A Composition Theorem for Randomized Query complexity

Anurag Anshu [*] Dmitry Gavinsky [†] Rahul Jain [‡] Srijita Kundu [§]
Troy Lee [¶] Priyanka Mukhopadhyay [‖] Miklos Santha [**] Swagato Sanyal [††]

June 2, 2017

Abstract

Let the randomized query complexity of a relation for error probability ϵ be denoted by $\mathsf{R}_\epsilon(\cdot)$. We prove that for any relation $f \subseteq \{0,1\}^n \times \mathcal{R}$ and Boolean function $g : \{0,1\}^m \to \{0,1\}$, $\mathsf{R}_{1/3}(f \circ g^n) = \Omega(\mathsf{R}_{4/9}(f) \cdot \mathsf{R}_{1/2-1/n^4}(g))$, where $f \circ g^n$ is the relation obtained by composing f and g. We also show that $\mathsf{R}_{1/3}\left(f \circ \left(g^{\oplus}_{O(\log n)}\right)^n\right) = \Omega(\log n \cdot \mathsf{R}_{4/9}(f) \cdot \mathsf{R}_{1/3}(g))$, where $g^{\oplus}_{O(\log n)}$ is the function obtained by composing the xor function on $O(\log n)$ bits and g^t.

1 Introduction

Given two Boolean functions $f : \{0,1\}^n \to \{0,1\}$ and $g : \{0,1\}^m \to \{0,1\}$, the composed function $f \circ g^n : (\{0,1\}^m)^n \to \{0,1\}$ is defined as follows: For $x = (x^{(1)}, \ldots, x^{(n)}) \in (\{0,1\}^m)^n$, $f \circ g^n(x) = f(g(x^{(1)}), \ldots, g(x^{(n)}))$. Composition of Boolean functions has long been a topic of active research in complexity theory. In many works, composition of Boolean function is studied in the context of a certain complexity measure. The objective is to understand the relation between the complexity of the composed function in terms of the complexities of the individual functions. Let $\mathsf{D}(\cdot)$ denote the deterministic query complexity. It is easy to see that $\mathsf{D}(f \circ g^n) \leq \mathsf{D}(f) \cdot \mathsf{D}(g)$ since $f \circ g$ can be computed by simulating an optimal query algorithm of f; whenever the algorithm makes a query, we simulate an optimal query algorithm for g and serve the query. It can be shown by an adversary argument that this is an optimal query algorithm and $\mathsf{D}(f \circ g^n) = \mathsf{D}(f) \cdot D(g)$.

However, such a characterization is not so obvious for randomized query complexity. Although a similar upper bound still holds true (possibly accommodating a logarithmic overhead), it is no more as clear that the same bound also asymptotically bounds the randomized query complexity of $f \circ g^n$ from below. Let $\mathsf{R}_\epsilon(\cdot)$ denote the ϵ-error randomized query complexity. Our main theorem in this work is the following.

[*] Centre for Quantum Technologies, National University of Singapore, Singapore. a0109169@u.nus.edu

[†] Institute of Mathematics, Czech Academy of Sciences, Žitna 25, Praha 1, Czech Republic. Part of this work was done when Dmitry Gavinsky was visiting the Centre for Quantum Technologies at the National University of Singapore.

[‡] Centre for Quantum Technologies, National University of Singapore and MajuLab, UMI 3654, Singapore. rahul@comp.nus.edu.sg

[§] Centre for Quantum Technologies, National University of Singapore, Singapore. srijita.kundu@u.nus.edu

[¶] Division of Mathematical Sciences, Nanyang Technological University, Singapore and Centre for Quantum Technologies, National University of Singapore, Singapore. troylee@ntu.edu.sg

[‖] Centre for Quantum Technologies, National University of Singapore, Singapore. a0109168@u.nus.edu

[**] IRIF, Université Paris Diderot, CNRS, 75205 Paris, France, and Centre for Quantum Technologies, National University of Singapore, Singapore. santha@irif.fr

[††] Division of Mathematical Sciences, Nanyang Technological University, Singapore and Centre for Quantum Technologies, National University of Singapore, Singapore. ssanyal@ntu.edu.sg

Theorem 1 (Main Theorem)**.** *For any relation $f \subseteq \{0,1\}^n \times \mathcal{R}$ and Boolean function $g : \{0,1\}^m \to \{0,1\}$,*

$$\mathsf{R}_{1/3}(f \circ g^n) = \Omega(\mathsf{R}_{4/9}(f) \cdot \mathsf{R}_{1/2-1/n^4}(g)).$$

This implies that if g is a function that is hard to compute with an error $1/2 - 1/n^4$, the function $f \circ g$ is hard to compute with error $1/3$.

In the special case where f is a Boolean function, Theorem 1 implies that $\mathsf{R}_{1/3}(f \circ g^n) = \Omega(\mathsf{R}_{1/3}(f) \cdot \mathsf{R}_{1/2-1/n^4}(g))$, since the success probability of query algorithms for Boolean functions can be boosted from $5/9$ to $2/3$ by constantly many independent repetitions followed by taking a majority of the different outputs.

Composition theorem for randomized query complexity has been an area of active research in the past. Göös and Jayram [6] showed a composition theorem for a constrained version of conical juntas degree, which is a lower bound on randomized query complexity. Composition theorem for approximate degree (which also lower bounds randomized query complexity) for the special case of TRIBES function has seen a long line of research culminating in independent works of Sherstov [12] and Bun and Thaler [3] who settle the question by proving optimal bounds.

Composition theorem has been studied and shown in the context of communication and query complexities by the works of Göös, Pitassi and Watson [7, 8], Chattopadhyay et al. [4] when the function g is the indexing function or the inner product function with large enough arity. The work of Hatami, Hosseini and Lovett [9] proves a composition theorem in the context of communication and parity query complexites when the function g is the two-bit xor function. Ben-David and Kothari [1] proved a composition theorem for the *sabotage complexity* of Boolean function, a novel complexity measure defined in the same work that the authors prove to give quadratically tight bound on the randomized query complexity.

