k-Forrelation Optimally Separates Quantum and Classical Query Complexity

Nikhil Bansal
CWI Amsterdam and TU Eindhoven
The Netherlands
n.bansal@cwi.nl

Makrand Sinha
CWI Amsterdam
The Netherlands
makrand@cwi.nl

ABSTRACT

Aaronson and Ambainis (SICOMP ‘18) showed that any partial function on N bits that can be computed with an advantage δ over a random guess by making q quantum queries, can also be computed classically with an advantage $\delta/2$ by a randomized decision tree making $O_q(N^{1-\frac{1}{2q}}\delta^{-2})$ queries. Moreover, they conjectured the k-Forrelation problem — a partial function that can be computed with $q = \lceil k/2 \rceil$ quantum queries — to be a suitable candidate for exhibiting such an extremal separation.

We prove their conjecture by showing a tight lower bound of $\widetilde{\Omega}(N^{1-1/k})$ for the randomized query complexity of k-Forrelation, where $\delta = 2^{-O(k)}$. By standard amplification arguments, this gives an explicit partial function that exhibits an $O_\epsilon(1)$ vs $\Omega(N^{1-\epsilon})$ separation between bounded-error quantum and randomized query complexities, where $\epsilon > 0$ can be made arbitrarily small. Our proof also gives the same bound for the closely related but non-explicit k-Rorrelation function introduced by Tal (FOCS ‘20).

Our techniques rely on classical Gaussian tools, in particular, Gaussian interpolation and Gaussian integration by parts, and in fact, give a more general statement. We show that to prove lower bounds for k-Forrelation against a family of functions, it suffices to bound the ℓ_1-weight of the Fourier coefficients between levels k and $(k-1)k$. We also prove new interpolation and integration by parts identities that might be of independent interest in the context of rounding high-dimensional Gaussian vectors.

CCS CONCEPTS

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

KEYWORDS

Quantum query complexity, Decision Trees, Forrelation, Stochastic Calculus, Gaussian Interpolation

ACM Reference Format:
Nikhil Bansal and Makrand Sinha. 2021. k-Forrelation Optimally Separates Quantum and Classical Query Complexity. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC '21), June 21–25, 2021, Virtual, Italy.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3406325.3451040

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

ACM ISBN 978-1-4503-8053-9/21/06.
https://doi.org/10.1145/3406325.3451040

1 INTRODUCTION

The last couple of decades have given us ample evidence to suggest that quantum computers can be exponentially more powerful in solving certain computational tasks than their classical counterparts. The *black-box* or *query* model offers a concrete setting to provably show such exponential speedups. In this model, a quantum algorithm has “black-box access” to the input and seeks to compute a function of the input while minimizing the number of queries. Most well-known quantum algorithms (e.g. [8, 13, 18, 25, 26]) are captured by this black-box access model. There are slightly different models of black-box access to the input and in this work, we consider the most basic access model where each query returns a bit of the input. In this case, the classical counterpart is also commonly known as a randomized decision tree. There are many connections between the settings of quantum and randomized query complexity and we refer the reader to the survey by Buhrman and de Wolf [10].

The above raises a natural question that was first asked by Buhrman, Fortnow, Newman and Röhrig [9]: what is the maximal possible separation between quantum and classical query complexities? Translating the results from slightly different query models to the setting where the queries return a bit of the input, Simon's problem [26] and a work of Childs et al. [12] exhibited a separation of $O(\log^2 N)$ quantum queries vs $\widetilde{\Omega}(\sqrt{N})$ randomized queries for partial functions on N bits, while another work of de Beaudrap, Cleve and Watrous [7] implied a 1 vs $\Omega(N^{1/4})$ separation. However, these works left open the possibility of a $O(1)$ vs $\Omega(N)$ separation, and towards answering this question, Aaronson and Ambainis [2] showed that for $q = O(1)$, any q-query quantum algorithm can be simulated with $O(N^{1-\frac{1}{2q}})$ randomized queries, thus ruling out the possibility of a $O(1)$ vs $\Omega(N)$ separation. In particular, they proved the following fundamental result.

Theorem 1.1 ([2]). *Let Q be a quantum algorithm that makes q queries to an input $x \in \{\pm 1\}^N$. Then, with high probability, one can estimate $\mathbb{P}[Q \text{ accepts } x]$ up to an additive δ error by making $O(4^q N^{1-\frac{1}{2q}}\delta^{-2})$ classical randomized queries to x.*

In the same paper, Aaronson and Ambainis showed that the (standard) Forrelation problem, exhibits a 1 vs $\widetilde{\Omega}(\sqrt{N})$ separation, improving upon a 1 vs $\Omega(N^{1/4})$ separation shown earlier by Aaronson [1] where the standard Forrelation problem was introduced. Given the above theorem and ignoring polylog(N) factors, this is the maximal separation possible when $q = 1$.

[2] asked if Theorem 1.1 is also tight for any $q > 1$. If true, this would imply an $O(1)$ vs $\Omega(N^{1-\epsilon})$ separation where $\epsilon = O(1/q)$ could be made arbitrarily small. Towards this end, they suggested a natural generalization of the standard Forrelation problem, that

they called k-Forrelation, which we introduce next in a slightly more general setting.

(δ, k)-Forrelation. Let $\mathsf{H} = \mathsf{H}_N$ denote the $N \times N$ Hadamard matrix where $N = 2^n$ for $n \in \mathbb{N}$ and H is normalized to have orthonormal columns, and hence operator norm 1. Let $k \geq 2$ be an integer and let $\underline{i} = (i_1, \cdots, i_k) \in [N]^k$, and $z := (z_1, \cdots, z_k) \in \{\pm 1\}^{kN}$. Define the function $\mathsf{forr}_k : \{\pm 1\}^{kN} \to \mathbb{R}$ as follows

$$\mathsf{forr}_k(z) = \frac{1}{N} \sum_{\underline{i} \in [N]^k} z_1(i_1) \cdot \mathsf{H}_{i_1, i_2} \cdot z_2(i_2) \cdots \cdot z_{k-1}(i_{k-1}) \cdot \mathsf{H}_{i_{k-1}, i_k} \cdot z_k(i_k). \quad (1.1)$$

Observe that this function can be written as the following quadratic form:

$$\mathsf{forr}_k(z) = \frac{1}{N} \cdot z_1^\top (\mathsf{H} \cdot \mathsf{Z}_2 \cdot \mathsf{H} \cdot \mathsf{Z}_3 \cdots \cdot \mathsf{H} \cdot \mathsf{Z}_{k-1} \cdot \mathsf{H}) z_k, \quad (1.2)$$

where $\mathsf{Z}_i = \mathrm{diag}(z_i)$ for $i \in \{2, \ldots, k-1\}$ is the diagonal matrix with $z_i \in \{\pm 1\}^N$ on the diagonal. From the above quadratic form description, it follows that $\mathsf{forr}_k(z) \in [-1, 1]$ always, since $z_1/\sqrt{N}$ and $z_k/\sqrt{N}$ are unit vectors, and the operator norm of the matrix appearing in the quadratic form is at most 1.

For a parameter $0 < \delta < 1$, the (δ, k)-*Forrelation function* is then defined in terms of forr_k as the following partial boolean function:

$$\mathsf{forr}_{\delta,k}(z) = \begin{cases} 1 & \text{if} \quad \mathsf{forr}_k(z) \geq \delta, \text{ and} \\ 0 & \text{if} \quad |\mathsf{forr}_k(z)| \leq \delta/2. \end{cases} \quad (1.3)$$

We overload the notation forr above to denote the real function forr_k, as well as the partial boolean function $\mathsf{forr}_{\delta,k}$, but the reader should not have any ambiguity as to what is meant. Note that the standard Forrelation promise problem of [2] is obtained by taking $\delta = 3/5$ and $k = 2$.

As already observed by [2], there is a simple and efficient quantum circuit that makes $\lceil k/2 \rceil$ queries and computes (δ, k)-Forrelation in the following manner.

Proposition 1.2 ([2]). *There exists a quantum circuit Q that makes $\lceil k/2 \rceil$ queries and uses $O(k \log N)$ gates, such that for any input $z \in \{\pm 1\}^{kN}$, it holds that $\mathbb{P}[Q \text{ accepts } z] = \frac{1}{2}(1 + \mathsf{forr}_k(z))$.*

The above implies a $\delta/4$ gap between the acceptance probabilities on the 1-inputs and 0-inputs for (δ, k)-Forrelation. Standard tricks can then be used to show that with $\lceil k/2 \rceil$ quantum queries and a quantum circuit of $O(k \log N)$ size, one can compute (δ, k)-Forrelation with error at most $\frac{1}{2} - \delta/16$ on any input.

Combined with Theorem 1.1, this also shows that the (δ, k)-Forrelation function can be computed by making $O(2^k N^{1-1/k} \delta^{-2})$ classical randomized queries[1], even non-adaptively. For even values of k, this exactly matches the bound in Theorem 1.1 (upto polylog(kN) factors assuming $k = O(\log \log N)$) and Aaronson and Ambainis [2] proposed (δ, k)-Forrelation as a candidate for extremal separations between classical and quantum query complexities.

On the lower bound side, as mentioned before, Aaronson and Ambainis [2] showed that $\Omega(\sqrt{N}/\log N)$ classical queries are required for standard Forrelation. They also showed a slightly weaker lower bound of $\Omega(\sqrt{N}/\log^{7/2} N)$ for (δ, k)-Forrelation, for $\delta = 3/5$ and $k > 2$. One can improve this lower bound slightly by observing the following: in the quadratic form description (1.2) above, if we take $z_2, \cdots, z_{k-1}$ to be the all-one strings, and k is even, then (δ, k)-Forrelation reduces to standard Forrelation as $\mathsf{H}^r = \mathsf{H}$ if r is an odd natural number. So, the same $\Omega(\sqrt{N}/\log N)$ lower bound holds for $(\delta = 3/5, k)$-Forrelation as well, if k is even. Similarly, although not obvious, one can also design an input distribution achieving the same lower bound for odd k.

Thus, the current lower bounds for (δ, k)-Forrelation do not exhibit a better than $O(1)$ vs $\widetilde{\Omega}(\sqrt{N})$ separation, still leaving whether Theorem 1.1 is tight for $q > 1$ wide open.

Beyond $O(1)$ vs $\widetilde{\Omega}(\sqrt{N})$ separation. Recently, motivated by this question, Tal [27] considered a different variant of the (δ, k)-Forrelation problem, that he refers to as *k-Rorrelation*, to show a $\lceil k/2 \rceil$ vs $\widetilde{\Omega}(N^{2/3 - O(1/k)})$ separation. In particular, Tal shows that if one replaces the Hadamard matrix H in (1.1) and (1.3) by a random orthogonal matrix U, then to compute the resulting random partial function, one requires $\widetilde{\Omega}\left(N^{2(k-1)/(3k-1)}\right)$ classical queries with high probability for parameters $(\delta = 2^{-k}, k)$. Moreover, any such function can still be computed with $\lceil k/2 \rceil$ quantum queries, giving the $\lceil k/2 \rceil$ vs $\widetilde{\Omega}(N^{2/3 - O(1/k)})$ separation.

While this breaks the $\sqrt{N}$ barrier, the k-Rorrelation function is not explicit, and even though it is computable with a small number of quantum queries, the corresponding unitaries may not be efficiently implementable as a quantum circuit. This is in contrast to (δ, k)-Forrelation, where the resulting quantum query algorithms can also be efficiently implemented as a quantum circuit of polylogarithmic size. Tal's proof does not imply a better lower bound for (δ, k)-Forrelation than the $\widetilde{\Omega}(\sqrt{N})$ bound mentioned before, as it relies on various strong properties of random orthogonal matrices that the Hadamard matrix does not satisfy.

1.1 Our Results

In this work, we confirm the conjecture of Aaronson and Ambainis that (δ, k)-Forrelation does exhibit an extremal separation between classical and quantum query complexities by proving the following lower bound.

Theorem 1.3. *Let $k \geq 2$ and $\delta = 2^{-5k}$. Then, any randomized decision tree that computes (δ, k)-Forrelation with error at most $\frac{1}{2} - \frac{\eta}{2}$, must make at least the following number of queries,*

$$\Omega\left(\frac{1}{k^{29}} \cdot \left(\frac{N}{\log(kN)}\right)^{1 - \frac{1}{k}} \cdot \eta^2\right).$$

Note that for an even $k = O(1)$ and an advantage $\eta = \delta/16$, the above lower bound is $\widetilde{\Omega}(N^{1-1/k})$ and it matches the upper bound for $(\delta = \epsilon^k, k)$-Forrelation implied by Theorem 1.1, up to a polylog(kN) factor. The bound is also tight for odd k, as mentioned before.

The previous statement gives a lower bound for randomized algorithms that have a $\Theta(\delta)$ advantage, since we wish to compare it

[1] For even k this follows from Theorem 1.1 as $\lceil k/2 \rceil = k/2$. The bound also holds for odd k as the proof of Theorem 1.1 in fact shows that any bounded *block-multilinear* degree-d polynomial can be approximated up to δ additive error with $O(2^d N^{1-1/d} \delta^{-2})$ randomized queries, and forr_k is a degree-k block-multilinear polynomial for all k. The connection with query complexity arises as the acceptance probability of any q-query quantum algorithm can be written as such a polynomial of degree $2q$.

to the advantage of the $\lceil k/2 \rceil$-query quantum algorithm which has a success probability of $1/2+\Theta(\delta)$. If one wants a success probability of at least $2/3$, by using standard amplification tricks, the quantum query complexity of (δ, k)-Forrelation becomes $O(k \cdot \delta^{-2}) = 2^{O(k)}$. This gives us that there exists an explicit partial boolean function on $M = kN$ bits that can be computed with error at most $1/3$ by quantum circuits of $O(2^{O(k)} \log M)$ size, making $2^{O(k)}$ queries, but requires $M^{1-1/k}$ randomized queries.

For $k = O(1)$, this gives an $O(1)$ vs $\Omega(N^{1-\epsilon})$ bounded-error separation and taking $k = \alpha(N)$ where α is an arbitrarily slowly growing function of N, this yields an $\alpha(N)$ vs $\Omega(N^{1-o(1)})$ bounded-error separation between the quantum vs classical query complexity of an explicit partial function. More precisely, we have the following.

