq [N] \setminus y^{-1}(\ast) \\ |S_1|=k, |S_2|=\ell-k}} \alpha_{S_1 \cup S_2} \cdot \chi_{S_2}(y) \cdot f_y(\rho(x)) \cdot \chi_{S_1}(x) \right] \quad (49)$$

Since $|\chi_{S_2}(y)| \leq 1$, applying Triangle Inequality gives

$$L_{1,\ell}^\alpha(f|\rho) \leq \sum_{k=0}^{\ell} \sum_{\substack{S_2 \subseteq [N] \setminus y^{-1}(\ast) \\ |S_2|=\ell-k}} \left| \mathbb{E}_{x \sim \{-1,1\}^N} \left[\sum_{\substack{S_1 \subseteq y^{-1}(\ast) \\ |S_1|=k}} \alpha_{S_1 \cup S_2} \cdot f_y(\rho(x)) \cdot \chi_{S_1}(x) \right] \right| \quad (50)$$

Define $\gamma \in [-1,1]^N$ by $\gamma_i = 1$ if $i \in y^{-1}(\ast)$ and 0 otherwise. For any fixed $k \in \{0, \dots, \ell\}$ and $S_2 \subseteq [N] \setminus y^{-1}(\ast)$ of size $\ell - k$, define signs α^{S_2} that are non-zero only for $S_1 \subseteq [N]$ with size k so that

$$\alpha_{S_1}^{S_2} := \alpha_{S_1 \cup S_2} \cdot \chi_{S_1}(\gamma).$$

Observe that $\chi_{S_1}(\gamma) = 1$ if $S_1 \subseteq y^{-1}(\ast)$ and 0 otherwise. Thus,

$$\sum_{\substack{S_1 \subseteq y^{-1}(\ast) \\ |S_1|=k}} \alpha_{S_1 \cup S_2} \cdot \chi_{S_1}(x) = \sum_{\substack{S_1 \subseteq [N] \\ |S_1|=k}} \alpha_{S_1 \cup S_2} \cdot \chi_{S_1}(\gamma) \cdot \chi_{S_1}(x) \triangleq \sum_{\substack{S_1 \subseteq [N] \\ |S_1|=k}} \alpha_{S_1}^{S_2} \cdot \chi_{S_1}(x).$$

Finally, we observe that

$$\mathbb{E}_{x \sim \{-1,1\}^N} \left[\sum_{\substack{S_1 \subseteq [N] \\ |S_1|=k}} \alpha_{S_1}^{S_2} \cdot f_y(\rho(x)) \cdot \chi_{S_1}(x) \right] \triangleq \sum_{\substack{S_1 \subseteq [N] \\ |S_1|=k}} \alpha_{S_1}^{S_2} \cdot \widehat{f_y|_\rho}(S_1) \triangleq L_{1,k}^{\alpha^{S_2}}(f_y|\rho).$$

Substituting this in [Equation \(50\)](#), we get

$$L_{1,\ell}^\alpha(f) \leq \sum_{k=0}^{\ell} \binom{d}{\ell-k} \cdot \max_{\alpha'} L_{1,k}^{\alpha'}(f_y|\rho),$$

where we used the fact that $N - |y^{-1}(\ast)| \leq d$. This completes the proof. \square

Proof of [Corollary A.7](#) from [Theorem A.6](#). Let \mathcal{F} (respectively \mathcal{F}') denote the class of d -query DQC_k algorithms with (respectively without) classical pre-processing. Applying [Theorem A.6](#), we have

$$L_{1,\ell}(\mathcal{F}) \leq \sum_{k=0}^{\ell} \binom{d}{\ell-k} \cdot L_{1,k}^{\alpha'}(f|\rho').$$

We now apply [Theorem 1.5](#) to bound each $L_{1,k}^{\alpha'}(f|\rho')$ and this gives

$$L_{1,\ell}(\mathcal{F}) \leq \sum_{k=0}^{\ell} \binom{d}{\ell-k} \cdot c^k \cdot d^3 \cdot N^{(k-2)/2} \cdot \log(N)^{k-2} \cdot \sqrt{k!} \leq \tilde{O} \left(d^{\ell+3} \cdot N^{(\ell-2)/2} \cdot \sqrt{\ell!} \right)$$

as desired. The proof for BQP algorithms is identical and we obtain a bound of $\tilde{O} \left(d^{\ell+1} \cdot N^{(\ell-1)/2} \cdot \sqrt{\ell!} \right)$. \square

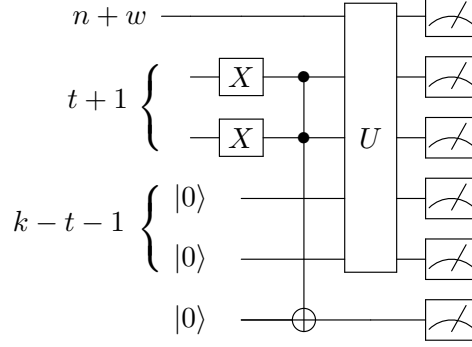


Figure 19: Simulating a DQC_k algorithm by a DQC_{k-t} algorithm.