Composition theorems have also been successfully used in the past in constructing separating examples for various complexity measures, and bounding one complexity measure in terms of another. Kulkarni and Tal [10] proved an upper bound on fractional block sensitivity in terms of degree by analyzing the behavior of fractional block sensitivity under function composition. Separation between block sensitivity and degree was obtained by composing Kushilevitz's icosadedron function repeatedly with itself (see [2]). Separation between parity decision tree complexity and Fourier sparsity has been obtained by O'Donnell et al. by studying the behavior of *parity kill number* under function composition [11].

1.1 Our techniques

In this section, we give a high level overview of our proof of Theorem 1. Let $\delta = \frac{1}{n^4}$ and $\epsilon = 1/2 - \delta$. Let μ be the distribution over the domain $\{0,1\}^m$ of g for which $\mathsf{R}_\epsilon(g)$ is achieved, i.e., $\mathsf{R}_\epsilon(g) = \mathsf{D}^\mu_\epsilon(g)$. (see Fact 1)

We show that for every probability distribution λ over the domain $\{0,1\}^n$ of f, there exists a deterministic query algorithm $\mathcal{A}$ with worst case query complexity at most $\mathsf{R}_{1/3}(f \circ g)/\mathsf{R}_\epsilon(g)$, such that $\Pr_{z \sim \lambda}[(z, \mathcal{A}(z)) \in f] \geq 5/9$. By the minimax principle (Fact 1) this will prove Theorem 1.

Now we define a probability distribution γ over $(\{0,1\}^m)^n$. To define γ, we begin by defining a family of distributions $\{\gamma^z : z \in \{0,1\}^n\}$ over $(\{0,1\}^m)^n$. For a fixed $z \in \{0,1\}^n$, we define γ^z by giving a sampling procedure:

1. For each $i = 1, \ldots, n$, sample $x^{(i)} = (x^{(i)}_1, \ldots, x^{(i)}_m)$ from $\{0,1\}^m$ independently according to μ_{z_i}.
2. Return $x = (x^{(1)}, \ldots, x^{(n)})$.

Thus for $z = (z_1, \ldots, z_n) \in \{0,1\}^n$ and $x = (x^{(1)}, \ldots, x^{(n)}) \in (\{0,1\}^m)^n$, $\gamma^z(x) = \Pi_{i=1}^n \mu_{z_i}(x^{(i)})$. Note that γ^z is supported only on strings x for each $r \in \mathcal{R}$, $(x, r) \in f \circ g^n$ if and only if $(z, r) \in f$.

Having defined the distributions γ^z, we define the distribution γ by giving a sampling procedure:

1. Sample a $z = (z_1, \ldots, z_n)$ from $\{0,1\}^n$ according to λ.
2. Sample an $x = (x^{(1)}, \ldots, x^{(n)})$ from $(\{0,1\}^m)^n$ according to γ^z.

By minmax principle (Fact 1), there is a deterministic query algorithm $\mathcal{B}$ of worst case complexity at most $\mathsf{R}_{1/3}(f \circ g)$ such that $\Pr_{x \sim \gamma}[(x, \mathcal{B}(x)) \in f \circ g] \geq 2/3$. We will use $\mathcal{B}$ to construct a randomized query algorithm $\mathcal{A}'$ for f with the desired properties. A deterministic query algorithm $\mathcal{A}$ for f with required performance guarantees can then be obtained by appropriately fixing the randomness of $\mathcal{B}$.

See Algorithm 1 for a formal description of $\mathcal{A}'$. Given an input $z = (z_1, \ldots, z_n)$, $\mathcal{A}'$ simulates $\mathcal{B}$. Recall that an input to $\mathcal{B}$ is an nm bit long string $(x_j^{(i)})_{\substack{i=1,\ldots,n \\ j=1,\ldots,m}}$. Whenever $\mathcal{B}$ asks for (queries) an input bit $x_j^{(i)}$, a response bit is appropriately generated and passed to $\mathcal{B}$. To generate a response to a query by $\mathcal{B}$, a bit in z may be queried; such queries will contribute to the query complexity of $\mathcal{A}'$. The queries are addressed as follows. Let the simulation of $\mathcal{B}$ request bit $x_j^{(i)}$.

- If less than $\mathsf{D}_\epsilon^\mu(g)$ queries have been made into $x^{(i)}$ (including the current query) then a bit b is sampled from the marginal distribution of $x_j^{(i)}$ according to μ, conditioned on the responses to the past queries. b is passed to the simulation of $\mathcal{B}$.
- If at least $\mathsf{D}_\epsilon^\mu(g)$ queries have been made into $x^{(i)}$ (including the current query) then first the input bit z_i is queried; then a bit b is sampled from the marginal distribution of $x_j^{(i)}$ according to μ_{z_i}, conditioned on the responses to the past queries. b is passed to the simulation of $\mathcal{B}$.

The simulation of $\mathcal{B}$ continues until $\mathcal{B}$ terminates in a leaf. Then $\mathcal{A}'$ also terminates and outputs the label of the leaf.

We use Claims 3 and 4 to prove that for a fixed $z \in \{0,1\}^n$, the probability distribution induced by $\mathcal{A}'$ on the leaves of $\mathcal{B}$ is statistically close to the probability distribution induced by $\mathcal{B}$ on its leaves for a random input from γ^z. Averaging over different z's, the correctness of $\mathcal{A}'$ follows from the correctness of $\mathcal{B}$.

2 Preliminaries

In this section, we define some basic concepts, and set up our notations. We begin with defining the 2-sided error randomized and distributional query complexity measures of relations.