Corollary 1.4 (Bounded Error Separation)**.** *Let $k \geq 2$ and $\delta = 2^{-5k}$. Then, there exists a quantum circuit with $O(k \cdot 2^{10k} \cdot \log N)$ gates, making $O(k \cdot 2^{10k})$ queries that computes (δ, k)-Forrelation with error at most $1/3$. On the other hand, any randomized decision tree that computes (δ, k)-Forrelation with error at most $1/3$, needs $\Omega\left(\frac{1}{k^{29}} \cdot \left(\frac{N}{\log(kN)}\right)^{1-\frac{1}{k}}\right)$ queries.*

Remark. Our proof also works even if one replaces the Hadamard matrix H in (1.1) and (1.3) by an arbitrary orthogonal matrix U where all entries are $\widetilde{O}(N^{-1/2})$ in magnitude. In particular, the $\widetilde{\Omega}(N^{1-1/k})$ lower bound given above also holds for k-Rorrelation as all entries of a random orthogonal matrix are $O((N/\log N)^{-1/2})$ with high probability.

Next, we discuss some applications of our results.

Query Separation for Total Boolean Functions. Our results also imply an improved separation for total boolean functions. Let $Q(\mathsf{f})$ (resp. $R(\mathsf{f})$) denote the minimum number of queries made by a quantum (resp. randomized) algorithm to compute a (partial or total) boolean function f with probability at least $2/3$.

Then, the results of Aaronson, Ben-David and Kothari [3] imply that an $M^{o(1)}$ vs $M^{1-o(1)}$ separation between the quantum and randomized query complexity of a partial boolean function on M bits implies the existence of a *total* boolean function with cubic separation between the two measures. Combined with our results, this yields the following corollary.

Corollary 1.5. *There exists an explicit total boolean function f for which $R(\mathsf{f}) \geq Q(\mathsf{f})^{3-o(1)}$.*

The recent work of Aaronson, Ben-David, Kothari, Rao and Tal [4] conjectures that for any total boolean function f, it always holds that $R(\mathsf{f}) = O(Q(\mathsf{f})^3)$, so if true, the above separation is optimal up to $o(1)$ factors in the exponent. The current best upper bound is a 4th power relationship which holds even for deterministic query algorithms: denoting by $D(\mathsf{f})$ the deterministic query complexity of f, [4] prove that $D(\mathsf{f}) = O(Q(\mathsf{f})^4)$. The above is tight for deterministic query algorithms due to an example of Ambainis et al. [5].

Separations in Communication Complexity. Using the query to communication lifting theorem of Chattopadhyay, Filmus, Koroth, Meir and Pitassi [11], our results also imply analogous improved separations between quantum and classical communication complexity. In particular, let $\mathsf{ip}(x, y)$ be the inner product function where $x, y \in \{\pm 1\}^{2^{15}\log m}$. Then, for any function $f : \{\pm 1\}^m \rightarrow \{\pm 1\}$, the results of [11] imply that for the composed two-party function

$$F(x, y) = f \circ \mathsf{ip}^m (x, y) := f(\mathsf{ip}(x_1, y_1), \ldots, \mathsf{ip}(x_m, y_m)),$$

the randomized communication complexity of F with error at most $1/3$, denoted by $\mathsf{RCC}(F)$, satisfies $\mathsf{RCC}(F) = \Omega(\log m \cdot R(f))$ where $R(f)$ is the randomized query complexity of f.

Using the above with our results and denoting by $\mathsf{QCC}(F)$ the quantum communication complexity of F with error $1/3$, we have the following corollary.

Corollary 1.6. *(a) There exists an explicit partial boolean function F on M bits, such that $\mathsf{QCC}(F) = O_\epsilon(\log M)$ while $\mathsf{RCC}(F) = \Omega(M^{1-\epsilon})$, where $\epsilon > 0$ can be made arbitrarily small.*

(b) There exists an explicit total boolean function F for which $\mathsf{RCC}(F) \geq \mathsf{QCC}(F)^{3-o(1)}$.

The above results give a near optimal separation between quantum vs classical communication for partial functions improving upon the previous best known separation of $O(\log M)$ vs $\widetilde{\Omega}(\sqrt{M})$ for explicit partial functions (see [16, 20]), or an $O(\log M)$ vs $\widetilde{\Omega}(M^{2/3-\epsilon})$ separation implied by the work of [27] for non-explicit functions. We remark that whether a polynomial relation holds between the quantum and classical communication complexity of a *total* boolean function remains a very interesting open problem.

1.2 Overview and Techniques

Our proof of Theorem 1.3 is based on classical Gaussian tools, and builds on the stochastic calculus approach of Raz and Tal [23] for their breakthrough result on oracle separation between BQP and PH (see also the simplification of the results of [23] by Wu [29]).

In fact, the input distribution that [23] use is a slight variant of the distribution used for standard Forrelation ($k = 2$) by [2]. However, as also noted by [27], it is unclear how to use stochastic calculus already for $k = 3$, as the hard input distribution for randomized query algorithms has a non-linear structure involving the product of two Gaussians (we elaborate more on this later).

To get around this, our proof relies on using multilinearity of functions on the discrete cube and the properties of the underlying input distribution in a careful way, together with additional tools such as Gaussian interpolation and Gaussian integration by parts. In this overview, we first focus on the special case of $k = 3$ and restrict to the simpler setting where the advantage $\delta = 1/\mathrm{polylog}^k(N)$. This setting will already suffice to illustrate the main difficulties in extending the previous approaches to prove lower bounds for (δ, k)-Forrelation.

The case of $k = 3$. In this case, for $\underline{i} = (i_1, i_2, i_3) \in [N]^3$ and $z = (z_1, z_2, z_3) \in \mathbb{R}^{3N}$, we have

$$\mathrm{forr}_3(z) = \frac{1}{N} \sum_{\underline{i} \in [N]^3} z_1(i_1) \cdot \mathsf{H}_{i_1, i_2} \cdot z_2(i_2) \cdot \mathsf{H}_{i_2, i_3} \cdot z_3(i_3).$$

It is not hard to see that the uniform distribution on $\{\pm 1\}^{3N}$ is mostly supported on 0-inputs for $\mathrm{forr}_3(z)$. We will give a distribution $p_1(Z)$ on $\{\pm 1\}^{3N}$ — a variant of the distribution considered in [23, 27] — that is mostly supported on 1-inputs.

Given an arbitrary randomized decision tree making d queries, let $f(z)$ be the acceptance probability of the decision tree on input

z. To prove a lower bound it suffices to show that for any such f, the distinguishing advantage $\left|\mathbb{E}_{p_1}[f(Z)] - f(0)\right|$ is small, as $f(0)$ is exactly the average acceptance probability under the uniform distribution.

The distribution $p_1(Z)$. Consider the $2N \times 2N$ covariance matrix $\Sigma = \epsilon \begin{pmatrix} \mathsf{I}_N & \mathsf{H}_N \\ \mathsf{H}_N & \mathsf{I}_N \end{pmatrix}$ with $\epsilon = \Theta(1/\log N)$. A random Gaussian vector distributed as $\mathcal{N}(0,\Sigma)$ will typically lie inside the cube $[-1/2, 1/2]^{2N}$ as the variance of each coordinate is $O(1/\log N)$, and in this overview we assume that this is always the case, to avoid technicalities that can be dealt with truncating and bounding the error separately. Then, $p_1(Z)$ is the following distribution: Take two independent $2N$-dimensional Gaussian vectors $G = (U_1, V_1)$ and $B = (U_2, V_2)$ distributed as $\mathcal{N}(0,\Sigma)$ and obtain a vector $Z \in \{\pm 1\}^{3N}$ by rounding each coordinate independently to ±1 with bias given by $(U_1, U_2 \odot V_1, V_2) \in [-1/2, 1/2]^{3N}$. Here $\odot$ denotes the Hadamard product[2] of two vectors. In other words, for $i \in [N]$,

$$\begin{aligned} &\mathbb{E}_{p_1}[Z_1(i) \mid G, B] = U_1(i) \text{ and} \\ &\mathbb{E}_{p_1}[Z_2(i) \mid G, B] = U_2(i)V_1(i) \text{ and} \\ &\mathbb{E}_{p_1}[Z_3(i) \mid G, B] = V_2(i). \end{aligned} \tag{1.4}$$

Therefore, we have

$$\begin{aligned} &\mathbb{E}_{p_1}[\mathsf{forr}_3(Z)] \\ &= \frac{1}{N} \sum_{\underline{i} \in [N]^3} \mathbb{E}[U_1(i_1) \cdot \mathsf{H}_{i_1,i_2} \cdot V_1(i_2) G_2(i_2) \cdot \mathsf{H}_{i_2,i_3} \cdot V_2(i_3)] \\ &= \frac{\epsilon^2}{N} \sum_{\underline{i} \in [N]^3} \mathsf{H}^2_{i_1,i_2} \mathsf{H}^2_{i_2,i_3} = \Theta\left(\frac{1}{\log^2 N}\right), \end{aligned} \tag{1.5}$$

as $\mathbb{E}[U_1(i)V_1(j)] = \mathbb{E}[U_2(i)V_2(j)] = \epsilon \cdot \mathsf{H}_{i,j}$, and since each entry of H is $\pm\frac{1}{\sqrt{N}}$ and $\epsilon = \Theta(1/\log N)$.

Extending f from $\{\pm 1\}^{3N}$ to a function from $\mathbb{R}^{3N}$ to $\mathbb{R}$, by identifying it with its Fourier expansion, and using the multilinearity of f and the equalities in (1.4), our task then reduces to showing that

$$\begin{aligned} &\left|\mathbb{E}_{p_1}[f(Z)] - f(0)\right| \\ &\quad = \left|\mathbb{E}[f(U_1, U_2 \odot V_1, V_2)] - f(0)\right| \ll 1/\log^2 N. \end{aligned} \tag{1.6}$$

Previous approaches and their limitations. This is the starting point of all[3] previous approaches to bounding the above, which essentially proceed in the following two ways.

(a) Bounding all moments and Fourier weight of all levels. As $f(z) = \sum_{S \subseteq [3N]} \hat{f}(S)\chi_S(z)$ where $\{\chi_S(z)\}_{S \subseteq [3N]}$ are Fourier characters, one can bound

$$\begin{aligned} &\left|\mathbb{E}[f(U_1, U_2 \odot V_1, V_2)] - f(0)\right| \\ &\quad \le \sum_{\ell=1}^{d} \mathsf{wt}_\ell(f) \cdot \max_{|S|=\ell} \left|\mathbb{E}[\chi_S(U_1, U_2 \odot V_1, V_2)]\right|, \end{aligned}$$

writing $\mathsf{wt}_\ell(f) = \sum_{|S|=\ell} |\hat{f}(S)|$, as the ℓ_1-weight of the Fourier coefficients at level ℓ.

This approach needs a bound on the Fourier weight $\mathsf{wt}_\ell(f)$ for all levels $\ell \le d$, as well as a bound on all the moments $|\mathbb{E}[\chi_S(U_1, U_2 \odot V_1, V_2)]|$, and consequently suffers from two drawbacks. First, the currently known bounds on wt_ℓ for decision trees degrade as ℓ gets large — [27] shows that if f is computable by a randomized decision tree of depth d, then $\mathsf{wt}_\ell(f) \le \widetilde{O}(d)^{\ell/2}$, which becomes weaker than the trivial bound of $\binom{d}{\ell}$ when $\ell \gg \sqrt{d}$. For this reason the bound of [27] for Rorrelation does not go beyond $\widetilde{\Omega}(N^{2/3-O(1/k)})$.

Second, the moments can be very large for the Hadamard matrix (e.g. due to very large submatrices with all $1/\sqrt{N}$ entries). This is not an issue if a random orthogonal matrix is used instead (which allows [27] to go beyond $N^{1/2}$ for Rorrelation). Another limitation is that using a worst case bound for the moment given by each Fourier character does not exploit the non-trivial cancellations that can occur for various terms in the sum. In fact, it is not even clear how to obtain the $\widetilde{\Omega}(N^{1/2})$ bound for $k = 2$ using this approach.

(b) Stochastic Calculus/Gaussian Interpolation. The second approach is based on utilizing the special properties of Gaussians and using tools from stochastic calculus [23, 29]. In this paper, we describe an alternate approach using the classical method of Gaussian interpolation which can also be recovered by stochastic (Itô) calculus. Gaussian interpolation is a way to continuously interpolate between jointly Gaussian random variables with different covariance structures. By choosing a suitable path to interpolate and controlling the derivatives along this path, one can compute functions of Gaussians with a more complicated covariance structure in terms of an easier one. Talagrand [28] dubs this the *smart path method* to stress the important of choosing the right path.

In particular, let $G \in \mathbb{R}^m$ be a multivariate Gaussian and for an interpolation parameter $t \in (0,1)$, define $\mathbf{G}(t) = \sqrt{t} \cdot G$. Then, the Gaussian interpolation formula (see Section 2.1) implies that for any *reasonable* function $h : \mathbb{R}^m \to \mathbb{R}$ one has

$$\begin{aligned} \mathbb{E}[h(G)] - h(0) &= \int_0^1 \frac{d}{dt}\left(\mathbb{E}[h(\mathbf{G}(t))]\right) dt \\ &= \frac{1}{2} \sum_{ij} \mathbb{E}[G_i G_j] \int_0^1 \mathbb{E}\left[\partial_{ij} h(\mathbf{G}(t))\right] dt. \end{aligned} \tag{1.7}$$

in terms of the covariance of G and the second derivatives ∂_{ij} of h.

Note that if h is a multilinear polynomial, then $\partial_{ij}h(0) = \hat{h}(ij)$ if $i \neq j$ while $\partial_{ii}h$ is identically zero. The right-hand side above involves partial derivatives at arbitrary points $\mathbf{G}(t)$, but these can be reduced to derivatives at 0 (and hence level-two Fourier coefficients $\hat{h}(ij)$) by a clever random restriction. In particular, the derivative $\partial_{ij}h(\mu)$ at any $\mu \in [-1/2, 1/2]^m$ can be interpreted as a Fourier coefficient with respect to a biased product measure (details given later). Thus, this approach only requires a bound on the level-two weight $\mathsf{wt}_2(f)$, and works very nicely for $k = 2$, as in that case our function is a multilinear function of a Gaussian and all the corresponding covariance entries in (1.7) where $i \neq j$ are $\pm\frac{1}{\sqrt{N}}$ (as opposed to the covariance entries where $i = j$ which are large). This gives a final bound of $\frac{\mathsf{wt}_2(f)}{\sqrt{N}}$ for the expression in (1.7).

[2] For $u, v \in \mathbb{R}^m$, the Hadamard product is the vector $u \odot v \in \mathbb{R}^m$ defined as $u \odot v = (u(1) \cdot v(1), \cdots, u(m) \cdot v(m))$.

[3] We remark that the original approach of [2] does not fit in this framework and it is not clear how to generalize it either for $k > 2$.

However for $k = 3$, as also noted by [27], it is not immediately clear how to use the interpolation approach to bound the expression in (1.6), as it involves a product of Gaussians. In particular, the second block of coordinates consists of products of coordinates of Gaussians U_2 and V_1.

