On the randomised query complexity of composition

Dmitry Gavinsky[*†] Troy Lee[‡†] Miklos Santha[§†]

January 7, 2018

Abstract

Let $f \subseteq \{0,1\}^n \times \Xi$ be a relation and $g : \{0,1\}^m \to \{0,1,*\}$ be a promise function. This work investigates the randomised query complexity of the relation $f \circ g^n \subseteq \{0,1\}^{m \cdot n} \times \Xi$, which can be viewed as one of the most general cases of composition in the query model (letting g be a relation seems to result in a rather unnatural definition of $f \circ g^n$).

We show that for every such f and g,

$$\mathcal{R}(f \circ g^n) \in \Omega\Big(\mathcal{R}(f) \cdot \sqrt{\mathcal{R}(g)}\Big),$$

where $\mathcal{R}$ denotes the randomised query complexity. On the other hand, we demonstrate a relation f_0 and a promise function g_0, such that $\mathcal{R}(f_0) \in \Theta(\sqrt{n})$, $\mathcal{R}(g_0) \in \Theta(n)$ and $\mathcal{R}(f_0 \circ g_0^n) \in \Theta(n)$ – that is, our composition statement is tight.

To the best of our knowledge, there was no known composition theorem for the randomised query complexity of relations or promise functions (and for the special case of total functions our lower bound gives multiplicative improvement of $\sqrt{\log n}$).

1 Introduction

Let $f \subseteq \{0,1\}^n \times \Xi$ be a relational problem over n-bit input strings, where Ξ is a finite set and $\xi \in \Xi$ is a correct answer to $f(z)$ if and only if $(z, \xi) \in f$ – in that case we write $\xi \in f(z)$, thus viewing $f(z)$ as the set of correct answers. Let $g : \{0,1\}^m \to \{0,1,*\}$ be a partial function over m-bit input strings, where "$*$" marks "forbidden" input strings – in other words, g is a Boolean promise function. We will call $x \in \{0,1\}^m$ a *legal* input value for g if $g(x) \in \{0,1\}$.

Define $f \circ g^n \subseteq \{0,1\}^{n \cdot m} \times \Xi$ as a relational problem over $n \cdot m$-bit strings $x = (x_1, \ldots, x_n)$, where

$$\begin{cases} f \circ g^n(x) = \Xi & \text{if } * \in \big\{g(x_i) \big| i \in [n]\big\}; \\ f \circ g^n(x) = f(g(x_1), \ldots, g(x_n)) & \text{otherwise.} \end{cases}$$

[*]Institute of Mathematics, Czech Academy of Sciences, Žitna 25, Praha 1, Czech Republic. Partially supported by the Grant No. P202/12/G061 of GA ČR and by RVO: 67985840. Part of this work was done while visiting the Centre for Quantum Technologies at the National University of Singapore.

[†]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.

[‡]Division of Mathematical Sciences, Nanyang Technological University, Singapore and Centre for Quantum Technologies, National University of Singapore, Singapore. mailto:troyjlee@gmail.com.

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

That is, if at least one of x_i-s as input to $g(\cdot)$ violates the promise, then any $\xi \in \Xi$ is a correct answer to $f \circ g^n(x)$; otherwise, $f \circ g^n(x)$ is defined naturally.

We investigate the randomised query complexity ($\mathcal{R}$) of the relation $f \circ g^n$. This setting can be viewed as one of the most general cases of so-called composition questions in the model of randomised query complexity: Arguably, the most general natural way of modelling a computational problem in the query model is via a relation; on the other hand, letting the "bottom" problem g be a relation seems to result in a rather awkward definition of the composed problem $f \circ g^n$ – so, we've chosen to restrict g by making it a promise function (the "top" problem f may be a relation).[1]

We show that for *every* such f and g,

$$\mathcal{R}(f \circ g^n) \in \Omega\Big(\mathcal{R}(f) \cdot \sqrt{\mathcal{R}(g)}\Big).$$

On the other hand, we demonstrate a relation f_0 and a promise function g_0, such that $\mathcal{R}(f_0) \in \Theta(\sqrt{n})$, $\mathcal{R}(g_0) \in \Theta(n)$ and $\mathcal{R}(f_0 \circ g_0^n) \in \Theta(n)$ – that is, our composition statement is tight.[2]