Definition 1 (2-sided Error Randomized Query Complexity)**.** Let $\mathcal{R}$ be any set. Let $f \subseteq \{0,1\}^n \times \mathcal{R}$ be any relation and $\epsilon \in [0, 1/2)$. The 2-sided error randomized query complexity $\mathsf{R}_\epsilon(f)$ is the minimum number of queries made in the worst case by a randomized query algorithm $\mathcal{A}$ (the worst case is over inputs and the internal randomness of the algorithm) that on each input x satisfies $\Pr[(x, \mathcal{A}(x)) \in f] \geq 1 - \epsilon$.

Definition 2 (Distributional Query Complexity)**.** Let $f \subseteq \{0,1\}^n \times \mathcal{R}$ be any relation, $b \in \{0,1\}$, μ a distribution on the domain $\{0,1\}^n$ of f, and $\epsilon \in [0, 1/2)$. The distributional query complexity $\mathsf{D}_\epsilon^\mu(f)$ is the minimum number of queries made in the worst case (over inputs) by a deterministic query algorithm $\mathcal{A}$ for which $\Pr_{x \sim \mu}[(x, \mathcal{A}(x)) \in f] \geq 1 - \epsilon$.

In particular, if f is a function and $\mathcal{A}$ is a randomized or distributional query algorithm computing f with error ϵ, then $\Pr_{x\sim\mu}[f(x) = \mathcal{A}(x)] \geq 1-\epsilon$.

The following theorem is Von Neumann's minimax principle stated for decision trees.

Fact 1 (minimax principle). *For any integer k, set $\mathcal{S}$, and relation $h \subseteq \{0,1\}^k \times \mathcal{S}$,*

$$R_\epsilon(h) = \max_\mu \mathsf{D}^\mu_\epsilon(h).$$

Let $g : \{0,1\}^m \to \{0,1\}$ be a Boolean function. Let μ be a probability distribution on $\{0,1\}^m$ which intersects non-trivially both with $g^{-1}(0)$ and with $g^{-1}(1)$. For each $z \in \{0,1\}$, let μ_z be the distribution obtained by restricting μ to $g^{-1}(z)$. In other words,

$$\mu_z(x) = \begin{cases} 0 & \text{if } g(x) \neq z \\ \frac{\mu(x)}{\sum_{y:g(y)=z}\mu(y)} & \text{if } g(x) = z \end{cases}$$

Notice that μ_0 and μ_1 are defined with respect to some Boolean function g, which will always be clear from the context.

For a positive integer m, $\mathcal{H}_m$ will stand for the m-dimensional Boolean hypercube $\{0,1\}^m$.

Definition 3 (Subcube, co-dimension). A subset $\mathcal{C}$ of $\mathcal{H}_m$ is called a subcube if there exists a set $S \subseteq \{1,\ldots,m\}$ of indices and *assignment function* $A : S \to \{0,1\}$ such that $\mathcal{C} = \{x \in \mathcal{H}_m : \forall i \in S, x_i = A(i)\}$. The codimension $\mathsf{codim}(\mathcal{C})$ of $\mathcal{C}$ is defined to be $|S|$.

Let $\mathcal{C} \subseteq \mathcal{H}_m$ be a subcube and μ be a probability distribution on $\mathcal{H}_m$. We will often abuse notation and use $\mathcal{C}$ to denote the event that a random string x belongs to the subcube $\mathcal{C}$. The probability $\Pr_{x\sim\mu}[x \in \mathcal{C}]$ will often be denoted by $\Pr_\mu[\mathcal{C}]$, and for two subcubes $\mathcal{C}_1$ and $\mathcal{C}_2$, the conditional probability $\Pr_{x\sim\mu}[x \in \mathcal{C}_2 \mid x \in \mathcal{C}_1]$ will be denoted by $\Pr_\mu[\mathcal{C}_2 \mid \mathcal{C}_1]$.

Definition 4 (Bias of a subcube). Let $g : \{0,1\}^m \to \{0,1\}$ be a Boolean function. Let μ be a probability distribution over $\{0,1\}^m$. Let $\mathcal{C} \subseteq \{0,1\}^m$ be a subcube such that $\Pr_{x\sim\mu}[x \in \mathcal{C}] > 0$. The bias of $\mathcal{C}$ with respect to μ, $\mathsf{bias}^\mu(\mathcal{C})$ is defined to be:

$$\mathsf{bias}^\mu(\mathcal{C}) = |\Pr[g(x) = 0 \mid x \in \mathcal{C}] - \Pr[g(x) = 1 \mid x \in \mathcal{C}]|.$$

A Boolean function g is implicit in the definition of bias, which will always be clear from the context.

Proposition 2. Let $g : \{0,1\}^m \to \{0,1\}$ be a Boolean functions, and $\mathsf{D}^\mu_\epsilon(g) > 0$. Then,

$$\min_{b\in\{0,1\}} \{\Pr_{x\sim\mu}[g(x) = b]\} > \epsilon.$$

In particular, $\mathsf{bias}^\mu(\mathcal{H}_m) < 1-2\epsilon$.

Proof. Towards a contradiction, assume that $\min_{b\in\{0,1\}}\{\Pr_{x\sim\mu}[g(x) = b]\} \leq \epsilon$. Then, the algorithm that outputs $\arg\max_{b\in\{0,1\}}\{\Pr_{x\sim\mu}[g(x) = b]\}$ makes 0 query and is correct with probability at least $1-\epsilon$. This contradicts the hypothesis that $\mathsf{D}^\mu_\epsilon > 0$. □

Now we define composition of two relations.