1.2.1 *Our Approach.*

Our main insight is that the advantage of f in (1.6) can essentially be bounded in terms of the Fourier weights of f between levels three and six. For the particular distribution $p_1(Z)$ given by (1.4), we can in fact bound the advantage of f only in terms of the third and sixth level Fourier weights (see (1.11) for the precise statement). More generally for any $k \geq 3$, the advantage of f can be bounded in terms of the Fourier weight of f between levels k and $(k-1)k$.

To show this in the simpler setting of the input distribution given by (1.6), we use Gaussian interpolation as in (1.7). In particular, for $k = 3$, given that our vector is of the form $(U_1, U_2 \odot V_1, V_2)$ and f is a multilinear polynomial, we can treat the function h in (1.7) as a function of the $4N$-dimensional Gaussian vector (U_1, U_2, V_1, V_2). Similarly, for an arbitrary k, using a suitable generalization of the distribution $p_1(Z)$, we get a function h of a $2(k-1)N$-dimensional Gaussian vector. The resulting expression in (1.7) is then a $k-1$ dimensional integral, which leads to partial derivatives of order $2k-2$ instead of ∂_{ij} in (1.7) above. However, due to the interactions between the variables of U_i and V_{i-1} (an issue which does not arise for $k = 2$), the partial deriatives with respect to U_i and V_{i-1} do not necessarily correspond to derivatives of f (with respect to its coordinates), and a key technical idea is to use Gaussian integration by parts to relate them. In particular, the order $2k-2$ derivatives of h can be related to derivatives of f of order between k and $(k-1)k$.

We remark that a recent work of Girish, Raz and Zhan [17] used a similar multi-dimensional stochastic walk to prove a lower bound for a different setting: they considered the partial function obtained by taking an XOR of multiple copies of the standard Forrelation problem, and their main focus was to prove a lower bound for quasipolynomially small advantage. The analysis for this setting is closer to the previously mentioned approaches of [23, 29] for the standard Forrelation problem. In particular, the technical challenges that arise while trying to prove a better than $\widetilde{\Omega}(\sqrt{N})$ lower bound for k-Forrelation for $k > 2$ do not arise in that case.

The case of $k = 3$ and polylogarithmic δ. We explain the idea for $k = 3$ and polylogarithmic δ first, which is quite a bit simpler, and then sketch the additional ideas needed for higher k and for improving the advantage δ to $2^{-O(k)}$. We will crucially leverage the multilinearity of the function f and the specific structure of the random vector $(U_1, U_2 \odot V_1, V_2) \in \mathbb{R}^{3N}$. In particular, let $S = S_1 \sqcup S_2 \sqcup S_3$ where S_r for $r \in [3]$ is the projection of the subset on the r^{th} block of coordinates and $\sqcup$ denotes the disjoint union of the sets. Consider the monomial $\chi_S(z)$ in the multilinear representation of f. Using the multiplicativity of the characters, we have that

$$\chi_S(U_1, U_2 \odot V_1, V_2) = \chi_{S_1}(U_1)\chi_{S_2}(U_2) \cdot \chi_{S_2}(V_1)\chi_{S_3}(V_2).$$

Our starting point is that as $G = (U_1, V_1)$ and $B = (U_2, V_2)$ are independent, one can interpolate them separately, which leads to a two-dimensional integral in (1.7), and the integrand on the right side ranges over the following derivatives

$$\begin{aligned}
&\mathbb{E}\left[\frac{\partial}{\partial u_1(i_1)\partial v_1(j_2)}\chi_{S_1}(\mathbf{U}_1(t_1))\chi_{S_2}(\mathbf{V}_1(t_1))\right] \\
&\qquad \cdot \mathbb{E}\left[\frac{\partial}{\partial u_2(i_2)\partial v_2(j_3)}\chi_{S_2}(\mathbf{U}_2(t_2))\chi_{S_3}(\mathbf{V}_2(t_2))\right] \\
&= \mathbb{E}\left[\chi_{S_1\setminus i_1}(\mathbf{U}_1(t_1))\chi_{S_2\setminus j_2}(\mathbf{V}_1(t_1))\right] \qquad (1.8)\\
&\qquad \cdot \mathbb{E}\left[\chi_{S_2\setminus i_2}(\mathbf{U}_2(t_2))\chi_{S_3\setminus j_3}(\mathbf{V}_2(t_2))\right] \\
&= \mathbb{E}\left[\chi_{S_1\setminus i_1}(\mathbf{U}_1(t_1))\chi_{S_2\setminus j_2}(\mathbf{V}_1(t_1)) \cdot \chi_{S_2\setminus i_2}(\mathbf{U}_2(t_2))\chi_{S_3\setminus j_3}(\mathbf{V}_2(t_2))\right],
\end{aligned}$$

where $(i_1, i_2) \in S_1 \times S_2$, and $(j_2, j_3) \in S_2 \times S_3$, and $t_1, t_2 \in (0, 1)$ are interpolation parameters which we will drop from the notation henceforth.

The main difference now from the $k = 2$ case is that because of the presence of products $U_2 \odot V_1$, the above derivatives can not be interpreted in general as derivatives $\frac{\partial f}{\partial z_A}(z)$ evaluated at $(\mathbf{U}_1, \mathbf{U}_2 \odot \mathbf{V}_1, \mathbf{V}_2)$.

Let us consider this more closely. Suppose that $i_2 = j_2$. In this case, (1.8) becomes

$$\mathbb{E}\left[\chi_{S_1\setminus i_1}(\mathbf{U}_1) \cdot \chi_{S_2\setminus j_2}(\mathbf{U}_2 \odot \mathbf{V}_1) \cdot \chi_{S_3\setminus j_3}(\mathbf{V}_2)\right],$$

which corresponds to a third derivative of $\chi_S(z)$ evaluated at $z = (\mathbf{U}_1, \mathbf{U}_2 \odot \mathbf{V}_1, \mathbf{V}_2)$.

However, if $i_2 \neq j_2$, then the term in (1.8) does not correspond to a derivative of $f(z)$ with respect to z. To handle this, we note that $\chi_{S_2\setminus j_2}(\mathbf{V}_1) \cdot \chi_{S_{2\setminus i_2}}(\mathbf{U}_2)$ can be written as $\chi_{S_2\setminus\{i_2,j_2\}}(\mathbf{U}_2 \odot \mathbf{V}_1) \cdot \mathbf{U}_2(j_2) \cdot \mathbf{V}_1(i_1)$, and hence (1.8) becomes

$$\mathbb{E}\left[\chi_{S_1\setminus i_1}(\mathbf{U}_1)\chi_{S_2\setminus\{i_2,j_2\}}(\mathbf{U}_2 \odot \mathbf{V}_1)\chi_{S_3\setminus j_3}(\mathbf{V}_2)\cdot\mathbf{U}_2(j_2)\cdot\mathbf{V}_1(i_1)\right], \quad (1.9)$$

In particular, the term in the expectation corresponds to the derivative of $\chi_S(\mathbf{U}_1, \mathbf{U}_2 \odot \mathbf{V}_1, \mathbf{V}_2)$ with respect to $J = \{i_1, i_2, j_2, j_3\}$ times the variables $\mathbf{U}_2(j_2)$ and $\mathbf{V}_1(i_2)$. However, this exactly fits the form required to use the Gaussian integration by parts formula (see Section 2.1), which says that for correlated real-valued Gaussians $B, G_1, \ldots, G_m$, and any reasonable function h in the variables $x_1, \ldots, x_m$, the following holds:

$$\mathbb{E}[B \cdot h(G_1, \ldots, G_m)] = \sum_{i=1}^{m} \mathbb{E}[BG_i]\,\mathbb{E}\left[\frac{\partial h}{\partial x_i}(G_1, \ldots, G_m)\right]. \qquad (1.10)$$

In particular, in (1.9), one can trade off the factors $\mathbf{U}_2(j_2)$ and $\mathbf{V}_1(i_2)$ for one additional derivative each, giving us the sixth order derivatives for χ_S. Both the cases above eventually allow us to bound the function in terms of Fourier weight of f at levels three and six.

To state our bound more formally, for $\mu \in [-1/2, 1/2]^{3N}$, consider the product measure on $\{\pm 1\}^{3N}$ where the i-th bit is 1 with probability $(1 + \mu_i)/2$ and -1 with probability $(1 - \mu_i)/2$, so that its bias is exactly μ_i. Define the level-ℓ Fourier weight with respect to bias μ as $\mathrm{wt}_\ell^\mu(f) = \sum_{|S|=\ell} |\hat{f}^\mu(S)|$, where $\hat{f}^\mu(S)$ is the Fourier coefficient with respect to the biased product measure above (see Section 2.2 for a formal definition). Then, we show the following key result towards bounding (1.6).

$$\left|\mathbb{E}[f(U_1, U_2 \odot V_1, V_2)] - f(0)\right| \lesssim \sup_{\mu \in [-1/2,1/2]^{3N}} \frac{\epsilon}{N} \cdot \mathrm{wt}_3^{\mu}(f) + \frac{\epsilon^2}{N^2} \cdot \mathrm{wt}_6^{\mu}(f). \quad (1.11)$$

By a random-restriction argument similar to that in previous works, the level-ℓ Fourier weight for a decision tree with respect to biased measures is essentially the same as the Fourier weight with respect to the uniform measure (see Corollary 3.5 later) and hence at most $\widetilde{O}(d)^{\ell/2}$ by the bounds in [27].

Plugging these bounds in (1.11) above, yields that for a depth-d randomized decision tree, the advantage is at most

$$\left|\mathbb{E}[f(U_1, U_2 \odot V_1, V_2)] - f(0)\right| \le \frac{\epsilon}{N} \cdot \widetilde{O}(d)^{3/2} + \left(\frac{\epsilon}{N} \cdot \widetilde{O}(d)^{3/2}\right)^2,$$

which is small for $d \ll N^{2/3}$. This gives the optimal bound for $(\delta, k = 3)$-Forrelation, where $\delta = \Theta(1/\log^2 N)$.

Arbitrary k and polylogarithmic δ. For $k > 3$, there is an additional complication that is not apparent in the case of $k = 3$. In this case, a suitable generalization of the distribution $p_1(Z)$ involves $k-1$ independent $2N$-dimensional Gaussian vectors (U_κ, V_κ), for $\kappa \in [k-1]$ distributed as $\mathcal{N}(0, \Sigma)$. Moreover, there are $k-2$ blocks of the form $U_\kappa \odot V_{\kappa-1}$ for $\kappa \in \{2, \ldots, k-1\}$ (see Section 3 for the exact form). Due to this, when we apply Gaussian integration by parts to trade off the (unmatched) factors $U_\kappa(i)$ and $V_\kappa(j)$ with extra derivatives, this can lead to several more additional factors.

For example, suppose we apply Gaussian integration by parts to remove the factor $U_2(i)$, then since $U_2(i)$ is correlated with various $V_2(j)$ and each $V_2(j)$ appears together with a $U_3(j)$ in $V_2 \odot U_3$, upon differentiating with respect to variables in V_2, this leads to multiple new terms with factors $U_3(j)$. Apriori, it is not obvious if applying Gaussian integration by parts leads to any progress. However, viewing this dynamics as a branching process and exploiting the multilinearity of the function f and the specific structure of the distribution $p_1(Z)$, we can show using a careful counting argument, that this process eventually terminates without giving too many higher order derivative terms.

In particular, even though the initial terms after the Gaussian interpolation step involve derivatives of order at most $2k-2$, we show that the final derivatives obtained after applying all the Gaussian integration by parts steps are of order $k, 2k, \ldots, (k-1)k$. This allows us to show an overall bound on the advantage of f, in terms of the Fourier weight of f at levels $k, 2k, 3k, \ldots, (k-1)k$ where the relative contribution of the higher level weights gets progressively smaller. In the end, plugging in the bounds on the Fourier weight, we can show that for an arbitrary k, the advantage a randomized depth-d decision tree has is at most

$$\begin{aligned}\left|\mathbb{E}_{p_1}[f(Z)] - f(0)\right| &\le \sum_{m=1}^{k-1} \left(\frac{\epsilon}{N}\right)^{m(k-1)/2} \cdot \widetilde{O}(d)^{mk/2} \\ &= \sum_{m=1}^{k-1} \left(\left(\frac{\epsilon}{N}\right)^{1-1/k} \cdot \widetilde{O}(d)\right)^{mk/2}, \quad (1.12)\end{aligned}$$

which is negligible if $d \ll N^{1-1/k}$. This gives the result for general k when $\delta = 1/\mathrm{polylog}^k(N)$. For a detailed proof along the above lines (for the setting of $\delta = 1/\mathrm{polylog}^k(N)$), we refer to the previous version [6] of our paper which might be more accessible for an unfamiliar reader since the analysis is simpler.

In the present version of the paper, we work with a different distribution, where $\delta = 2^{-O(k)}$. This requires additional ideas that make the current analysis more involved and also leads to a bound in terms of the Fourier weight of all the levels between k and $(k-1)k$ (see Theorem 3.2), as opposed to only the levels $k, 2k, 3k, \ldots, (k-1)k$ that appear in (1.12) while analyzing the previous input distribution that had a polylogarithmic advantage.

Improving δ to $2^{-O(k)}$ with new Interpolation and Integration by parts Identities. To improve δ from $1/\mathrm{polylog}^k(N)$ to $2^{-O(k)}$, we need to revisit the issues that arise from rounding. Recall that eventually we want to generate an input distribution on the discrete hypercube $\{\pm 1\}^{kN}$. One natural approach to do this is to truncate the high-dimensional Gaussians to $[-1/2, 1/2]^{kN}$ so that one can round them to $\{\pm 1\}^{kN}$ as in (1.4).

The choice of a suitable truncation function is crucial to be able to analyze the resulting quantities. In the proof overview given above, as well as in the previous version of our paper, this was achieved by scaling the Gaussians so that each coordinate has variance $\Theta(1/\log N)$. This way the Gaussians themselves lie in $[-1/2, 1/2]^{kN}$ typically, and then one can just work with the underlying Gaussian distribution directly in the analysis up to a small error that can be bounded separately. Revisiting (1.5), this results in the advantage being $1/\mathrm{polylog}^k(N)$.

To improve the advantage to $2^{-O(k)}$, we want the underlying Gaussians to have constant variance, but in this case working with the Gaussians directly causes a large rounding error, so that the previous proof strategy does not give any bounds.

This necessitates working with a different truncation function. A natural choice is the function $\frac{1}{2} \cdot \mathrm{sign} : \mathbb{R} \to \left\{-\frac{1}{2}, \frac{1}{2}\right\}$ [4]. This is difficult to analyze directly (although this can perhaps be done using the techniques presented here in conjunction with the work of Eldan and Naor [15]), and since for our application the exact constants are not so important, we work with the following truncation function: let us define $\varphi : \mathbb{R} \to \left[-\frac{1}{2}, \frac{1}{2}\right]$ as

$$\varphi(s) = \Phi(s) - \frac{1}{2} = \int_0^s \gamma(s) ds, \quad (1.13)$$

where γ and Φ are the density and cumulative distribution functions for the standard Gaussian in $\mathbb{R}$ (see Section 2).