Previous work

To the best of our knowledge, prior to this work no general lower bound was known for the randomised query complexity of composed promise functions or relations. For the special case of g being a total function, Ben-David and Kothari [BDK16] have shown that

$$\mathcal{R}(f \circ g^n) \in \Omega\left(\mathcal{R}(f) \cdot \sqrt{\frac{\mathcal{R}(g)}{\log\left(\mathcal{R}(g)\right)}}\right).$$

Our approach

To argue that $\mathcal{R}(f \circ g^n) \in \Omega\Big(\mathcal{R}(f) \cdot \sqrt{\mathcal{R}(g)}\Big)$, we will assume that a protocol for computing $f \circ g^n$ is given and use it to construct a protocol for computing f. Our construction will be such that the new protocol will be accurate (with respect to f) if the given protocol was accurate (with respect to $f \circ g^n$), and the query complexity of the new protocol will be small if that of the given protocol was small. As the query complexity of a protocol computing f cannot be below $\mathcal{R}(f)$, a lower bound on the query complexity of the given protocol for $f \circ g^n$ will follow.

2 Polarised protocol trees

In this part we describe a tree-like primitive for representing query protocols. Some special properties of this representation will be useful for our analysis.

Let $f \subseteq \{0,1\}^n \times \Xi$, $g : \{0,1\}^m \to \{0,1,*\}$ and $f \circ g^n \subseteq \{0,1\}^{n \cdot m} \times \Xi$ be as described above. Denote by $X = (X_1, \ldots, X_n)$ the input to $f \circ g^n$, where every $X_i \in \{0,1\}^m$ is input to

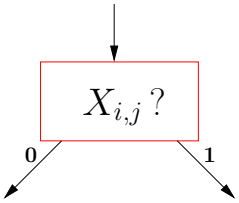
[1] Our lower bound argument would probably generalise to the case of both f and g being relations, though we haven't verified that.

[2] It may be worth noticing that the same example witnesses the possibility of $\mathcal{R}(f \circ g^n) \in O(\mathcal{R}(g))$ when $\mathcal{R}(f) \in \Omega(\sqrt{n})$.

g (we will write $X_{i,j}$ to address the j'th bit of X_i). Denote by $Z \in \{0,1,*\}^n$ the input to f – in other words, $\forall i:\ Z_i = g(X_i)$.

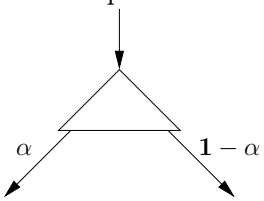
All protocol trees will have leaves, usually labelled with the answer returned by the protocol upon reaching this leaf.

A protocol for $f \circ g^n$ queries the values of $X_{i,j}$, so it will be represented by a tree, containing naturally defined nodes



Leaves and X-queries are the only types of nodes that will occur in a tree representing a protocol for $f \circ g^n(X_1, \ldots, X_n)$. Note that the actions corresponding to these nodes are *deterministic*, and such will be our protocols for $f \circ g^n$.

Our protocols for f will use *randomness*, represented by *randomised forks* in a tree:

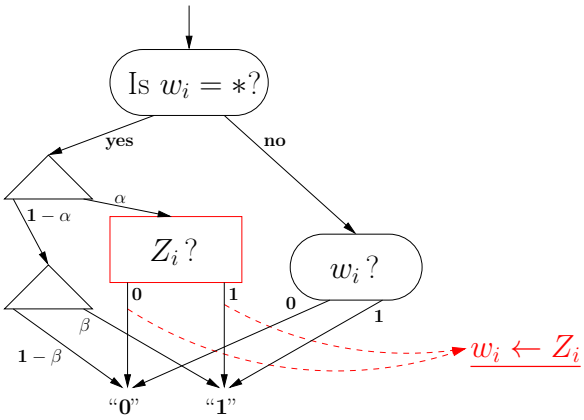


Here the left son is selected with probability α and the right one with probability $1-\alpha$.

As we describe next, our protocol trees for f will use *generic* nodes, somewhat non-standard: Usually a vertex in a tree corresponds to certain state of the protocol and fully determines the "history" at that state – namely, which queries have been made so far and what answers have been received from the oracle. For reasons that will become clear later, our trees will have vertices corresponding to "randomised Z-queries", where certain Z_i is queried with some probability between 0 and 1 – accordingly, a pointer to a vertex coming after such "uncertain query" does not reveal whether Z_i has actually been queried. To address this, we accompany our trees for computing $f(Z)$ with "memory" $w \in \{0,1,*\}^n$, such that $w_i = Z_i$ if Z_i has already been queried by the protocol (in which case we will always assume $Z_i \in \{0,1\}$, as explained next) and $w_i = *$ otherwise.