Definition 5 (Composition of relations). Let $f \subseteq \{0,1\}^n \times \mathcal{R}$ and $g \subseteq \{0,1\}^m \times \{0,1\}$ be two relations. The composed relation $f \circ g^n \subseteq (\{0,1\}^m)^n \times \mathcal{R}$ is defined as follows: For $x = (x^{(1)},\ldots,x^{(n)}) \in (\{0,1\}^m)^n$ and $r \in \mathcal{R}$, $(x,r) \in f \circ g^n$ if and only if there exists $b = (b^{(1)},\ldots,b^{(n)}) \in \{0,1\}^n$ such that for each $i = 1,\ldots,n$, $(x^{(i)}, b^{(i)}) \in g$ and $(b,r) \in f$.

We will often view a deterministic query algorithm as a binary decision tree. In each vertex v of the tree, an input variable is queried. Depending on the outcome of the query, the computation goes to a child of v. The child of v corresponding to outcome b to the query made is denoted by v_b. It is well known that the set of inputs that lead the computation of a decision tree to a certain vertex form a subcube. We will denote the subcube corresponding to a vertex v by $\mathcal{C}_v$.

We next prove two claims about bias, probability and codimension of subcubes that will be useful. Claim 3 states that for a function with large distributional query complexity, the bias of most shallow leaves of any deterministic query procedure is small.

Claim 3. Let $g : \{0,1\}^m \to \{0,1\}$ be a Boolean function. Let $\epsilon \in [1/4, 1/2)$ and let $\delta = 1/2 - \epsilon$. Let μ be a probability distribution on $\{0,1\}^m$, and $\mathsf{D}^\mu_\epsilon(g) = c > 0$. Let $\mathcal{B}$ be any deterministic query algorithm for m bit strings. For each $y \in \{0,1\}^m$, let ℓ_y be the unique leaf of $\mathcal{B}$ that contains y. Then,

(a) $\Pr_{y\sim\mu}[\mathsf{codim}(\ell_y) < c \text{ and } \mathsf{bias}^\mu(\ell_y) \geq 2\delta^{1/2}] < \delta^{1/2}$.

(b) For each $b \in \{0,1\}$, $\Pr_{y\sim\mu_b}[\mathsf{codim}(\ell_y) < c \text{ and } \mathsf{bias}^\mu(\ell_y) \geq 2\delta^{1/2}] < 4\delta^{1/2}$.

In the above claim $\mathcal{B}$ could just be a deterministic procedure that makes queries and eventually terminates; whether or not it makes any output upon termination is not of any consequence here.

Proof. We first show that part (a) implies part (b). To this end, assume part (a) and fix a $b \in \{0,1\}$. Let $a(y)$ be the indicator variable for the event $\mathsf{codim}(\ell_y) < c$ and $\mathsf{bias}^\mu(\ell_y) \geq 2\delta^{1/2}$. Thus, part (a) states that $\Pr_{y\sim\mu}[a(y) = 1] < \delta^{1/2}$. Now,

$$\begin{aligned}
&\Pr_{y\sim\mu_b}[\mathsf{codim}(\ell_y) < c \text{ and } \mathsf{bias}^\mu(\ell_y) \geq 2\delta^{1/2}] \\
&= \sum_{y:a(y)=1} \mu_b(y) \\
&= \frac{1}{\sum_{y:f(y)=b}\mu(y)} \sum_{y:a(y)=1} \mu(y) \quad \text{(From the definition of } \mu_b) \\
&\leq \frac{1}{\epsilon} \Pr_{y\sim\mu}[a(y) = 1] \quad \text{(From Proposition 2)} \\
&< 4\delta^{1/2}. \quad \text{(By the hypothesis } \epsilon \geq 1/4 \text{ and part (a))}
\end{aligned}$$

We now prove part (a). Towards a contradiction assume that

$$\Pr_{y\sim\mu}[\mathsf{codim}(\ell_y) < c \text{ and } \mathsf{bias}^\mu(\ell_y) \geq 2\delta^{1/2}] \geq \delta^{1/2}.$$

Now consider the following decision tree algorithm $\mathcal{A}$ on m bit strings:

Begin simulating $\mathcal{B}$. Let $\mathcal{C}$ be the subcube associated with the current node of $\mathcal{B}$ in the simulation. Simulate $\mathcal{B}$ unless one of the following happens.

- $\mathcal{B}$ terminates.
- The number of queries made is $c - 1$.
- $\mathsf{bias}^\mu(\mathcal{C}) \geq 2\delta^{1/2}$.

Upon termination, If $\mathsf{bias}^\mu(\mathcal{C}) \geq 2\delta^{1/2}$, output $\arg\max_{b\in\{0,1\}} \Pr_{y\sim\mu}[g(y) = b \mid y \in \mathcal{C}]$. Else output a uniformly random bit.

It immediately follows that the worst case query complexity of $\mathcal{A}$ is at most $c-1$. Now, we will prove that $\Pr_{y\sim\mu}[\mathcal{A}(y)=g(y)] \geq 1-\epsilon$. This will contradict the hypothesis that $\mathsf{D}^{\mu}_{\epsilon}(g)=c$. Let $\mathcal{L}$ be the node of $\mathcal{B}$ at which the computation of $\mathcal{A}$ ends. Let $\Pr_{y\sim\mu}[\mathsf{bias}^{\mu}(\mathcal{L}) \geq 2\delta^{1/2}] = p$. By our assumption, $p \geq \delta^{1/2}$.

$$\begin{aligned}
&\Pr_{y\sim\mu}[\mathcal{A}(y)=g(y)] \\
&= \Pr_{y\sim\mu}[\mathsf{bias}^{\mu}(\mathcal{L}) \geq 2\delta^{1/2}] \cdot \Pr_{y\sim\mu}[\mathcal{A}(y)=g(y) \mid \mathsf{bias}^{\mu}(\mathcal{L}) \geq 2\delta^{1/2}] + \\
&\qquad\qquad \Pr_{y\sim\mu}[\mathsf{bias}^{\mu}(\mathcal{L}) < 2\delta^{1/2}] \cdot \Pr_{y\sim\mu}[\mathcal{A}(y)=g(y) \mid \mathsf{bias}^{\mu}(\mathcal{L}) < 2\delta^{1/2}] \\
&\geq p \cdot (1/2+\delta^{1/2}) + (1-p).\frac{1}{2} \quad \text{(from our assumption)} \\
&= 1/2 + p\cdot\delta^{1/2} \\
&\geq 1/2+\delta \quad (\text{since } p \geq \delta^{1/2}) \\
&= 1-\epsilon.
\end{aligned}$$

This completes the proof. □

The next claim states that if a subcube has low bias with respect to a distribution μ, then the distributions μ_0 and μ_1 ascribe almost the same probability to it.