We show that if G is a multivariate Gaussian in $\mathbb{R}^n$, then $\varphi(G) := \varphi(G_1), \ldots, \varphi(G_n)$ morally behaves like a Gaussian for our analysis and satisfies analogous interpolation and integration by parts identities. For instance, we show that the following remarkable analogue of (1.10) holds: if $B, G_1, \ldots, G_m$ are real-valued random variables that are jointly Gaussian, then for any reasonable function h, we

[4] Note that we truncate to $\left[-\frac{1}{2}, \frac{1}{2}\right]$ since the Fourier weights under the biased and unbiased measures are essentially the same if the bias is bounded away from ±1 (see Theorem 3.4). Such a statement might still be true even if the bias is arbitrarily close to ±1, but this seems more challenging to prove and is not needed for our analysis.

have

$$\mathbb{E}[\varphi(B)\cdot h(G_1,\ldots,G_m)] = \sum_{i=1}^{m}\mathbb{E}[BG_i]\,\mathbb{E}\left[\Psi(B)\cdot\frac{\partial h}{\partial x_i}(G_1,\ldots,G_m)\right], \tag{1.14}$$

where Ψ is a non-negative function that is always bounded by one. With some additional care, the identity above can be used in lieu of (1.10) to carry out the previous proof strategy even in the case of $\delta = 2^{-O(k)}$.

For more details, and for other related identities, we refer the reader to Section 4. These identities might be of independent interest in the context of rounding high-dimensional Gaussian vectors.

Independent Work of Sherstov, Storozhenko and Wu. In an independent work, Sherstov, Storozhenko and Wu [24] obtained a $\widetilde{\Omega}(N^{1-1/k})$ lower bound on the randomized query complexity of the non-explicit k-Rorrelation partial function with advantage $\delta = 2^{-O(k)}$. The proof of [24] follows the previous approach of [27] and improves the Fourier bound on level-ℓ weight of depth d-decision trees from $\sqrt{d^\ell\, O(\log N)^{\ell-1}}$ to $\sqrt{\binom{d}{\ell}\, O(\log N)^{\ell-1}}$ for all levels $\ell \leq d$. This was the only bottleneck in the approach of [27] for k-Rorrelation, and thus [24] obtain a $\widetilde{\Omega}(N^{1-1/k})$ lower bound on the randomized query complexity of k-Rorrelation.

Using the new ideas in the current version of our paper (where δ is improved to $2^{-O(k)}$ from $1/\text{polylog}^k(N)$), our work gives the same $\widetilde{\Omega}(N^{1-1/k})$ lower bound for k-Rorrelation. We also obtain the same results for the explicit k-Forrelation problem and it is unclear if this can be achieved with the other approach.

The techniques of [24] are incomparable to ours as their main focus is on proving optimal bounds on the Fourier weights of decision trees. In contrast, we improve upon a different aspect of the proof — we show a finer bound on the advantage of any depth-d decision tree where the requirements are relaxed in two ways. First, we only need low-level Fourier weights, and in this regime the previous bounds of [27] are already sufficient to give us a tight lower bound on the randomized decision tree complexity of k-Forrelation/Rorrelation, and second, the only property of the underlying orthogonal matrix we need is an absolute bound of $\widetilde{O}(N^{-1/2})$ on the entries, which holds for the Hadamard matrix as well as for a random unitary matrix, as compared to the approach of [27] and [24] which requires strong bounds on the operator norm of all large submatrices — the latter being the main reason why our approach works for the Hadamard matrix.

In addition, there have been significant recent breakthroughs in analyzing functions over the discrete cube with continuous methods, such as a stochastic characterization of Goemens-Williamson rounding [15], or the work of Eldan and Gross [14] that proved a conjecture of Talagrand in the analysis of boolean functions. The new interpolation and integration by parts identities we prove here give us additional tools that might be useful in further application of continuous techniques in theoretical computer science and mathematics.

1.2.2 *Organization.*

The rest of the paper is organized as follows. We introduce the notation and basic preliminaries in Section 2. Section 3 gives the input distribution, shows that the chosen input distribution has a large support on the 1 and 0 inputs of (δ, k)-Forrelation, and also gives a formal outline of the main proof. Section 4 introduces new interpolation and integration by parts identities that will be used repeatedly in the proof. Section 5 contains the proof of the lower bound on randomized query algorithms.

2 PRELIMINARIES

Notation. Throughout this paper, log denotes the natural logarithm unless the base is explicitly mentioned. We use $[k]$ to denote the set $\{1, 2, \ldots, k\}$. For a singleton set $\{x\}$, we sometimes write x for brevity. The set of natural numbers including zero is denoted by $\mathbb{N}_0$. Matrices are denoted by capital serif fonts (e.g. A).

For a random vector (or bit-string) z in $\mathbb{R}^n$, we will use z_i or $z(i)$ to denote the i-th coordinate of z, depending on whether we need to use the subscript for another index. If $z \in \mathbb{R}^{kn}$, then we will write $z = (z_1, \ldots, z_k)$ where $(z_\kappa)_{\kappa\in[k]}$ are vectors in $\mathbb{R}^n$ to denote the projections on the coordinates $\{(\kappa-1)n, \ldots, \kappa n\}$ — in this case, we will explicitly mention that $(z_\kappa)_{\kappa\in[k]}$ are vectors so that there is no ambiguity that z_κ refers to a coordinate of z. The operator and Frobenius norms of a matrix M are denoted by $\|\mathsf{M}\|_{\text{op}}$ and $\|\mathsf{M}\|_F$.

Random variables are denoted by capital letters (e.g. A) and values they attain are denoted by lower-case letters possibly with subscripts and superscripts (e.g. a, a_1, a', etc.). Events in a probability space will be denoted by script letters (e.g. $\mathcal{B}$). We use $\mathbf{1}_{\mathcal{B}}$ or $\mathbf{1}[\mathcal{B}]$ to denote the indicator random variable for the event $\mathcal{B}$. Given a random variable X in a probability space p, we write $p(X)$ to denote the distribution of X in the probability space. For random variables X, Y, we write $p(X, Y)$ to denote the joint distribution and $p(X)$ to denote the marginal distribution. We write $p(\mathcal{B})$ to denote the probability of the event $\mathcal{B}$. For $\lambda \in [0, 1]$, we use $\lambda p(X) + (1-\lambda)p'(X)$ to denote the convex combination of the two distributions, where the random variable X is sampled from $p(X)$ with probability λ, and from $p'(X)$ with probability $1-\lambda$.

For a real valued function f, we write $\mathbb{E}_p[f(X)]$ to denote the expectation of the random variable $f(X)$ where X is in the probability space p. Similarly, $\mathbb{E}_p[f(X) \mid Y]$ denotes the conditional expectation of $f(X)$ with respect to Y. If the probability space p is clear from the context, we simply write $\mathbb{E}[f(X)]$ and $\mathbb{E}[f(X) \mid Y]$. We use $\mathcal{N}(0, \sigma^2)$ to denote a Gaussian random variable in $\mathbb{R}$ with mean zero and variance σ^2. For a positive semi-definite matrix $\Sigma \in \mathbb{R}^{m\times m}$, we write $\mathcal{N}(0, \Sigma)$ to denote a centered (mean-zero) Gaussian random variable in $\mathbb{R}^m$ with covariance Σ. We call an m-dimensional Gaussian standard, if Σ is the identity matrix I_m.

2.1 Gaussian Tools

Gaussian Concentration. Let us denote the density and cumulative distribution function for the standard Gaussian $\mathcal{N}(0, 1)$ by

$$\gamma(s) = \frac{1}{\sqrt{2\pi}}e^{-s^2/2} \quad \text{and} \quad \Phi(s) = \int_{-\infty}^{s}\gamma(t)dt.$$

The following estimate is standard.

Proposition 2.1 (Gaussian Concentration). *For any $a > 0$, we have $1 - \Phi(a) \leq \frac{1}{2}e^{-a^2/2}$.*

Recalling the double factorial notation, $(2k+1)!! = (2k+1)\cdot(2k-1)\cdot\dots\cdot 3\cdot 1$ for any non-negative integer k, the following series expansion for the normal CDF will be very convenient.

Proposition 2.2 (Series Expansion)**.** *For every $a\in\mathbb{R}$, we have that*

$$\Phi(a) = \frac{1}{2} + \gamma(a)\sum_{k=0}^{\infty}\frac{a^{2k+1}}{(2k+1)!!}.$$

Gaussian Derivatives. From the definition of Hermite polynomials, we have that

$$\gamma^{(n)}(s) = \frac{d}{ds^n}\gamma(s) = (-1)^n\cdot h_n(s)\gamma(s), \tag{2.1}$$

where the $h_n(s)$ are the probabilists' Hermite polynomials. In addition, it is also well known (see [19]) that $|h_n(s)| \le \sqrt{n!}\cdot e^{s^2/4}$. This implies that

$$|\gamma^{(n)}(s)| \le \frac{1}{\sqrt{2\pi}}\cdot\sqrt{n!} \le n^{n/2} \text{ for every } s\in\mathbb{R}. \tag{2.2}$$

Owen's-T function. Owen's-T function [22] is defined as

$$T(h,\sigma) = \frac{1}{2\pi}\left(\arctan\sigma - \int_0^h\int_0^{\sigma x} e^{-(x^2+y^2)/2}dydx\right). \tag{2.3}$$

Note that $T(-h,\sigma) = T(h,\sigma)$ and $T(h,-\sigma) = -T(h,\sigma)$. Moreover, for $h,\sigma \ge 0$, the value $T(h,a)$ equals the probability $\mathbb{P}[X \ge h \text{ and } 0 \le Y \le \sigma X]$ where (X,Y) is standard Gaussian in $\mathbb{R}^2$.

An alternate expression (see (3.3) in [22]) for $T(h,\sigma)$, involving only a single integral is

$$T(h,\sigma) = \frac{1}{2\pi}\int_0^{\sigma}\frac{e^{-h^2(1+x^2)/2}}{1+x^2}dx. \tag{2.4}$$

To see that the expressions in (2.4) and (2.3) are equal, one can differentiate the right hand side of (2.4) with respect to h and integrate it back after a substitution.

Gaussian Interpolation and the Smart Path Method. We refer to Talagrand's book [28] for a nice exposition, and in particular, §1.3 and Appendix A.4 there, for proofs of the lemmas given below.

Let $f:\mathbb{R}^n\to\mathbb{R}$ be an infinitely differentiable function. We say that f is of moderate growth if all partial derivatives of f satisfy the following

$$\lim_{\|x\|\to\infty}\left|\partial_{\underline{i}}f(x)\right| e^{-a\|x\|^2} = 0$$
$$\text{for every } \underline{i} = (i_1,\cdots,i_n)\in\mathbb{N}_0^n \text{ and } a\in\mathbb{R}_{>0}, \tag{2.5}$$

where $\partial_{\underline{i}}$ denotes the partial derivative $\frac{\partial}{\partial x_1^{i_1}}\cdots\frac{\partial}{\partial x_n^{i_n}}$ and $\|\cdot\|$ is the Euclidean norm. One can check that multivariate polynomials are always of moderate growth, and also, the truncation function φ given in (1.13) is of moderate growth, since all its derivatives are bounded as shown by (2.2). Moreover, if $f,g:\mathbb{R}^n\to\mathbb{R}$ are of moderate growth, then so is $f(x)g(x)$. Lastly, if f is a multivariate polynomial and $q:\mathbb{R}\to\mathbb{R}$ satisfies (2.5), then $f(q(x_1),\cdots,q(x_n))$ also satisfies the moderate growth condition of (2.5).

Consider $f:\mathbb{R}^n\to\mathbb{R}$ satisfying the moderate growth condition and consider two centered jointly Gaussian random vectors G and B in $\mathbb{R}^n$. Let us define $\mathbf{G}(t) = (\mathbf{G}_i(t))_{i\le n}$ where

$$\mathbf{G}_i(t) = \sqrt{t}\,G_i + \sqrt{1-t}\,B_i, \tag{2.6}$$

so that $G = \mathbf{G}(1)$ and $B = \mathbf{G}(0)$ and consider the function

$$\zeta(t) = \mathbb{E}[f(\mathbf{G}(t))]. \tag{2.7}$$

For clarity, we will use boldface font to refer to the interpolating Gaussian.

Lemma 2.3 (Gaussian Interpolation)**.** *For $0<t<1$ we have*

$$\zeta'(t) = \frac{1}{2}\sum_{ij}\left(\mathbb{E}[G_iG_j] - \mathbb{E}[B_iB_j]\right)\,\mathbb{E}\left[\frac{\partial f}{\partial x_i\partial x_j}(\mathbf{G}(t))\right].$$

Choosing the covariance of B to be the all zero matrix, we have that $\mathbf{G}(t) = \sqrt{t}\,G$, and the following useful identity follows from the previous lemma by the fundamental theorem of calculus

$$\begin{aligned}\mathbb{E}[f(G)] - f(0) &= \int_0^1\zeta'(t)dt\\ &= \frac{1}{2}\sum_{ij}\mathbb{E}[G_iG_j]\int_0^1\mathbb{E}\left[\frac{\partial f}{\partial x_i\partial x_j}(\mathbf{G}(t))\right]dt.\end{aligned}$$

We remark that one can derive the same formula using Itô calculus.

Another important tool that we will use is the multivariate Gaussian integration by parts formula.

Lemma 2.4 (Gaussian Integration by Parts)**.** *If $B, G_1,\dots,G_n$ are real-valued random variables that are jointly Gaussian and $f:\mathbb{R}^n\to\mathbb{R}$ is of moderate growth, then*

$$\mathbb{E}[B\cdot f(G_1,\dots,G_n)] = \sum_{i=1}^n\mathbb{E}[BG_i]\,\mathbb{E}\left[\frac{\partial f}{\partial x_i}(G_1,\dots,G_n)\right].$$

Note that this formula replaces the expectation of the product of a Gaussian random variable with the function f, with a weighted sum of expectation of the derivatives of f.

The Gaussian integration by parts formula can be used to prove Lemma 2.3 and it turns out that it also uniquely characterizes the multivariate Gaussian distribution.