A.4 Simulating DQC_k algorithms by DQC_{k-t} algorithms.

Claim A.8. *Let $g(x)$ be the bias of a d -query DQC_k algorithm. Then, there is a d -query DQC_{k-t} algorithm whose bias is $g(x) \cdot 2^{-t-1}$.*

Proof of Claim A.8. Given a d -query DQC_k algorithm with $n+w$ noisy bits, consider a DQC_{k-t} algorithm which uses $n+w+t+1$ noisy bits and $k-t$ clean qubits as follows. Firstly, the algorithm applies the X gate to the last $t+1$ noisy qubits and applies a Toffoli controlled on these qubits with the target as the final clean qubit. Then, apply the DQC_k algorithm on the first $n+w$ noisy qubits and the first k clean qubits. Finally, measure the last clean qubit. If it results in an outcome 1, then return the outcome of the DQC_k algorithm, otherwise, return a random bit (by taking an additional noisy qubit for instance).

Observe that this algorithm behaves identically to the original one whenever the $t+1$ noisy qubits are in the all-zeroes state, which happens with probability 2^{-t-1} . In all other cases, the algorithm returns a uniformly random bit. Thus, the bias of the resulting algorithm is $2^{-t-1} \cdot g(x)$. \square

A.5 Acceptance Probability of Quantum Algorithms

DQC_k algorithms.

Proof of Claim 2.10. Consider a d -query DQC_k algorithm and let U_1, \dots, U_{d+1} be the unitary operators of the algorithm and $\mathcal{S} = [NW] \times \{1\}, \mathcal{F} \subseteq [NWK]$ be the set of initial and accepting final states as in Definition 2.9 and Figure 1. The final state of the algorithm can be expressed as a uniform mixture over $I_1 \in \mathcal{S}$ of the pure state $U_{d+1} \cdot (O_x \otimes \mathbb{I}) \cdot U_d \cdots (O_x \otimes \mathbb{I}) \cdot U_1 |I_1\rangle$. Let $\mathcal{F} \subseteq [NWK]$ be the subset of final basis states that is accepted by the algorithm. We can thus express the acceptance probability of the algorithm as an average over $I_1 \in \mathcal{S}$ of

$$\sum_{I_{d+2} \in \mathcal{F}} |\langle I_{d+2} | U_{d+1} \cdot O \cdot U_d \cdots O \cdot U_1 | I_1 \rangle|^2$$

Since there are NW elements in \mathcal{S} , the overall acceptance probability of the algorithm is given by

$$\begin{aligned} f(x) &:= \frac{1}{NW} \sum_{\substack{I_1 \in \mathcal{S} \\ I_{d+2} \in \mathcal{F}}} |\langle I_{d+2} | U_{d+1} \cdot O \cdot U_d \cdots O \cdot U_1 | I_1 \rangle|^2 \\ &= \frac{1}{NW} \sum_{\substack{I_1 \in \mathcal{S} \\ I_{d+2} \in \mathcal{F}}} \langle I_1 | U_1^\dagger \cdot O \cdots U_d^\dagger \cdot O \cdot U_{d+1}^\dagger | I_{d+2} \rangle \cdot \langle I_{d+2} | U_{d+1} \cdot O \cdot U_d \cdots O \cdot U_1 | I_1 \rangle \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{NW} \sum_{\substack{I_1 \in \mathcal{S} \\ I_{d+2} \in \mathcal{F}}} \text{Tr} \left(U_1 |I_1\rangle \langle I_1| U_1^\dagger \cdot O \cdots U_d^\dagger \cdot O \cdot U_{d+1}^\dagger |I_{d+2}\rangle \langle I_{d+2}| U_{d+1} \cdot O \cdot U_d \cdots U_2 \cdot O \right) \\
&= \frac{1}{NW} \text{Tr} \left(U_1 \left(\sum_{I_1 \in \mathcal{S}} |I_1\rangle \langle I_1| \right) U_1^\dagger \cdot O \cdots O \cdot U_{d+1}^\dagger \left(\sum_{I_{d+2} \in \mathcal{F}} |I_{d+2}\rangle \langle I_{d+2}| \right) U_{d+1} \cdot O \cdots U_2 \cdot O \right).
\end{aligned}$$