Claim 4. Let $g:\{0,1\}^m \to \{0,1\}$ be a Boolean function and $\delta \in (0,\frac{1}{2}]$. Let μ be a distribution on $\{0,1\}^m$. Let $\mathcal{C}$ be a subcube such that $\Pr_{\mu}[\mathcal{C}] > 0$ and $\mathsf{bias}^{\mu}(\mathcal{C}) \leq \delta$. Also assume that $\mathsf{bias}^{\mu}(\mathcal{H}_m) \leq \delta$. Then for any $z \in \{0,1\}$ we have,

(a) $\Pr_{\mu}[\mathcal{C}] \leq (1+4\delta)\cdot \Pr_{\mu_z}[\mathcal{C}]$,

(b) $\Pr_{\mu}[\mathcal{C}] \geq (1-4\delta)\cdot \Pr_{\mu_z}[\mathcal{C}]$.

Proof. We prove part (a) of the claim. The proof of part (b) is similar.

By the definition of bias and the hypothesis, for each $z \in \{0,1\}$,

$$\sum_{x:g(x)=z} \mu(x) \leq \left(\frac{1}{2}+\frac{\delta}{2}\right)\cdot \sum_{x}\mu(x), \tag{1}$$

$$\sum_{x\in\mathcal{C}:g(x)=z} \mu(x) \geq \left(\frac{1}{2}-\frac{\delta}{2}\right)\cdot \sum_{x\in\mathcal{C}}\mu(x) > 0. \tag{2}$$

Now,

$$\begin{aligned}
\Pr_{\mu_z}[\mathcal{C}] &= \sum_{x\in\mathcal{C}} \mu_z(x) \\
&= \frac{\sum_{x\in\mathcal{C}:g(x)=z}\mu(x)}{\sum_{x:g(x)=z}\mu(x)} \\
&\geq \frac{(1/2-\delta/2)\cdot\sum_{x\in\mathcal{C}}\mu(x)}{1/2+\delta/2} \quad \text{(From Equations 1 and 2)} \\
&= \frac{1/2-\delta/2}{1/2+\delta/2}\cdot\Pr_{\mu}[\mathcal{C}]
\end{aligned}$$

Thus,

$$\Pr_{\mu}[\mathcal{C}] \leq \frac{1/2+\delta/2}{1/2-\delta/2}\cdot\Pr_{\mu_z}[\mathcal{C}] \leq (1+4\delta)\cdot\Pr_{\mu_z}[\mathcal{C}]. \quad (\text{since } \delta \leq \tfrac{1}{2})$$

□

3 Composition Theorem

In this section we prove our main theorem. We restate it below.

Theorem 1 (Main Theorem)**.** *For any relation $f \subseteq \{0,1\}^n \times \mathcal{R}$ and Boolean function $g : \{0,1\}^m \to \{0,1\}$,*

$$\mathsf{R}_{1/3}(f \circ g^n) = \Omega(\mathsf{R}_{4/9}(f) \cdot \mathsf{R}_{1/2-1/n^4}(g)).$$

Proof. We begin by recalling the notations defined in Section 1.1 that we will use in this proof.

Let $\delta = \frac{1}{n^4}$ and $\epsilon = 1/2 - \delta$. Let μ be the distribution over the domain $\{0,1\}^m$ of g for which $\mathsf{R}_\epsilon(g)$ is achieved, i.e., $\mathsf{R}_\epsilon(g) = \mathsf{D}^\mu_\epsilon(g)$. (see Fact 1)

We show that for every probability distribution λ over the domain $\{0,1\}^n$ of f, there exists a deterministic query algorithm $\mathcal{A}$ with worst case query complexity at most $\mathsf{R}_{1/3}(f \circ g)/\mathsf{R}_\epsilon(g)$, such that $\Pr_{z\sim\lambda}[(z, \mathcal{A}(z)) \in f] \geq 5/9$. By the minimax principle (Fact 1) this will prove Theorem 1.

We define a probability distribution γ over $(\{0,1\}^m)^n$. We first define a family of distributions $\{\gamma^z : z \in \{0,1\}^n\}$ over $(\{0,1\}^m)^n$. For a fixed $z \in \{0,1\}^n$, we define γ^z by giving a sampling procedure:

1. For each $i = 1, \ldots, n$, sample $x^{(i)} = (x^{(i)}_1, \ldots, x^{(i)}_m)$ from $\{0,1\}^m$ independently according to μ_{z_i}.
2. Return $x = (x^{(1)}, \ldots, x^{(n)})$.

Thus for $z = (z_1, \ldots, z_n) \in \{0,1\}^n$ and $x = (x^{(1)}, \ldots, x^{(n)}) \in (\{0,1\}^m)^n$, $\gamma^z(x) = \Pi_{i=1}^n \mu_{z_i}(x^{(i)})$. Note that γ^z is supported only on strings x for each $r \in \mathcal{R}$, $(x, r) \in f \circ g^n$ if and only if $(z, r) \in f$.

Now, we define the distribution γ by giving a sampling procedure:

1. Sample a $z = (z_1, \ldots, z_n)$ from $\{0,1\}^n$ according to λ.
2. Sample an $x = (x^{(1)}, \ldots, x^{(n)})$ from $(\{0,1\}^m)^n$ according to γ^z.