2.2 Fourier Analysis on the Discrete Cube

We give some facts from Fourier analysis on the discrete cube that we will need, and for more details we refer to the book [21]. Every boolean function $f:\{\pm1\}^m\to\mathbb{R}$ can be written uniquely as a sum of monomials $\chi_S(x) = \prod_{i\in S}x_i$,

$$f(x) = \sum_{S\subseteq[m]}\hat{f}(S)\chi_S(x), \tag{2.8}$$

where $\hat{f}(S) = \mathbb{E}_p[f(X)\chi_S(X)]$ is the Fourier coefficient with respect to the uniform measure p on $\{\pm1\}^m$. The monomials $\chi_S(x) = \prod_{i\in S}x_i$ form an orthonormal basis for real-valued functions on $\{\pm1\}^m$, called the *Fourier basis.*

Any function on $\{\pm1\}^m$ can be extended to $\mathbb{R}^m$ by identifying it with the multilinear polynomial given by (2.8), which is also called the *harmonic extension* of f and is unique. We will denote the harmonic extension of f also by f and in general, we have the following identity by interpolating the values of f on the vertices of the discrete hypercube.

$$f(x) = \sum_{y \in \{\pm 1\}^m} w_x(y) f(y),$$
$$\text{where} \quad w_x(y) = \prod_{i=1}^{m} \frac{1 + x_i y_i}{2} \text{ for any } x \in \mathbb{R}^m. \tag{2.9}$$

The above implies that for a boolean function $f : \{\pm 1\}^m \to [-1, 1]$, the harmonic extension of f also satisfies $\max_{x \in [1,1]^m} |f(x)| \leq 1$.

The discrete derivative of a function on the hypercube $\{\pm 1\}^m$ is given by

$$\partial_i f(x) = \frac{1}{2}(f(x^{i \to 1}) - f(x^{i \to -1})),$$

where $x^{i \to b}$ is the same as x except that the i-th coordinate is set to b. It is easily checked that the harmonic extension of $\partial_i f(x)$ is the real partial derivative $\frac{\partial}{\partial x_i}$ of the harmonic extension of f and we will identify it as such. Furthermore, for a boolean function $f : \{\pm 1\}^m \to [-1, 1]$, the discrete derivative at any point $x \in \{\pm 1\}^m$ also satisfies $|\partial_i f(x)| \leq 1$ and hence (2.9) implies that $\max_{x \in [1,1]^m} |\partial_A f(x)| \leq 1$ for any $A \subseteq [m]$ identifying $\partial_A f$ as the harmonic extension of the real partial derivative of f. Moreover, from (2.8), it also follows that

$$\partial_A f(x) = \sum_{S : S \supseteq A} \hat{f}(S) \chi_{S \setminus A}(x) \tag{2.10}$$

for any subset $A \subseteq [m]$. The above also implies that $\partial_A f(0) = \hat{f}(A)$.

The level-ℓ Fourier weight of f is defined as

$$\mathrm{wt}_\ell(f) = \sum_{|S| = \ell} |\hat{f}(S)|.$$

For a function $f(x_1, \ldots, x_m)$, a restriction $\rho \in \{-1, 1, \star\}^m$ gives a partial assignment to the variables $(x_i)_{i \leq m}$. We denote the set of coordinates of ρ whose value is $\star$ as free(ρ) while the set of coordinates that are fixed to ± 1 is denoted by fix(ρ). We use f_ρ to denote the function obtained from f by setting the variables in fix(ρ) to the values given by ρ.

Fourier basis for biased measures. For a proofs of the results below, see Chapter 8 in [21]. Given any $\mu \in (-1, 1)^m$, let $p_\mu(X)$ be the biased product distribution over $\{\pm 1\}^m$ such that each coordinate of $X \in \{\pm 1\}^m$ is sampled independently so that $X_i = 1$ with probability $(1 + \mu_i)/2$ and $X_i = -1$ with probability $(1 - \mu_i)/2$. So the expectation and the variance of X_i are

$$\mathbb{E}_{p_\mu}[X_i] = \mu_i, \quad \text{and} \quad \mathbb{E}_{p_\mu}[(X_i - \mu_i)^2] = 1 - \mu_i^2.$$

Then, the Fourier basis with respect to the biased product measure p_μ is given by the following functions indexed by subsets $S \subseteq [n]$:

$$\phi_S^\mu(x) = \prod_{i \in S} \phi_i^\mu(x), \quad \text{where} \quad \phi_i^\mu(x) = \frac{x_i - \mu_i}{\sigma_i},$$

with $\sigma_i = (1 - \mu_i^2)^{1/2}$ being the standard deviation of the biased random bit X_i. Note that

$$\begin{aligned} \mathbb{E}_{p_\mu}[\phi_S^\mu(X)^2] &= \prod_{i \in S} \mathbb{E}_{p_\mu}[\phi_i^\mu(X)^2] \\ &= \prod_{i \in S} \frac{1}{\sigma_i^2} \cdot \mathbb{E}_{p_\mu}[(X_i - \mu_i)^2] = 1, \end{aligned}$$

and that $\mathbb{E}_{p_\mu}[\phi_S^\mu(X)\phi_T^\mu(X)] = 0$ if $S \neq T$. So the functions $\phi_S^\mu(x)$ form an orthonormal basis for real-valued functions on $\{\pm 1\}^m$ with respect to the inner product obtained by taking expectation under p_μ. The Fourier expansion with respect to the biased product measure p_μ is given by

$$f(x) = \sum_{S \subseteq [n]} \hat{f}^\mu(S) \phi_S^\mu(x), \tag{2.11}$$

where $\hat{f}^\mu(S) = \mathbb{E}_{p_\mu(x)}[f(x)\phi_S^\mu(x)]$ are the Fourier coefficients with respect to p_μ.

The discrete derivative with respect to ϕ_i^μ is defined as

$$\begin{aligned} \partial_i^\mu f(x) &:= \frac{f(x^{i \to 1}) - f(x^{i \to -1})}{\phi_i^\mu(1) - \phi_i^\mu(-1)} \\ &= \sigma_i \cdot \frac{f(x^{i \to 1}) - f(x^{i \to -1})}{2} = \sigma_i \cdot \partial_i f(x), \end{aligned} \tag{2.12}$$

where $\partial_i f(x)$ is the discrete derivative with respect to the standard Fourier basis (with respect to the uniform measure over $\{\pm 1\}^m$).

Since $\partial_i f$ can be viewed as the real partial derivative of the harmonic extension of f, using the chain rule for taking derivatives, $\frac{\partial f}{\partial \phi_i^\mu} = \sigma_i \cdot \partial_i f$, so one can identify $\partial_i^\mu f$ as the real partial derivative $\frac{\partial f}{\partial \phi_i^\mu}$ for the harmonic extension of f. Moreover, from (2.11), it also follows that $\partial_S^\mu f(\mu) = \hat{f}^\mu(S)$ for any subset $S \subseteq [n]$, so μ acts as the origin with respect to the biased measure.

The level-ℓ Fourier weight of f with respect to bias μ is defined as $\mathrm{wt}_\ell^\mu(f) = \sum_{|S| = \ell} |\hat{f}^\mu(S)|$.

3 INPUT DISTRIBUTION AND THE PROOF OUTLINE

We now give a formal outline of the proof. We first give an input distribution for which (δ, k)-Forrelation is easy to compute using quantum queries, but hard for classical queries. Our distribution is a variant of that used in [27] with a different truncation function.

To define the distribution we first introduce some notation. Recall the truncation function $\varphi : \mathbb{R} \to \left[-\frac{1}{2}, \frac{1}{2}\right]$ defined as

$$\varphi(s) = \Phi(s) - \frac{1}{2} = \int_0^s \gamma(s) ds. \tag{3.1}$$

For notational convenience, we will write $\varphi(s_1, \ldots, s_m)$ to denote $(\varphi(s_1), \ldots, \varphi(s_m))$. Let us also introduce the following *block shifted Hadamard product* of two vectors: given vectors $x := (x_1, \cdots, x_{k-1}) \in \mathbb{R}^{(k-1)N}$ and $y := (y_1, \cdots, y_{k-1}) \in \mathbb{R}^{(k-1)N}$, we define $x \diamond y$ to be the following vector in $\mathbb{R}^{kN}$,

$$\begin{aligned} x \diamond y &= (x_1, \cdots, x_{k-1}, \mathbf{1}) \odot (\mathbf{1}, y_1, \cdots, y_{k-1}) \\ &= (x_1, y_1 \odot x_2, y_2 \odot x_3, \ldots, y_{k-2} \odot x_{k-1}, y_{k-1}), \end{aligned} \tag{3.2}$$

where $\mathbf{1}$ is the all ones vector in $\mathbb{R}^N$ and $\odot$ is the Hadamard product of two vectors. The above product will allow a natural generalization of the input distribution described in Section 1.2 to the case of arbitrary k. To see some examples, for $k = 2$ and vectors $x, y \in \mathbb{R}^n$, we have that $x \diamond y = (x, y)$; while for $k = 3$, we have that $x \diamond y = (x_1, y_1 \odot x_2, y_2) = (x_1, x_2 \odot y_1, y_2)$ reminiscent of the expression appearing in (1.6).

We can now describe the input distribution. Recall that $\delta = 2^{-5k}$ and let $\Sigma = \begin{pmatrix} \mathsf{I}_N & \mathsf{H}_N \\ \mathsf{H}_N & \mathsf{I}_N \end{pmatrix}$. Then, our input distribution $p(Z) = \frac{1}{2}p_0(Z) + \frac{1}{2}p_1(Z)$ where $p_0(Z)$ and $p_1(Z)$ are defined in Figure 1.

We now show that $p_b(Z)$ for $b \in \{0,1\}$ has a large support on b-inputs for (δ, k)-Forrelation.

Theorem 3.1. *For the input distribution defined in Figure 1,*

$$p_0(\mathsf{forr}_{\delta,k} \textit{ outputs } 0) \geq 1 - \frac{4}{\delta^2 N} \quad \textit{and} \quad p_1(\mathsf{forr}_{\delta,k} \textit{ outputs } 1) \geq 6\delta.$$

Proof. We first consider p_0. Since $p_0(z)$ is uniform on $\{\pm 1\}^{kN}$ and $\mathsf{forr}_k(z)$ is a multilinear and homogeneous polynomial, clearly $\mathbb{E}_{p_0(z)}[\mathsf{forr}_k(z)] = 0$. Next, we claim that $\mathbb{E}_{p_0}[\mathsf{forr}_k(Z)^2] \leq 1/N$. To see this, we use the quadratic form description (1.2). Fix any values $z_2, \ldots, z_{k-1}$, and let $\mathsf{A} = \mathsf{H} \cdot \mathrm{diag}(z_2) \cdots \cdot \mathsf{H} \cdot \mathrm{diag}(z_{k-1}) \cdot \mathsf{H}$ be the matrix appearing in the quadratic form which satisfies $\|\mathsf{A}\|_{\mathrm{op}} \leq 1$. Then, we have

$$\begin{aligned}\mathbb{E}_{p_0}[\mathsf{forr}_k(Z)^2] &= \frac{1}{N^2}\mathbb{E}_{p_0}[(Z_1^\top \mathsf{A} Z_k)^2] \\ &= \frac{1}{N^2}\sum_{ij,rs}\mathbb{E}_{p_0}[\mathsf{A}_{ij}\mathsf{A}_{rs} \cdot Z_1(i)Z_k(j)Z_1(r)Z_k(s)] \\ &= \frac{1}{N^2}\sum_{ij}\mathsf{A}_{ij}^2 = \frac{\|\mathsf{A}\|_F^2}{N^2} \leq \frac{N\|\mathsf{A}\|_{\mathrm{op}}^2}{N^2} \leq \frac{1}{N}.\end{aligned}$$

By Chebshev's inequality, it follows that $p_0(\mathsf{forr}_{\delta,k}(Z)$ outputs $1) \leq p_0(|\mathsf{forr}_k(Z)| \geq \delta/2) \leq (4/\delta^2 N)$.

We now consider p_1. As $\mathsf{forr}_k(z)$ is a multilinear polynomial, from the description of $p_1(Z)$, we have that $\mathbb{E}_{p_1}[\mathsf{forr}_k(z) \mid U, V] = \mathsf{forr}_k(\varphi(U) \diamond \varphi(V))$. Defining $X = \varphi(U)$ and $Y = \varphi(V)$, Lemma 4.3 proved in Section 4, implies that $\mathbb{E}[X_\kappa(i) \cdot \mathsf{H}_{i,j} \cdot Y_\kappa(j)] \geq \frac{1}{32} \cdot \mathsf{H}_{ij}^2$ for any $i, j \in [N]$ and $\kappa \in [k-1]$ since $\mathbb{E}[U_\kappa(i)V_\kappa(j)] = \mathsf{H}_{ij}$. Therefore,

$$\begin{aligned}&\mathbb{E}_{p_1}[\mathsf{forr}_k(Z)] = \mathbb{E}_{p_1}[\mathsf{forr}_k(X \diamond Y)] \\ &= \frac{1}{N}\sum_{\underline{i}}\mathbb{E}[X_1(i_1) \cdot \mathsf{H}_{i_1,i_2} \cdot Y_1(i_2)X_2(i_2) \cdot \mathsf{H}_{i_2,i_3} \cdots \\ &\qquad \cdot \mathsf{H}_{i_{k-2},i_{k-1}} \cdot Y_{k-2}(i_{k-1})X_{k-1}(i_k) \cdot \mathsf{H}_{i_{k-1},i_k} \cdot Y_{k-1}(i_k)] \\ &= \frac{1}{N}\sum_{\underline{i}}\mathbb{E}[X_1(i_1) \cdot \mathsf{H}_{i_1,i_2} \cdot Y_1(i_2)] \cdot \mathbb{E}[X_2(i_2) \cdot \mathsf{H}_{i_2,i_3} \cdot Y_2(i_3)] \cdots \\ &\qquad \cdot \mathbb{E}[X_{k-1}(i_k) \cdot \mathsf{H}_{i_{k-1},i_k} \cdot Y_{k-1}(i_k)] \\ &\geq \frac{1}{N}\sum_{\underline{i}}\left(\frac{1}{32}\right)^{k-1} \cdot \mathsf{H}_{i_1,i_2}^2 \cdots \mathsf{H}_{i_{k-1},i_k}^2 \\ &= \frac{1}{N}\sum_{\underline{i}}\left(\frac{1}{32}\right)^{k-1} \cdot \frac{1}{N^{k-1}} = \left(\frac{1}{32}\right)^{k-1},\end{aligned}$$

where the second equality used that (X_κ, Y_κ) are independent for different values of κ, the inequality follows from the implication of Lemma 4.3 discussed above, and the fourth equality follows since each entry of H is $\pm\frac{1}{\sqrt{N}}$ and the sum is over N^k indices. It thus follows that $\mathbb{E}_{p_1}[\mathsf{forr}_k(Z)] \geq \left(\frac{1}{32}\right)^{k-1} = 32\delta$.