We will further simplify this expression by introducing $M \times M$ matrices V_1, \dots, V_{2d} as follows. Let $V_1 = \sum_{I_1 \in \mathcal{S}} U_1 |I_1\rangle \langle I_1| U_1^\dagger$. For $t \in [2, d]$, let $V_t := U_t^\dagger$. Let $V_{d+1} = \sum_{I_{d+2} \in \mathcal{F}} U_{d+1}^\dagger |I_{d+2}\rangle \langle I_{d+2}| U_{d+1}$ and for $t \in [d-1]$, let $V_{d+1+t} := U_{d-t+1}$. This allows us to express $f(x)$ as

$$f(x) = (NW)^{-1} \cdot \text{Tr} (V_1 \cdot O \cdot V_2 \cdot O \cdots V_{2d} \cdot O).$$

This gives us the desired expression. Finally we observe that $\|V_t\|_{\text{op}} \leq 1$ for all t , and V_1 is (up to multiplication by unitary matrices) equal to a diagonal matrix with at most $|\mathcal{S}| = NW$ non-zero entries of value 1, hence $\|V_1\|_{\text{frob}} \leq \sqrt{NW}$. \square

$\frac{1}{2}$ BQP algorithms.

Proof of Claim 2.12. Let \mathcal{F} be the accepting pairs of initial and final states of a $\frac{1}{2}$ BQP algorithm and U_1, \dots, U_{d+1} be unitary operators as in Definition 2.11 and Figure 2. For I_1, I_{d+1} , we use $F_{I_1, I_{d+1}}$ to denote 1 when $(I_1, I_{d+1}) \in \mathcal{F}$ and 0 otherwise. It is fairly straightforward to see that the acceptance probability $f(x)$ of the algorithm is given by

$$\begin{aligned}
f(x) &:= M^{-1} \sum_{I_1, I_{d+2} \in [M]} F_{I_1, I_{d+2}} \cdot |\langle I_{d+2} | U_{d+1} \cdot O \cdot U_d \cdots O \cdot U_1 | I_1 \rangle|^2 \\
&= M^{-1} \sum_{I_1, I_{d+2} \in [M]} F_{I_1, I_{d+2}} \cdot \langle I_1 | U_1^\dagger \cdot O \cdots O \cdot U_{d+1}^\dagger | I_{d+2} \rangle \langle I_{d+2} | U_{d+1} \cdot O \cdots O \cdot U_1 | I_1 \rangle
\end{aligned}$$

as desired. \square

A.6 Fourier Coefficients of Quantum Algorithms

DQC_k Algorithms.

Proof of Claim 4.1. From Claim 2.10, the acceptance probability $f(x)$ of a d -query DQC_k algorithm is given by $f(x)$ where

$$\begin{aligned}
f(x) &= (NW)^{-1} \cdot \text{Tr} ((O_x \otimes \mathbb{I}) \cdot V_1 \cdots (O_x \otimes \mathbb{I}) \cdot V_{2d}) \\
&= (NW)^{-1} \sum_{I_1, \dots, I_{2d} \in [M]} \prod_{t \in [2d]} (V_t[I_t | I_{t+1}] \cdot x_{i_t})
\end{aligned} \tag{51}$$

with the convention that $I_{2d+1} = I_1$. We now replace x by $\rho(x)$ in Equation (51) to obtain

$$f(\rho(x)) = (NW)^{-1} \sum_{I_1, \dots, I_{2d} \in [M]} \prod_{t \in [2d]} (V_t[I_t | I_{t+1}] \cdot \rho(x)_{i_t}) \tag{52}$$

Since the coordinates in L are unfixed and the rest are fixed,

$$\rho(x)_{i_t} = \begin{cases} x_{i_t} & \text{if } i_t \in L \\ \rho_{i_t} & \text{if } i_t \notin L \end{cases}.$$

In particular,

$$\prod_{t \in [2d]} \rho(x)_{i_t} = \left(\prod_{\substack{t \in [2d] \\ \text{with } i_t \notin L}} \rho_{i_t} \right) \cdot \left(\prod_{\substack{t \in [2d] \\ \text{with } i_t \in L}} x_{i_t} \right)$$

Substituting this in [Equation \(52\)](#), we get

$$f(\rho(x)) = (NW)^{-1} \sum_{I_1, \dots, I_{2d} \in [M]} \prod_{t \in [2d]} (V_t[I_t|I_{t+1}]) \cdot \left(\prod_{\substack{t \in [2d] \\ \text{with } i_t \notin L}} \rho_{i_t} \right) \cdot \left(\prod_{\substack{t \in [2d] \\ \text{with } i_t \in L}} x_{i_t} \right). \quad (53)$$