By the minmax principle (Fact 1), there is a deterministic query algorithm $\mathcal{B}$ of worst case complexity at most $\mathsf{R}_{1/3}(f \circ g)$ such that $\Pr_{x\sim\gamma}[(x, \mathcal{B}(x)) \in f \circ g] \geq 2/3$. We will use $\mathcal{B}$ to construct a randomized query algorithm $\mathcal{A}'$ for f with the desired properties. A deterministic query algorithm $\mathcal{A}$ for f with required performance guarantees can then be obtained by appropriately fixing the randomness of $\mathcal{B}$. Algorithm 1 formally defines the algorithm $\mathcal{A}'$ that we construct.

From the definition of bias one can verify that the events in steps 5 and 10 in Algorithm 1 that are being conditioned on, have non-zero probabilities under the respective distributions; hence, the probabilistic processes are well-defined.

From the description of $\mathcal{A}'$ it is immediate that z_i is queried only if the underlying simulation of $\mathcal{B}$ queries at least $\mathsf{R}_\epsilon(g)$ locations in $x^{(i)}$. Thus the worst-case query complexity of $\mathcal{A}'$ is at most $\mathsf{R}_{1/3}(f \circ g^n)/\mathsf{R}_\epsilon(g)$.

We are left with the task of bounding the error of $\mathcal{A}'$. Let $\mathcal{L}$ be the set of leaves of the decision tree $\mathcal{B}$. Each leaf $\ell \in \mathcal{L}$ is labelled with a bit $b_\ell \in \{0,1\}$; whenever the computation reaches ℓ, the bit b_ℓ is output.

For a vertex v, let the corresponding subcube $\mathcal{C}_v$ be $\times_{i=1}^n \mathcal{C}_v^{(i)}$, where $\mathcal{C}_v^{(i)}$ is a subsube of the domain of the i-th copy of g (corresponding to the input $x^{(i)}$).

For each leaf $\ell \in \mathcal{L}$ and $i = 1, \ldots, n$, define $\mathsf{snip}^{(i)}(\ell)$ to be 1 if there is a node t in the unique path from the root of $\mathcal{B}$ to ℓ such that $\mathsf{codim}(\mathcal{C}_t^{(i)}) < \mathsf{D}^\mu_\epsilon$ and $\mathsf{bias}^\mu(C_t^{(i)}) \geq \frac{2}{n^2}$. Define $\mathsf{snip}^{(i)}(\ell) = 0$ otherwise. Define $\mathsf{snip}(\ell) = \vee_{i=1}^n \mathsf{snip}^{(i)}(\ell)$.

Algorithm 1: Randomized query algorithm $\mathcal{A}'$ for f

```
Input: $z \in \{0,1\}^n$
Initialize $v \leftarrow$ root of the decision tree $\mathcal{B}$, $Q \leftarrow \emptyset$
while $v$ is not a leaf do
    Let a bit in $x^{(i)}$ be queried at $v$
    if $i \notin Q$ then                /* $\mathsf{codim}(\mathcal{C}_v^{(i)}) < D_\epsilon^\mu(g)$ if this is satisfied */
        Set $v \leftarrow v_b$ with probability $\Pr_\mu[\mathcal{C}_{v_b} \mid \mathcal{C}_v]$
        if $\mathsf{codim}(\mathcal{C}_v^{(i)}) = D_\epsilon^\mu(g)$ then
            Query $z_i$
            Set $Q = Q \cup \{i\}$
    else
        Set $v \leftarrow v_b$ with probability $\Pr_{\mu_{z_i}}[\mathcal{C}_{v_b} \mid \mathcal{C}_v]$
Output label of $v$.
```

For each $\ell \in \mathcal{L}$, define p_ℓ^z to be the probability that for an input drawn from γ^z, the computation of $\mathcal{B}$ terminates at leaf ℓ. We have,

$$\Pr_{x \sim \gamma^z}[(x, \mathcal{B}(x)) \in f \circ g^n] = \Pr_{x \sim \gamma^z}[(z, \mathcal{B}(x)) \in f] = \sum_{\ell \in \mathcal{L} : (z, b_\ell) \in f} p_\ell^z. \tag{3}$$

From our assumption about $\mathcal{B}$ we also have that,

$$\Pr_{x \sim \gamma}[(x, \mathcal{B}(x)) \in f \circ g^n] = \mathop{\mathbb{E}}_{z \sim \lambda} \Pr_{x \sim \gamma^z}[(x, \mathcal{B}(x)) \in f \circ g^n] \geq \frac{2}{3}. \tag{4}$$

Now, consider a run of $\mathcal{A}'$ on z. For each $\ell \in \mathcal{L}$ of $\mathcal{B}$, define q_ℓ^z to be the probability that the computation of $\mathcal{A}'$ on z terminates at leaf ℓ of $\mathcal{B}$. Note that the probability is over the internal randomness of $\mathcal{A}'$.

To finish the proof, we need the following two claims. The first one states that the leaves $\ell \in \mathcal{L}$ are sampled with similar probabilities by $\mathcal{B}$ and $\mathcal{A}'$.

Claim 5. For each $\ell \in \mathcal{L}$ such that $\mathsf{snip}(\ell) = 0$, and for each $z \in \{0,1\}^n$, $\frac{8}{9} \cdot p_\ell^z \leq q_\ell^z \leq \frac{10}{9} \cdot p_\ell^z$.

The next Claim states that for each z, the probability according to γ^z of the leaves ℓ for which $\mathsf{snip}(\ell) = 1$ is small.