Let us denote $\alpha = p_1(\mathsf{forr}_k(Z) \geq \delta)$. Recalling (1.2), we have that $|\mathsf{forr}_k(z)| \leq 1$ for $z \in [-1,1]^{kN}$. So, the above gives that $\alpha + (1-\alpha)\delta \geq 32\delta$ and hence in particular that $\alpha \geq 6\delta$, because $\delta \leq 1/2^{10}$. ■

To prove a lower bound for classical query algorithms (decision trees), we show that the advantage of any bounded real-valued function on $\{\pm 1\}^{kN}$ can be computed in terms of the low-level Fourier weight of the function f with respect to biased measures, as mentioned in Section 1.2. In particular, for $\mu \in [-1/2, 1/2]^{kN}$, consider the product measure p_μ induced on $Z \in \{\pm 1\}^{kN}$ by sampling each bit independently so that $\mathbb{E}_{p_\mu}[Z_i] = \mu_i$. Then, we prove the following which is the main contribution of this work.

Theorem 3.2. *Let $f : \{\pm 1\}^{kN} \to [0,1]$. Then,*

$$\begin{aligned}&\left|\mathbb{E}_{p_1}[f(Z)] - \mathbb{E}_{p_0}[f(Z)]\right| \\ &\quad \leq \sup_{\mu \in [-\frac{1}{2}, \frac{1}{2}]^{kN}} \sum_{\ell=k}^{k(k-1)} \left(\frac{1}{\sqrt{N}}\right)^{\ell(1-\frac{1}{k})} \cdot (8k)^{14\ell} \cdot \mathsf{wt}_\ell^\mu(f).\end{aligned}$$

Note that in the previous work of [23] for the standard Forrelation problem ($k = 2$), one only gets an upper bound in terms of the level-2 weight of the function f, but here we have an upper bound in terms of level ℓ weights where ℓ is between k and $k(k-1)$. We stress that the weight of the higher levels ($\ell > k$) can be much larger than the level-k weight, but the extra $1/\sqrt{N}$ factors in the above theorem takes care of it.

To bound the level-ℓ Fourier weight with respect to biased measures, we use the following bound proven in [27] for Fourier weights under the uniform measure.

Theorem 3.3 ([27]). *Let $f : \{\pm 1\}^m \to [0,1]$ be the acceptance probability function of a randomized depth-d decision tree. Then, for any $\ell \leq d$, the following holds for a universal constant c,*

$$\mathsf{wt}_\ell(f) \leq \left((cd)^\ell \log^{\ell-1} m\right)^{1/2},$$

where the Fourier weight $\mathsf{wt}_\ell(f)$ is with respect to the uniform measure on $\{\pm 1\}^m$.

We prove the following general statement showing that if a function and all its restrictions have a small Fourier weight on level-ℓ with respect to the uniform measure, then the Fourier weight with respect to an arbitrary bias $\mu \in \left[-\frac{1}{2}, \frac{1}{2}\right]^m$ is also small.

Theorem 3.4. *Let $f : \{\pm 1\}^m \to \mathbb{R}$ and $\ell \in [m]$. Let w be such that for any restriction $\rho \in \{-1, 1, \star\}^m$, we have $\mathsf{wt}_\ell(f_\rho) \leq w$ where the Fourier weight is with respect to the uniform measure. Then, for any $\mu \in \left[-\frac{1}{2}, \frac{1}{2}\right]^m$, we have $\mathsf{wt}_\ell^\mu(f) \leq 4^\ell w$.*

Since depth-d decision trees are closed under restrictions, combining Theorem 3.4 with Theorem 3.3 gives us that the level-ℓ weight of depth-d decision trees with an arbitrary bias $\mu \in \left[-\frac{1}{2}, \frac{1}{2}\right]^m$ is also bounded by $\left((cd)^\ell \log^{\ell-1}(m)\right)^{1/2}$.

Corollary 3.5. *Let $f : \{\pm 1\}^m \to [0,1]$ be the acceptance probability function of a randomized depth-d decision tree. Then, for any $\mu \in \left[-\frac{1}{2}, \frac{1}{2}\right]^m$ and $\ell \leq d$, we have $\mathsf{wt}_\ell^\mu(f) \leq \left((cd)^\ell \log^{\ell-1} m\right)^{1/2}$ for a universal constant c.*

Distribution $p_0(Z)$**:** Z is uniform over $\{\pm 1\}^{kN}$.

Distribution $p_1(Z)$**:** Let $(U_\kappa, V_\kappa)_{\kappa \in [k-1]}$ be independent random variables in $\mathbb{R}^{2N}$ that are distributed as $\mathcal{N}(0, \Sigma)$. Write $U = (U_\kappa)_{\kappa \in [k-1]}$ and $V = (V_\kappa)_{\kappa \in [k-1]}$ and define $W = \varphi(U) \diamond \varphi(V)$ where $W \in [-1/2, 1/2]^{kN}$. Let $Z = (Z_1, \ldots, Z_k) \in \{\pm 1\}^{kN}$ be obtained by rounding each coordinate of the vector W independently to ± 1 by interpreting them as means, i.e., for each coordinate $i \in [kN]$, we have $\mathbb{E}[Z(i) \mid U, V] = W(i)$.

Figure 1: Input Distributions $p_0(Z)$ **and** $p_1(Z)$

Combined with Theorem 3.2, the above implies that if the depth d of the decision tree satisfies $d \ll N^{1-1/k}$, then the advantage of f would be much smaller than δ.

4 INTERPOLATION AND INTEGRATION BY PARTS IDENTITIES

As mentioned before, we will use the truncation function $\varphi : \mathbb{R} \to [-\frac{1}{2}, \frac{1}{2}]$ defined in (3.1) to truncate vectors in $\mathbb{R}^m$ to $[-\frac{1}{2}, \frac{1}{2}]^m$. This necessitates generalizing the Gaussian integration by parts and Gaussian interpolation identities to handle expressions of the form $\mathbb{E}[f(\varphi(G_1), \cdots, \varphi(G_m))]$ or $\mathbb{E}[\varphi(U) \cdot f(G_1, \cdots, G_m)]$ where $U, G_1, \cdots, G_m$ are jointly Gaussian. In this section we include some such identities that will be repeatedly used throughout the paper. Their proofs can be found in the full version of the paper.

Below if $s = s_1, \ldots, s_m$, then for brevity, we write $\varphi(s)$ to denote $\varphi(s_1), \ldots, \varphi(s_m)$. The first lemma gives us an interpolation formula for functions of the form $f(\varphi(s))$.

Lemma 4.1 (Interpolation). *Let* $\Sigma = \begin{pmatrix} \mathsf{I}_n & \mathsf{M} \\ \mathsf{M}^\top & \mathsf{I}_n \end{pmatrix}$ *where* M *is an* $n \times n$ *orthogonal matrix. Let* $(U, V) \in \mathbb{R}^{2n}$ *be distributed as* $\mathcal{N}(0, \Sigma)$ *and for* $t \in (0, 1)$*, define the interpolation*

$$\zeta(t) = \mathbb{E}[f(\varphi(\mathbf{U}, \mathbf{V}))] \text{ where } (\mathbf{U}, \mathbf{V}) := (\mathbf{U}(t), \mathbf{V}(t)) = \sqrt{t} \cdot (U, V).$$

Then, for any multilinear polynomial $f(x, y)$ *where* $x = x_1, \ldots, x_n$ *and* $y = y_1, \ldots, y_n$*, the following holds*

$$\zeta'(t) = \frac{1}{1+t} \cdot \sum_{i,j=1}^{n} \mathsf{M}_{ij} \cdot \mathbb{E}\left[\frac{\partial f}{\partial x_i \partial y_j}(\varphi(\mathbf{U}, \mathbf{V}))\gamma(\mathbf{U}_i)\gamma(\mathbf{V}_j)\right].$$

The next lemma is an analogue of Gaussian integration by parts for computing expressions of the form $\mathbb{E}[\varphi(U) f(G_1, \ldots, G_m)]$.

Lemma 4.2 (Integration by Parts). *Let* $h : \mathbb{R}^m \to \mathbb{R}$ *be a moderately growing function in the variables* $x_1, \ldots, x_m$ *and let* $B, G_1, \ldots, G_m$ *be real-valued random variables that are jointly Gaussian with* $\mathbb{E}[B^2] = \sigma^2$ *with* $\sigma \in (0, 1]$*. Then, writing* $G = (G_1, \ldots, G_m)$*, we have*

$$\mathbb{E}[\varphi(B) \cdot h(G)] = \sum_{i=1}^{m} \mathbb{E}[BG_i] \cdot \mathbb{E}\left[\Psi_\sigma(B) \cdot \frac{\partial h}{\partial x_i}(G)\right],$$

where $\Psi_\sigma : \mathbb{R} \to \left[0, \frac{1}{\sqrt{2\pi}}\right]$ *is the non-negative function defined as*

$$\Psi_\sigma(s) = \frac{1}{\sqrt{2\pi}} \int_0^1 \frac{e^{-s^2 y^2/2}}{1 + \sigma^2 y^2} dy. \tag{4.1}$$

Moreover, for every integer $n \geq 1$*, the* n^{th} *derivative* $|\Psi_\sigma^{(n)}(s)| \leq n^{n/2}$ *for every* $s \in \mathbb{R}$*.*

The next lemma shows that the truncation function φ preserves correlations up to a constant factor.

Lemma 4.3. *Let* $\rho \in (-1, 1)$ *and* B, G *be real valued random variables such that* (B, G) *is distributed as* $\mathcal{N}\left(0, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$*. Then, we have* $\rho \cdot \mathbb{E}[\varphi(B)\varphi(G)] \geq \frac{\rho^2}{32}$*.*

5 LOWER BOUND FOR DECISION TREES

Here we prove Theorem 3.2 that bounds the advantage of the randomized decision tree in terms of biased Fourier weights. Theorem 3.4 regarding the Fourier weight of a function under a biased measure can be proven using a standard random restriction argument and its proof can be found in the full version.

5.1 Advantage in Terms of Fourier Weight: Proof of Theorem 3.2

By multiliearity, we have that

$$\mathbb{E}_{p_1}[f(Z)] - f(0) = \mathbb{E}_{p_1}[f(\varphi(U) \diamond \varphi(V))] - f(0). \tag{5.1}$$

To evaluate the first term on the right hand side, we will use interpolation identity given by Lemma 4.1. Recall that $(U_\kappa, V_\kappa)_{\kappa \in [k-1]}$ are independent multivariate Gaussians. We will interpolate them separately. In particular, for each $\kappa \in [k-1]$ and $t_\kappa \in (0, 1)$, we define

$$(\mathbf{U}_\kappa(t_\kappa), \mathbf{V}_\kappa(t_\kappa)) = \sqrt{t_\kappa} \cdot (U_\kappa, V_\kappa).$$

We will refer to the interpolation parameter $t = (t_1, \cdots, t_{k-1})$ as time and we will drop the time index and just write $\mathbf{U}$ and so on, if there is no ambiguity. We remind the reader of our convention that bold fonts will always refer to the interpolated random variables.

To use interpolation, we consider the function $\zeta : (0, 1)^{k-1} \to \mathbb{R}$ defined as

$$\zeta(t) = \mathbb{E}[f(\varphi(\mathbf{U}(t)) \diamond \varphi(\mathbf{V}(t)))].$$

For any fixed values of $t_1, \cdots, t_{k-2}$, by the fundamental theorem of calculus we have that

$$\begin{aligned} &\mathbb{E}[f(\varphi(\mathbf{U}(t_1, \cdots, t_{k-2}, 1)) \diamond \varphi(\mathbf{V}(t_1, \cdots, t_{k-2}, 1)))] \\ &\quad - \mathbb{E}[f(\varphi(\mathbf{U}(t_1, \cdots, t_{k-2}, 0)) \diamond \varphi(\mathbf{V}(t_1, \cdots, t_{k-2}, 0)))] \\ &\quad = \int_0^1 \frac{\partial \zeta}{\partial t_{k-1}}(t) dt_{k-1}. \end{aligned}$$

Repeating the above and fixing each index of the time parameter one by one, we obtain

$$\begin{aligned}\mathbb{E}[f(\varphi(U)\diamond\varphi(V))]-f(0)\\ &=\mathbb{E}[f(\varphi(\mathbf{U}(\mathbf{1}))\diamond\varphi(\mathbf{V}(\mathbf{1}))]-\mathbb{E}[f(\varphi(\mathbf{U}(0))\diamond\varphi(\mathbf{V}(0))]\\ &=\int_{[0,1]^{k-1}}\cdots\int\frac{\partial\zeta}{\partial t_1\cdots\partial t_{k-1}}(t)dt_{k-1}\cdots dt_1, \qquad (5.2)\end{aligned}$$

where $\mathbf{1}$ is the all ones vector in $\mathbb{R}^{k-1}$.

To bound the value of the above partial derivative (taken with respect to the time parameters) at any point, we will use Lemma 4.1. Since $f(z)$ is a multilinear polynomial, it suffices to compute the derivative of a character and towards this end, we show the following key lemma in terms of derivatives $\partial_J f=\frac{\partial f}{\partial z_J}$ where the order of the derivative $|J|$ is always between k and $k(k-1)$.

Lemma 5.1. *Let $t\in(0,1)^{k-1}$ and $S\subseteq[kN]$. Defining $\zeta_S(t)=\mathbb{E}[\chi_S(\varphi(\mathbf{U}(t))\diamond\varphi(\mathbf{V}(t)))]$, the following holds*

$$\begin{aligned}&\frac{\partial\zeta_S}{\partial t_1\cdots\partial t_{k-1}}(t)\\ &=\sum_{\ell=k}^{k(k-1)}\left(\frac{1}{\sqrt{N}}\right)^{\ell(1-\frac{1}{k})}\cdot\sum_{\substack{J\subseteq S\\|J|=\ell}}\mathbb{E}\big[\chi_{S\setminus J}(\varphi(\mathbf{U}(t))\diamond\varphi(\mathbf{V}(t)))\\ &\qquad\qquad\cdot\ \theta_J(t,\mathbf{U}(t),\mathbf{V}(t))\big],\end{aligned}$$

where $\theta_J:(0,1)^{k-1}\times\mathbb{R}^{(k-1)N}\times\mathbb{R}^{(k-1)N}\to\mathbb{R}$ is a function that only depends on J (and not on S) and satisfies $\max_{t,u,v}|\theta_J(t,u,v)|\le(4k)^{14|J|}$.