To simplify this expression and get rid of the ρ_{i_t} , we will define a $M \times M$ diagonal matrix D^ρ and $M \times M$ unitary matrices V_t^ρ for $t \in [2d]$ as follows. For $I \in [M]$, define D^ρ to be a diagonal matrix whose I -th diagonal entry is ρ_i if $i \notin L$ and 1 otherwise. Define $V_t^\rho = D^\rho \cdot V_t$ for all $t \in [2d]$. Observe this allows us to simplify [Equation \(53\)](#) and obtain

$$f(\rho(x)) = (NW)^{-1} \sum_{I_1, \dots, I_{2d} \in [M]} \left(\prod_{t \in [2d]} V_t^\rho[I_t|I_{t+1}] \right) \cdot \left(\prod_{\substack{t \in [2d] \\ \text{with } i_t \in L}} x_{i_t} \right)$$

From here, we see that the only non-zero Fourier coefficients correspond to $S \subseteq L$ and satisfy the defining equation as in [Claim 4.1](#). The bounds on the norms of V_t^ρ follow immediately from the corresponding bounds on V_t from [Claim 2.10](#) and the fact that $\|D^\rho\|_{\text{op}} \leq 1$. \square

$\frac{1}{2}$ BQP Algorithms.

Proof of [Claim 5.1](#). Recall from [Claim 2.12](#) that the acceptance probability of a d -query $\frac{1}{2}$ BQP algorithm is given by $f(x)$ where

$$f(x) := M^{-1} \sum_{I_1, I_{d+2} \in [M]} F_{I_1, I_{d+2}} \cdot \langle I_1 | U_1^\dagger \cdot O \cdots O \cdot U_{d+1}^\dagger | I_{d+2} \rangle \langle I_{d+2} | U_{d+1} \cdot O \cdots O \cdot U_1 | I_1 \rangle. \quad (54)$$

To simplify notation, for all $t \in [d+1]$, we define $V_t := U_t^\dagger$ and $V_{2d+3-t} = U_t$. Substituting this in [Equation \(54\)](#), we get

$$f(x) := M^{-1} \sum_{I_1, I_{d+2} \in [M]} F_{I_1, I_{d+2}} \sum_{\substack{I_2, \dots, I_{d+1} \in [M] \\ I_{d+3}, \dots, I_{2d+2} \in [M]}} \left(\prod_{t \in [2d+2]} V_t[I_t|I_{t+1}] \right) \cdot \left(\prod_{t \in [2d+2] \setminus \{1, d+2\}} x_{i_t} \right). \quad (55)$$

Substituting $\rho(x)$ in place of x in [Equation \(55\)](#), we get

$$f(\rho(x)) = M^{-1} \sum_{I_1, I_{d+2} \in [M]} F_{I_1, I_{d+2}} \sum_{\substack{I_2, \dots, I_{d+1} \in [M] \\ I_{d+3}, \dots, I_{2d+2} \in [M]}} \left(\prod_{t \in [2d+2]} V_t[I_t|I_{t+1}] \right) \cdot \left(\prod_{\substack{t \in [2d+2] \setminus \{1, d+2\} \\ \text{with } i_t \notin L}} \rho_{i_t} \right) \cdot \left(\prod_{\substack{t \in [2d+2] \setminus \{1, d+2\} \\ \text{with } i_t \in L}} x_{i_t} \right). \quad (56)$$

As in the proof of [Claim 4.1](#), we will simplify this expression by defining D^ρ to be a diagonal matrix whose I -th diagonal entry is ρ_i if $i \notin L$ and 1 otherwise and let $V_1^\rho = V_1$, $V_{d+2}^\rho = V_{d+2}$ and let $V_t^\rho = D^\rho \cdot V_t$ for $t \neq 1, d+2$. This allows us to simplify [Equation \(56\)](#) as

$$f(\rho(x)) = M^{-1} \sum_{I_1, \dots, I_{2d+2} \in [M]} F_{I_1, I_{d+2}} \prod_{t \in [2d+2]} V_t^\rho [I_t | I_{t+1}] \cdot \left(\prod_{\substack{t \in [2d+2] \setminus \{1, d+2\} \\ \text{with } i_t \in L}} x_{i_t} \right).$$

From here, we see that only Fourier coefficients with $S \subseteq L$ are non-zero and are given by the defining equation in [Claim 5.1](#). The norm bounds on V_t^ρ follow immediately from the corresponding bounds in [Claim 2.12](#). This completes the proof. \square