Claim 6.

$$\forall z \in \{0,1\}^n, \sum_{\ell \in \mathcal{L}, \mathsf{snip}(\ell) = 1} p_\ell^z \leq \frac{4}{n}.$$

We first finish the proof of Theorem 1 assuming Claim 5 and 6, and then prove the claims. For a fixed input $z \in \{0,1\}^n$, the probability that $\mathcal{A}'$, when run on z, outputs an r such that $(z, r) \in f$, is at least,

$$\begin{aligned} \sum_{\substack{\ell \in \mathcal{L} \\ (z, b_\ell) \in f, \mathsf{snip}(\ell) = 0}} q_\ell^z &\geq \sum_{\substack{\ell \in \mathcal{L} \\ (z, b_\ell) \in f, \mathsf{snip}(\ell) = 0}} \frac{8}{9} \cdot p_\ell^z \quad \text{(By Claim 5)} \\ &= \frac{8}{9} \left(\sum_{\substack{\ell \in \mathcal{L} \\ (z, b_\ell) \in f}} p_\ell^z - \sum_{\substack{\ell \in \mathcal{L} \\ (z, b_\ell) \in f, \mathsf{snip}(\ell) = 1}} p_\ell^z \right) \end{aligned}$$

$$\geq \frac{8}{9}\left(\sum_{\substack{\ell\in\mathcal{L}\\(z,b_\ell)\in f}} p^z_\ell - \frac{4}{n}\right). \quad \text{(By Claim 6)} \tag{5}$$

Thus, the success probability of $\mathcal{A}'$ is at least

$$\begin{aligned}
\mathop{\mathbb{E}}_{z\sim\lambda} \sum_{\substack{\ell\in\mathcal{L}\\(z,b_\ell)\in f,\mathsf{snip}(\ell)=0}} q^z_\ell \geq & \frac{8}{9}\cdot\left(\mathop{\mathbb{E}}_{z\sim\lambda}\sum_{\substack{\ell\in\mathcal{L}\\(z,b_\ell)\in f}} p^z_\ell - \frac{4}{n}\right) \quad \text{(By Equation (5))}\\
&\geq \frac{8}{9}\cdot\left(\frac{2}{3}-\frac{4}{n}\right) \quad \text{(By Equations (3) and (4))}\\
&\geq \frac{5}{9}. \quad \text{(For large enough } n\text{)}
\end{aligned}$$

We now give the proofs of Claims 5 and 6.

Proof of Claim 5. We will prove the first inequality. The proof of the second inequality is similar.

Fix a $z \in \{0,1\}^n$ and a leaf $\ell \in \mathcal{L}$. For each $i = 1,\ldots,n$, assume that $\mathsf{codim}(\mathcal{C}^{(i)}_\ell) = d^{(i)}$, and in the path from the root of $\mathcal{B}$ to ℓ the variables $x^{(i)}_1,\ldots,x^{(i)}_{d^{(i)}}$ are set to bits $b_1,\ldots,b_{d^{(i)}}$ in this order. The computation of $\mathcal{A}'$ terminates at leaf ℓ if the values of the different bits $x^{(i)}_j$ sampled by $\mathcal{A}'$ agree with the leaf ℓ. The probability of that happening is given by

$$q^z_\ell = \prod_{i=1}^n \Pr_{\mathcal{A}'}[x^{(i)}_1 = b_1,\ldots,x^{(i)}_{d^{(i)}} = b_{d^{(i)}} \mid z] \tag{6}$$

$$\begin{aligned}
&= \prod_{i=1}^n \Pr_{x\sim\mu}[x^{(i)}_1 = b_1,\ldots,x^{(i)}_{\mathsf{D}^\mu_\epsilon-1} = b_{\mathsf{D}^\mu_\epsilon-1}]\cdot\\
&\qquad \Pr_{x\sim\mu_{z_i}}[x^{(i)}_{D^\mu_\epsilon} = b_{D^\mu_\epsilon},\ldots,x^{(i)}_{d^{(i)}} = b_{d^{(i)}} \mid x^{(i)}_1 = b_1,\ldots,x^{(i)}_{\mathsf{D}^\mu_\epsilon-1} = b_{\mathsf{D}^\mu_\epsilon-1}].
\end{aligned} \tag{7}$$

The second equality above follows from the observation that in Algorithm 1, the first $\mathsf{D}^\mu_\epsilon - 1$ bits of $x^{(i)}$ are sampled from their marginal distributions with respect to μ, and the subsequent bits are sampled from their marginal distributions with respect to μ_{z_i}.

In equation (7), $\Pr_{x\sim\mu_{z_i}}[x^{(i)}_{D^\mu_\epsilon} = b_{D^\mu_\epsilon},\ldots,x^{(i)}_{d^{(i)}} = b_{d^{(i)}} \mid x^{(i)}_1 = b_1,\ldots,x^{(i)}_{\mathsf{D}^\mu_\epsilon-1} = b_{\mathsf{D}^\mu_\epsilon-1}]$ is interpreted as 1 if $d^{(i)} < \mathsf{D}^\mu_\epsilon$. Continuing from Equation (7), by Claim 4(b) (notice that since $\mathsf{D}^\mu_\epsilon(g) > 0$, by Proposition 2 the bias of $\mathcal{H}_m$ is at most $2/n^4$) and the assumption that $\mathsf{snip}(\ell) = 0$ we have,

$$\begin{aligned}
q^z_\ell \geq & \prod_{i=1}^n (1-8/n^4) \Pr_{x\sim\mu_{z_i}}[x^{(i)}_1 = b_1,\ldots,x^{(i)}_{\mathsf{D}^\mu_\epsilon-1} = b_{\mathsf{D}^\mu_\epsilon-1}]\cdot\\
&\qquad \Pr_{x\sim\mu_{z_i}}[x^{(i)}_{D^\mu_\epsilon} = b_{D^\mu_\epsilon},\ldots,x^{(i)}_{d^{(i)}} = b_{d^{(i)}} \mid x^{(i)}_1 = b_1,\ldots,x^{(i)}_{\mathsf{D}^\mu_\epsilon-1} = b_{\mathsf{D}^\mu_\epsilon-1}]\\
&= (1-8/n^4)^n \prod_{i=1}^n \Pr_{x\sim\mu_{z_i}}[x^{(i)}_1 = b_1,\ldots,x^{(i)}_{d^{(i)}} = b_{d^{(i)}}]\\
&\geq \frac{8}{9}\cdot p^z_\ell. \quad \text{(For large enough n)}
\end{aligned}$$