We first finish the proof of Theorem 3.2 and then prove the above lemma. Given Lemma 5.1, since $\zeta(t)=\sum_{S\subseteq[kN]}\hat{f}(S)\zeta_S(t)$, by linearity of expectation and exchanging the order of summation, it follows that for a given time t,

$$\begin{aligned}&\frac{\partial\zeta}{\partial t_1\cdots\partial t_{k-1}}(t)\\ &=\sum_{\ell=k}^{k(k-1)}\left(\frac{1}{\sqrt{N}}\right)^{\ell(1-\frac{1}{k})}\mathbb{E}\Big[\sum_{\substack{J\subseteq[kN]\\|J|=\ell}}\sum_{S:S\supseteq J}\hat{f}(S)\chi_{S\setminus J}(\varphi(\mathbf{U})\diamond\varphi(\mathbf{V}))\\ &\qquad\qquad\cdot\theta_J(t,\mathbf{U},\mathbf{V})\Big]\\ &=\sum_{\ell=k}^{k(k-1)}\left(\frac{1}{\sqrt{N}}\right)^{\ell(1-\frac{1}{k})}\mathbb{E}\Big[\sum_{\substack{J\subseteq[kN]\\|J|=\ell}}\theta_J(t,\mathbf{U},\mathbf{V})\cdot\partial_J f(\varphi(\mathbf{U})\diamond\varphi(\mathbf{V}))\Big]\\ &\qquad\qquad (5.3)\end{aligned}$$

where the second equality uses (2.10).

Next, we express the derivatives in (5.3) as biased Fourier coefficients. For any fixed values $u,v\in\mathbb{R}^{(k-1)N}$, define $\mu:=\mu(u,v)\in\left[-\frac{1}{2},\frac{1}{2}\right]^{kN}$ as $\mu=\varphi(u)\diamond\varphi(v)$ and recalling the identity (2.12), we see that $\partial_J f(z)=\hat{f}^\mu(J)/\sigma_J$ where $\sigma_J=\prod_{i\in J}\sigma_i$ with $\sigma_i=\sqrt{1-\mu_i^2}\ge1/2$. Furthermore, as $\max_{t,u,v}|\theta_J(t,u,v)|\le(4k)^{14|J|}$, equation (5.3) gives us that the following holds for any $t\in(0,1)^{k-1}$,

$$\begin{aligned}&\left|\frac{\partial\zeta}{\partial t_1\cdots\partial t_{k-1}}(t)\right|\\ &\le\sup_{u,v\in\mathbb{R}^{(k-1)N}}\sum_{\ell=k}^{k(k-1)}\left(\frac{1}{\sqrt{N}}\right)^{\ell(1-\frac{1}{k})}\sum_{\substack{J\subseteq[kN]\\|J|=\ell}}|\theta_J(t,u,v)|\cdot\frac{|\hat{f}^\mu(J)|}{\sigma_J}.\\ &\le\sup_{\mu\in[-\frac{1}{2},\frac{1}{2}]^{kN}}\sum_{\ell=k}^{k(k-1)}\left(\frac{1}{\sqrt{N}}\right)^{\ell(1-\frac{1}{k})}\cdot(8k)^{14\ell}\cdot\mathrm{wt}_\ell^\mu(f).\end{aligned}$$

Finally, using (5.2) and (5.1), the above implies that

$$\begin{aligned}&\left|\mathbb{E}_{p_1}[f(Z)]-f(0)\right|\\ &\le\sup_{\mu\in[-\frac{1}{2},\frac{1}{2}]^{kN}}\sum_{\ell=k}^{k(k-1)}\left(\frac{1}{\sqrt{N}}\right)^{\ell(1-\frac{1}{k})}\cdot(8k)^{14\ell}\cdot\mathrm{wt}_\ell^\mu(f),\end{aligned}$$

completing the proof of Theorem 3.2 given Lemma 5.1, which we prove next.

5.2 Proof of Lemma 5.1

To illustrate the key ideas, here we include the proof for the simpler case of $k=3$. The application of integration by parts is much easier here, as it does not recursively lead to other terms. For larger values of k, we need more technical care and additional ideas in the form of a careful counting argument, which can be found in the full version of the paper.

Proof for the $k=3$ Case. In this case, we shall prove that

$$\begin{aligned}&\frac{\partial\zeta_S}{\partial t_1\partial t_2}(t)\\ &=\sum_{\ell=3}^{6}\sum_{\substack{J\subseteq S\\|J|=\ell}}\left(\frac{1}{\sqrt{N}}\right)^{2\ell/3}\cdot\mathbb{E}[\chi_{S\setminus J}(\varphi(\mathbf{U})\diamond\varphi(\mathbf{V}))\cdot\theta_J(t,\mathbf{U},\mathbf{V})], \quad (5.4)\end{aligned}$$

where $\theta_J(t,u,v)$ is a function that only depends on J and not on S and satisfies $\max_{t,u,v}|\theta_J(t,u,v)|\le12^{14|J|}$.

Let $S=S_1\sqcup S_2\sqcup S_3$ where $S_1\subseteq[N],S_2\subseteq\{N+1,\ldots,2N\}$ and $S_3\subseteq\{2N+1,\ldots,3N\}$. Let us also define $X_\kappa=\varphi(U_\kappa)$ and $Y_\kappa=\varphi(V_\kappa)$ for $\kappa\in[2]$ and analogously we define $\mathbf{X}_\kappa$ and $\mathbf{Y}_\kappa$ in terms of the interpolated Gaussians $\mathbf{U}_\kappa$ and $\mathbf{V}_\kappa$. We first observe that because of the multiplicativity of the characters χ_S and the definition of block-shifted Hadamard product, we have that for any $x,y\in\mathbb{R}^{2N}$,

$$\chi_S(x\diamond y)=\chi_{S_1}(x_1)\chi_{S_2}(y_1)\cdot\chi_{S_2}(y_2)\chi_{S_3}(y_2). \qquad (5.5)$$

We will treat the functions $\chi_{S_1}(x_1)\chi_{S_2}(y_1)$ and $\chi_{S_2}(x_2)\chi_{S_3}(y_2)$ as function in the variables $x_1=(x_1(i))_{i\in S_1},y_1=(y_1(j))_{j\in S_2}$ and $x_2=(x_2(i))_{i\in S_2},y_2=(y_2(j))_{j\in S_3}$ respectively and use $\frac{\partial}{\partial x_1(i)},\frac{\partial}{\partial x_1(j)}$ to denote the corresponding partial derivatives. To prevent any confusion, we clarify that $\partial_i=\frac{\partial}{\partial z_i}$ will always denote the derivative with respect to z.

Now, since (U_1,V_1) and (U_2,V_2) are independent Gaussians and they are being interpolated separately, we can apply the interpolation formula given by Lemma 4.1 separately to compute the expectations $\mathbb{E}[\chi_{S_1}(X_1)\chi_{S_2}(Y_1)]$ and $\mathbb{E}[\chi_{S_2}(X_2)\chi_{S_3}(Y_2)]$ appearing in (5.5).

Therefore, applying Lemma 4.1 and using linearity of expectation, we have that $\frac{\partial \zeta_S}{\partial t_1 \partial t_2}(t)$ equals

$$\begin{aligned}
&\sum_{\underline{i},\underline{j}} \frac{\mathsf{H}_{i_1,j_2}}{1+t_1} \cdot \frac{\mathsf{H}_{i_2,j_3}}{1+t_2} \\
&\qquad \cdot \mathbb{E}\left[\frac{\partial}{\partial x_1(i_1)\partial y_1(j_2)} \chi_{S_1}(\mathbf{X}_1)\chi_{S_2}(\mathbf{Y}_1)\gamma(\mathbf{U}_1(i_1))\gamma(\mathbf{V}(j_2))\right] \\
&\qquad \cdot \mathbb{E}\left[\frac{\partial}{\partial x_2(i_2)\partial y_2(j_3)} \chi_{S_2}(\mathbf{X}_2)\chi_{S_3}(\mathbf{Y}_2)\gamma(\mathbf{U}_2(i_2))\gamma(\mathbf{V}_2(j_3))\right] \\
&= \sum_{\underline{i},\underline{j}} \frac{\mathsf{H}_{i_1,j_2}}{1+t_1} \cdot \frac{\mathsf{H}_{i_2,j_3}}{1+t_2} \cdot \mathbb{E}[\chi_{S_1\setminus i_1}(\mathbf{X_1})\chi_{S_2\setminus j_2}(\mathbf{Y_1})\gamma(\mathbf{U}_1(i_1))\gamma(\mathbf{V}(j_2))] \\
&\qquad \cdot \mathbb{E}[\chi_{S_2\setminus i_2}(\mathbf{X_2})\chi_{S_3\setminus j_3}(\mathbf{Y_2})\gamma(\mathbf{U}_2(i_2))\gamma(\mathbf{V}_2(j_3))],
\end{aligned} \tag{5.6}$$

writing $\underline{i} = (i_1, i_2) \in S_1 \times S_2$ and $\underline{j} = (j_2, j_3) \in S_2 \times S_3$. Note that the indices are shifted for $\underline{j}$ to clarify that they lie in the corresponding set S_r and we will keep using this indexing convention.

We can classify the terms in (5.6) into two types: terms where $i_2 = j_2$ and where $i_2 \neq j_2$. These behave very differently, and we bound their contributions separately.

(a) Terms where $i_2 = j_2$: In this case, defining $i_3 = j_3$, extending the tuple $\underline{i} = (i_1, i_2, i_3)$, the corresponding terms in (5.6) are given by

$$\begin{aligned}
&\frac{\mathsf{H}_{i_1,j_2}}{1+t_1} \cdot \frac{\mathsf{H}_{i_2,j_3}}{1+t_2} \cdot \mathbb{E}[\chi_{S_1\setminus i_1}(\mathbf{X_1})\chi_{S_2\setminus i_2}(\mathbf{Y_1} \odot \mathbf{X_2})\chi_{S_3\setminus i_3}(\mathbf{Y_2}) \\
&\qquad\qquad \cdot\ \gamma(\mathbf{U_1}(i_1))\gamma(\mathbf{V_1}(i_2))\gamma(\mathbf{U_2}(i_2))\gamma(\mathbf{V_2}(i_3))] \\
&= \mathsf{H}_{i_1,i_2} \cdot \mathsf{H}_{i_2,i_3} \cdot \mathbb{E}\Big[\chi_{S\setminus\{i_1,i_2,i_3\}}(\mathbf{X} \diamond \mathbf{Y}) \\
&\qquad\qquad \cdot\ \frac{\gamma(\mathbf{U_1}(i_1))\gamma(\mathbf{V_1}(i_2))\gamma(\mathbf{U_2}(i_2))\gamma(\mathbf{V_2}(i_3))}{(1+t_1)(1+t_2)}\Big] \\
&= \left(\frac{1}{\sqrt{N}}\right)^2 \cdot \mathbb{E}[\chi_{S\setminus\{i_1,i_2,i_3\}}(\varphi(\mathbf{U}) \diamond \varphi(\mathbf{V}))\ \cdot\ \theta_{\underline{i}}(t, \mathbf{U}, \mathbf{V})],
\end{aligned}$$

where we used that $\mathbf{X} = \varphi(\mathbf{U})$ and $\mathbf{Y} = \varphi(\mathbf{V})$ and the function $\theta_{\underline{i}}(t, u, v)$ is defined as

$$\mathsf{sign}(\mathsf{H}_{i_1,i_2} \cdot \mathsf{H}_{i_2,i_3}) \cdot \frac{\gamma(u_1(i_1))\gamma(v_1(i_2))\gamma(u_2(i_2))\gamma(v_2(i_3))}{(1+t_1)(1+t_2)}.$$

Viewing the tuple $\underline{i}$ as a set $J \subseteq S$ of size 3, this gives us that the sum of all the terms in (5.6) where $i_2 = j_2$ is exactly

$$\sum_{\substack{J\subseteq S\\ |J|=3}} \left(\frac{1}{\sqrt{N}}\right)^2 \cdot \mathbb{E}[\chi_{S\setminus J}(\varphi(\mathbf{U}) \diamond \varphi(\mathbf{V})) \cdot \theta_J(t, \mathbf{U}, \mathbf{V})], \tag{5.7}$$

where $\max_{t,u,v} |\theta_J(t, u, v)| \leq 1$.

(b) Terms where $i_2 \neq j_2$: To bound these terms, we use the integration by parts identity of Lemma 4.2 to reduce them to derivatives of orders 4, 5 and 6. Consider a fixed term where $i_2 \neq j_2$. Then, the corresponding expectation term in (5.6) is

$$\begin{aligned}
&\mathbb{E}[\chi_{S_1\setminus i_1}(\mathbf{X_1})\chi_{S_2\setminus j_2}(\mathbf{Y_1}) \cdot \gamma(\mathbf{U}_1(i_1))\gamma(\mathbf{V}_1(j_2))] \\
&\quad \cdot \mathbb{E}[\chi_{S_2\setminus i_2}(\mathbf{X_2})\chi_{S_3\setminus j_3}(\mathbf{Y_2}) \cdot \gamma(\mathbf{U}_2(i_2))\gamma(\mathbf{V}_2(j_3))].
\end{aligned} \tag{5.8}$$

As $i_2 \neq j_2$, the term $\chi_{S_2\setminus j_2}(\mathbf{Y_1})$ still depends on the random variable $\mathbf{Y}_1(i_2)$ (while $\chi_{S_2\setminus j_2}(\mathbf{X_2})$ does not). Since eventually we need a function of $\mathbf{X}_2 \odot \mathbf{Y}_1$, we pull out $\mathbf{Y}_1(i_2)$ and write,

$$\begin{aligned}
&\mathbb{E}[\chi_{S_{1\setminus i_1}}(\mathbf{X_1})\chi_{S_2\setminus j_2}(\mathbf{Y_1}) \cdot \gamma(\mathbf{U}_1(i_1))\gamma(\mathbf{V}_1(j_2))] \\
&= \mathbb{E}\left[\mathbf{Y}_1(i_2) \cdot \chi_{S_1\setminus i_1}(\mathbf{X_1})\chi_{S_2\setminus\{i_2,j_2\}}(\mathbf{Y_1}) \cdot \gamma(\mathbf{U}_1(i_1))\gamma(\mathbf{V}_1(j_2))\right]
\end{aligned} \tag{5.9}$$

Recalling that $\mathbf{X_1} = \varphi(\mathbf{U_1})$ and $\mathbf{Y_1} = \varphi(\mathbf{V_1})$, we can now apply Lemma 4.2 with

$$h := h(u, v) = \chi_{S_1\setminus i_1}(\varphi(u_1))\chi_{S_2\setminus\{i_2,j_2\}}(\varphi(v_1)) \cdot \gamma(u_1(i_1))\gamma(v_1(j_2)),$$

and $B = \mathbf{Y_1}(i_2) = \varphi(\mathbf{V_1}(i_2))$. Note the very crucial fact that h does not depend on the variable $v_1(i_2)$ as the term $\chi_{S_2\setminus\{i_2,j_2\}}(\varphi(v_1))$ does not contain it and neither does $\gamma(v_1(j_2))$ as $j_2 \neq i_2$. Thus, to apply Lemma 4.2, we only need to care about the terms corresponding to $\mathbb{E}[\mathbf{V_1}(i_2)\mathbf{U_1}(q_1)] = t_1\mathsf{H}_{q_1,i_2}$ as the other terms will disappear — those corresponding to $\mathbb{E}[\mathbf{V_1}(i_2)\mathbf{V_1}(q_2)]$, where $i_2 \neq q_2$, disappear because $\mathbb{E}[\mathbf{V_1}(i_2)\mathbf{V_1}(q_2)] = 0$, and the term corresponding to $\mathbb{E}[\mathbf{V_1}(i_2)^2]$ disappears as $\frac{\partial h}{\partial v_1(i_2)} = 0$.