□

Proof of Claim 6. Fix a $z \in \{0,1\}^n$. We shall prove that for each i, $\sum_{\ell\in\mathcal{L},\mathsf{snip}^{(i)}(\ell)=1} p^z_\ell \leq \frac{1}{n^2}$. That will prove the claim, since $\sum_{\ell\in\mathcal{L},\mathsf{snip}(\ell)=1} p^z_\ell \leq \sum_{i=1}^n \sum_{\ell\in\mathcal{L},\mathsf{snip}^{(i)}(\ell)=1} p^z_\ell$.

To this end, fix an $i \in \{1, \ldots, n\}$. For a random x drawn from γ^z, let p be the probability that in strictly less than D^μ_ϵ queries the computation of $\mathcal{B}$ reaches a node t such that $\mathsf{bias}(\mathcal{C}_t^{(i)})$ is at least $\frac{2}{n^2}$. Note that this probability is over the choice of the different $x^{(i)}$'s. We shall show that $p \le \frac{1}{n^2}$. This is equivalent to showing that $\sum_{\ell \in \mathcal{L}, \mathsf{snip}^{(i)}(\ell)=1} p_\ell^z \le \frac{1}{n^2}$.

Note that each $x^{(j)}$ is independently distributed according to μ_{z_j}. By averaging, there is a choice of $x^{(j)}$ for each $j \neq i$ such that for a random $x^{(i)}$ chosen according to μ_{z_i}, a node t as above is reached within at most $\mathsf{D}^\mu_\epsilon - 1$ steps with probability at least p. Claim 6 follows from Claim 3(b) with $\epsilon = \frac{1}{2} - \frac{1}{n^4}$ (note that $\epsilon \ge \frac{1}{4}$ for large enough n). □

This completes the proof of Theorem 1. □

3.1 Hardness Amplification Using XOR Lemma

Theorem 1 is useful only when the function g is hard against randomized query algorithms even for error $1/2 - 1/n^4$. In this section we show a procedure that, given any g that is hard against randomized query algorithms with error $1/3$, obtains another function on a slightly larger domain that is hard against randomized query algorithms with error $1/2 - 1/n^4$.

Let $g : \{0,1\}^m \to \{0,1\}$ be a function. Let $g_t^\oplus : (\{0,1\}^m)^t \to \{0,1\}$ is defined as follows. For $x = (x^{(1)}, \ldots, x^{(t)}) \in (\{0,1\}^m)^t$,

$$g_t^\oplus(x) = \oplus_{i=1}^t g(x^{(i)}).$$

The following theorem is obtained by specializing Theorem 1 of Andrew Drucker's paper [5] to this setting.

Theorem 7 (Drucker 2011)**.**

$$\mathsf{R}_{1/2 - 2^{-\Omega(t)}}(g_t^\oplus) = \Omega(t \cdot \mathsf{R}_{1/3}(g)).$$

Setting $t = O(\log n)$ and combining Theorem 7 with Theorem 1 we have that,

Theorem 8.

$$\mathsf{R}_{1/3}\left(f \circ \left(g^\oplus_{O(\log n)}\right)^n\right) = \Omega(\log n \cdot \mathsf{R}_{4/9}(f) \cdot \mathsf{R}_{1/3}(g)).$$

Acknowledgements: This work was partially supported by the National Research Foundation, including under NRF RF Award No. NRF-NRFF2013-13, the Prime Ministers Office, Singapore and the Ministry of Education, Singapore under the Research Centres of Excellence programme and by Grant No. MOE2012-T3-1- 009.

D.G. is partially funded by the grant P202/12/G061 of GA ČR and by RVO: 67985840. M. S. is partially funded by the ANR Blanc program under contract ANR-12-BS02-005 (RDAM project).

References

[1] Shalev Ben-David and Robin Kothari. Randomized query complexity of sabotaged and composed functions. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 60:1–60:14, 2016.

[2] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: a survey. *Theor. Comput. Sci.*, 288(1):21–43, 2002.

[3] Mark Bun and Justin Thaler. Dual lower bounds for approximate degree and markov-bernstein inequalities. In *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part I*, pages 303–314, 2013.

[4] Arkadev Chattopadhyay, Michal Koucký, Bruno Loff, and Sagnik Mukhopadhyay. Simulation theorems via pseudorandom properties. *CoRR*, abs/1704.06807, 2017.

[5] Andrew Drucker. Improved direct product theorems for randomized query complexity. In *Proceedings of the 26th Annual IEEE Conference on Computational Complexity, CCC 2011, San Jose, California, June 8-10, 2011*, pages 1–11, 2011.

[6] Mika Göös and T. S. Jayram. A composition theorem for conical juntas. In *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan*, pages 5:1–5:16, 2016.

[7] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1077–1088, 2015.

[8] Mika Göös, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for BPP. *CoRR*, abs/1703.07666, 2017.

[9] Hamed Hatami, Kaave Hosseini, and Shachar Lovett. Structure of protocols for XOR functions. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 282–288, 2016.

[10] Raghav Kulkarni and Avishay Tal. On fractional block sensitivity. *Chicago J. Theor. Comput. Sci.*, 2016, 2016.

[11] Ryan O'Donnell, John Wright, Yu Zhao, Xiaorui Sun, and Li-Yang Tan. A composition theorem for parity kill number. In *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014*, pages 144–154, 2014.

[12] Alexander A. Sherstov. Approximating the AND-OR tree. *Theory of Computing*, 9:653–663, 2013.