Since, $\mathbb{E}[\mathbf{V}(i_2)\mathbf{U}_1(q_1)] = t_1\mathsf{H}_{q_1,i_2}$, and

$$\begin{aligned}
\frac{\partial h}{\partial u_1(q_1)} &= \chi_{S_1\setminus\{i_1,q_1\}}(\varphi(u_1))\chi_{S_2\setminus\{i_2,j_2\}}(\varphi(v_1)) \\
&\qquad \cdot \gamma(u_1(i_1))\gamma(v_1(j_2)),\ \text{if } q_1 \neq i_1,\ \text{and,} \\
\frac{\partial h}{\partial u_1(i_1)} &= \chi_{S_1\setminus i_1}(\varphi(u_1))\chi_{S_2\setminus\{i_2,j_2\}}(\varphi(v_1)) \\
&\qquad \cdot \gamma'(u_1(i_1))\gamma(v_1(j_2)),\ \text{otherwise,}
\end{aligned}$$

Lemma 4.2 gives us that (5.9) equals

$$\begin{aligned}
&\sum_{\substack{q_1\in S_1,\\ q_1\neq i_1}} t_1\mathsf{H}_{q_1,i_2} \cdot \mathbb{E}\Big[\chi_{S_1\setminus\{i_1,q_1\}}(\mathbf{X_1})\chi_{S_2\setminus\{i_2,j_2\}}(\mathbf{Y_1}) \\
&\qquad\qquad \cdot\ \gamma(\mathbf{U}_1(i_1))\gamma(\mathbf{U}_1(q_1))\Psi_{t_1}(\mathbf{V}_1(i_2))\gamma(\mathbf{V}_1(j_2))\Big] \\
&+ t_1\mathsf{H}_{i_1,i_2} \cdot \mathbb{E}\Big[\chi_{S_1\setminus i_1}(\mathbf{X_1})\chi_{S_2\setminus\{i_2,j_2\}}(\mathbf{Y_1}) \\
&\qquad\qquad \cdot\ \gamma'(\mathbf{U}_1(i_1))\Psi_{t_1}(\mathbf{V}_1(i_2))\gamma(\mathbf{V}_1(j_2))\Big]
\end{aligned} \tag{5.10}$$

Analogously, for the second expectation in (5.8), the term $\chi_{S_2\setminus i_2}(\mathbf{X_1})$ still depends on the random variable $\mathbf{X}_2(j_2) = \varphi(\mathbf{U}_2(j_2))$. Applying integration by parts as above, one gets that the second expectation in (5.8) equals

$$\begin{aligned}
&\sum_{\substack{q_3\in S_3,\\ q_3\neq j_3}} t_2\mathsf{H}_{j_2,q_3} \cdot \mathbb{E}\Big[\chi_{S_2\setminus\{j_2,i_2\}}(\mathbf{X_2})\chi_{S_2\setminus\{j_3,q_3\}}(\mathbf{Y_2}) \\
&\qquad\qquad \cdot\ \gamma(\mathbf{U}_2(i_2))\Psi_{t_2}(\mathbf{U}_2(j_2))\gamma(\mathbf{Y}_2(j_3))\gamma(\mathbf{V}_2(q_3))\Big] \\
&+ t_2\mathsf{H}_{j_2,q_3} \cdot \mathbb{E}\Big[\chi_{S_2\setminus\{j_2,i_2\}}(\mathbf{X_2})\chi_{S_2\setminus j_3}(\mathbf{Y_2}) \\
&\qquad\qquad \cdot\ \gamma(\mathbf{U}_2(i_2))\Psi_{t_2}(\mathbf{U}_2(j_2))\gamma'(\mathbf{V}_2(j_3))\Big]
\end{aligned} \tag{5.11}$$

Combining (5.10) and (5.11), we get the sum of all the terms in (5.6) where $i_2 \neq j_2$. In particular, defining new tuples $\underline{\alpha} = (i_1, j_2, q_3) \in S_1 \times S_2 \times S_3$ and $\underline{\beta} = (q_1, i_2, j_3) \in S_1 \times S_2 \times S_3$, where we allow $q_1 = i_1$ and $q_3 = i_3$, we get that the sum of the terms in (5.6) where

$i_2 \neq j_2$ equals

$$\begin{aligned}
&\sum_{\underline{\alpha},\underline{\beta}} \left(\frac{1}{\sqrt{N}}\right)^4 s_{\underline{\alpha},\underline{\beta}} \cdot \mathbb{E}[\chi_{S_1\setminus\{i_1,q_1\}}(\mathbf{X}_1)\cdot\chi_{S_2\setminus\{j_2,i_2\}}(\mathbf{Y}_1) \\
&\qquad\qquad \cdot \chi_{S_2\setminus\{j_2,i_2\}}(\mathbf{X}_2)\cdot\chi_{S_3\setminus\{j_3,q_3\}}(\mathbf{Y}_2) \;\cdot\; v_{\underline{\alpha},\underline{\beta}}(t,\mathbf{U},\mathbf{V})] \\
&= \sum_{\underline{\alpha},\underline{\beta}} \left(\frac{1}{\sqrt{N}}\right)^4 s_{\underline{\alpha},\underline{\beta}} \cdot \mathbb{E}[\chi_{S_1\setminus\{i_1,q_1\}}(\mathbf{X}_1)\cdot\chi_{S_2\setminus\{j_2,i_2\}}(\mathbf{Y}_1\odot\mathbf{X}_2) \\
&\qquad\qquad \cdot \chi_{S_3\setminus\{j_3,q_3\}}(\mathbf{Y}_2) \;\cdot\; v_{\underline{\alpha},\underline{\beta}}(t,\mathbf{U},\mathbf{V})] \\
&= \sum_{\underline{\alpha},\underline{\beta}} \left(\frac{1}{\sqrt{N}}\right)^4 \mathbb{E}[\chi_{S\setminus\{i_1,q_1,i_2,j_2,j_3,q_3\}}(\mathbf{X}\diamond\mathbf{Y}) \;\cdot\; \theta_{\underline{\alpha},\underline{\beta}}(t,\mathbf{U},\mathbf{V})],
\end{aligned} \tag{5.12}$$

where the sum ranges over all possible tuples $\underline{\alpha},\underline{\beta}$ satisfying $i_2 \neq j_2$ and $s_{\underline{\alpha},\underline{\beta}} := \mathsf{sign}(\mathsf{H}_{i_1,j_2}\mathsf{H}_{j_2,q_3}\mathsf{H}_{q_1,i_2}\mathsf{H}_{i_2,j_3})$, the function $v_{\underline{\alpha},\underline{\beta}}(t,u,v)$ is some function that is always bounded by one (since γ,γ' and Ψ_σ are all bounded by one in magnitude and $t \in (0,1)^2$), and the function $\theta_{\underline{\alpha},\underline{\beta}}(t,u,v) := s_{\underline{\alpha},\underline{\beta}} \cdot v_{\underline{\alpha},\underline{\beta}}(t,u,v)$.

Note that there are at most 8 possible tuples $\underline{\alpha},\underline{\beta}$ that give rise to the set $J = \{i_1,q_1,i_2,j_2,j_3,q_3\}$. It follows that the sum in (5.12) is exactly

$$\sum_{\ell=4}^{6} \sum_{\substack{J\subseteq S\\ |J|=\ell}} \left(\frac{1}{\sqrt{N}}\right)^4 \cdot \mathbb{E}[\chi_{S\setminus J}(\varphi(\mathbf{U})\diamond\varphi(\mathbf{V})) \;\cdot\; \theta_J(t,\mathbf{U},\mathbf{V})], \tag{5.13}$$

where $\theta_J(t,u,v)$ only depends on J and $\max_{t,u,v}|\theta_J(t,u,v)| \leq 8$. The level four and five weights appear since we allow the possibility that $i_1 = q_1$ or $j_3 = q_3$.

Then, plugging in the bounds from (5.7) and (5.12) for the two cases in (5.6), we get (5.4).

ACKNOWLEDGMENTS

We thank Ronald de Wolf for discussions and for providing helpful feedback on the writing, Avishay Tal for useful comments and for encouraging us to extend our results to obtain the improved parameter $\delta = 2^{-O(k)}$, and Arkadev Chattopadhyay and Suhail Sherif for pointing out the applications to communication complexity. This work was supported by the NWO VICI grant 639.023.812.

REFERENCES

[1] Scott Aaronson. 2010. BQP and the Polynomial Hierarchy *(STOC '10)*. New York, NY, USA, 141–150. https://doi.org/10.1145/1806689.1806711

[2] Scott Aaronson and Andris Ambainis. 2018. Forrelation: A Problem That Optimally Separates Quantum from Classical Computing. *SIAM J. Comput.* 47, 3 (2018), 982–1038. https://doi.org/10.1137/15M1050902

[3] Scott Aaronson, Shalev Ben-David, and Robin Kothari. 2016. Separations in Query Complexity Using Cheat Sheets. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*. 863–876. https://doi.org/10.1145/2897518.2897644

[4] Scott Aaronson, Shalev Ben-David, Robin Kothari, Shravas Rao, and Avishay Tal. 2020. Degree vs. Approximate Degree and Quantum Implications of Huang's Sensitivity Theorem. (October 2020). arXiv:2010.12629

[5] Andris Ambainis, Kaspars Balodis, Aleksandrs Belovs, Troy Lee, Miklos Santha, and Juris Smotrovs. 2017. Separations in Query Complexity Based on Pointer Functions. *J. ACM* 64, 5, Article 32 (Sept. 2017). https://doi.org/10.1145/3106234

[6] Nikhil Bansal and Makrand Sinha. 2020. k-Forrelation Optimally Separates Quantum and Classical Query Complexity. (August 2020). arXiv:2008.07003v1

[7] J. Niel de Beaudrap, Richard Cleve, and John Watrous. 2002. Sharp Quantum versus Classical Query Complexity Separations. *Algorithmica* 34, 4 (2002), 449–461. https://doi.org/10.1007/s00453-002-0978-1

[8] Ethan Bernstein and Umesh Vazirani. 1997. Quantum Complexity Theory. *SIAM J. Comput.* 26, 5 (1997), 1411–1473. https://doi.org/10.1137/S0097539796300921

[9] Harry Buhrman, Lance Fortnow, Ilan Newman, and Hein Röhrig. 2008. Quantum Property Testing. *SIAM J. Comput.* 37, 5 (2008), 1387–1400. https://doi.org/10.1137/S0097539704442416

[10] Harry Buhrman and Ronald de Wolf. 2002. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science* 288, 1 (2002), 21 – 43. https://doi.org/10.1016/S0304-3975(01)00144-X Complexity and Logic.

[11] Arkadev Chattopadhyay, Yuval Filmus, Sajin Koroth, Or Meir, and Toniann Pitassi. 2019. Query-To-Communication Lifting for BPP Using Inner Product. In *ICALP (LIPIcs, Vol. 132)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 35:1–35:15. https://doi.org/10.4230/LIPIcs.ICALP.2019.35

[12] Andrew M. Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A. Spielman. 2003. Exponential Algorithmic Speedup by a Quantum Walk *(STOC '03)*. 59–68. https://doi.org/10.1145/780542.780552

[13] David Deutsch and Richard Jozsa. 1992. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* 439, 1907 (1992), 553–558. https://doi.org/10.1098/rspa.1992.0167

[14] Ronen Eldan and Renan Gross. 2020. Concentration on the Boolean hypercube via pathwise stochastic analysis. In *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020*. ACM, 208–221. https://doi.org/10.1145/3357713.3384230

[15] Ronen Eldan and Assaf Naor. 2019. Krivine diffusions attain the Goemans-Williamson approximation ratio. (June 2019). arXiv:1906.10615

[16] Dmitry Gavinsky. 2020. Entangled Simultaneity Versus Classical Interactivity in Communication Complexity. *IEEE Trans. Inf. Theory* 66, 7 (2020), 4641–4651. https://doi.org/10.1145/2897518.2897545

[17] Uma Girish, Ran Raz, and Wei Zhan. 2020. Lower Bounds for XOR of Forrelations. (July 2020). arXiv:2007.03631

[18] Lov K. Grover. 1996. A Fast Quantum Mechanical Algorithm for Database Search *(STOC '96)*. 212–219. https://doi.org/10.1145/237814.237866

[19] Jack Indritz. 1961. An Inequality for Hermite Polynomials. *Proc. Amer. Math. Soc.* 12, 6 (1961), 981–983. https://doi.org/10.1090/S0002-9939-1961-0132852-2

[20] Bo'az Klartag and Oded Regev. 2011. Quantum one-way communication can be exponentially stronger than classical communication. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC*. ACM, 31–40. https://doi.org/10.1145/1993636.1993642

[21] Ryan O'Donnell. 2014. *Analysis of Boolean Functions*. Cambridge University Press. https://doi.org/10.1017/CBO9781139814782

[22] Donald B. Owen. 1956. Tables for Computing Bivariate Normal Probabilities. *Ann. Math. Statist.* 27, 4 (12 1956), 1075–1090. https://doi.org/10.1214/aoms/1177728074

[23] Ran Raz and Avishay Tal. 2019. Oracle separation of BQP and PH. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019*. ACM, 13–23. https://doi.org/10.1145/3313276.3316315

[24] Alexander A. Sherstov, Andrey A. Storozhenko, and Pei Wu. 2020. An Optimal Separation of Randomized and Quantum Query Complexity. (November 2020). arXiv:2008.10223

[25] Peter W. Shor. 1997. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* 26, 5 (Oct. 1997), 1484–1509. https://doi.org/10.1137/S0097539795293172

[26] Daniel R. Simon. 1997. On the Power of Quantum Computation. *SIAM J. Comput.* 26, 5 (Oct. 1997), 1474–1483. https://doi.org/10.1137/S0097539796298637

[27] A. Tal. 2020. Towards Optimal Separations between Quantum and Randomized Query Complexities. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*. 228–239. https://doi.org/10.1109/FOCS46700.2020.00030

[28] Michel Talagrand. 2011. *Mean Field Models for Spin Glasses, Volume I: Basic Examples*. Springer-Verlag Berlin Heidelberg. https://doi.org/10.1007/978-3-642-15202-3

[29] Xinyu Wu. 2020. A stochastic calculus approach to the oracle separation of BQP and PH. (July 2020). arXiv:2007.02431