Our trees will have two types of generic vertices, corresponding to certain complex protocol's actions. The reason why we prefer to treat these complex actions as single tree nodes is the following: Recall that we will construct a protocol for f, based on a (given) protocol for $f \circ g^n$; each generic vertex in the new protocol will correspond to a single $X_{i,j}$-query in the original protocol; as a result, the tree of the constructed protocol for f will be *isomorphic* to the tree of the given protocol for $f \circ g^n$. Allowing generic vertices and using the auxiliary registers w_i are the "price" of keeping this very convenient isomorphism between the original and the constructed protocol trees.

The first generic type is a *Z-node*, described by $(i, \alpha, \beta) \in [n] \times [0,1] \times [0,1]$:

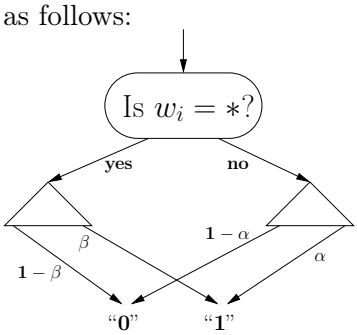


In words, this generic node corresponds to the following action:

- If $w_i = *$, then with probability α a query is made to read the value of Z_i and one of the two outgoing edges is selected accordingly; otherwise (i.e., with probability $1-\alpha$), the outgoing edge marked with "1" is selected with probability β and the edge marked with "0" is selected otherwise (i.e., with probability $1-\beta$).
- If $w_i \neq *$, then the outgoing edge is selected according to the (already known) value $Z_i = w_i$.
- If a query to Z_i has been made (which only occurs with probability α when $w_i = *$), then the value of the register w_i is updated to contain the value of Z_i.

Observe that in the above description we have assumed that if a query to Z_i has been made (i.e., $w_i \neq *$), then $Z_i \in \{0,1\}$ – namely, X_i is a legal input value for g. We will be making this convenient assumption (sometimes implicitly) throughout our lower-bound analysis, as our target distributions are always supported on legal input values only.

The second generic type is a *Z-mixer*, described by $(i, \alpha, \beta) \in [n] \times [0,1] \times [0,1]$ and acting as follows:



In words, this generic node corresponds to the following action:

- If $w_i = *$, then the outgoing edge marked with "1" is selected with probability β and the edge marked with "0" is selected otherwise – i.e., with probability $1-\beta$.
- If $w_i \neq *$, then the outgoing edge marked with "1" is selected with probability α and the edge marked with "0" is selected otherwise – i.e., with probability $1-\alpha$.

2.1 Some properties of our trees

When constructing a protocol for computing f, we will try to keep low the (expected) number of actual Z-queries made by the protocol. Note that the only node type where Z_i may be

queried are the Z-nodes. There a query to Z_i can only take place if $w_i = *$, and when that happens the value of w_i is updated – therefore, *each Z_i can be queried at most once.*

The other property of interest to us is also related to "saving" Z-queries. We call it *polarity*, it says that (for all $i \in [n]$) the set of computational paths[3] leading to the same tree vertex can either consist of *paths that have not queried Z_i and those where $Z_i = 0$* or of *paths that have not queried Z_i and of those where $Z_i = 1$* – in other words, *a path "knowing" that $Z_i = 0$ and a path "knowing" that $Z_i = 1$ cannot lead to the same vertex* in our tree. That is why we are calling our trees *polarised*.

To see that they are indeed polarised, note that only the two generic node types have merging paths; Z-mixers never make queries, and therefore cannot affect the polarity; a Z-node can only merge paths of the "same polarity" (if $w_i = 1$, then the corresponding paths can only mix with the 1-outcome of the Z-query, and vice versa), thus preserving the polarity.

Informally, the reason why polarity will be important for us is this. Suppose that a tree were not polarised and let l_0 be a leaf, such that exactly half of the paths leading to it "know" that $Z_i = 0$, while the other half "know" that $Z_i = 1$. On the one hand, the "leaf-wise" knowledge in l_0 about the value of Z_i is the lowest possible (the entropy of the bit Z_i, conditioned on reaching l_0 is 1); on the other, the probability that Z_i has actually been queried by the protocol when l_0 is reached is 1 – so, conditional on reaching l_0, a query to Z_i has been fully "wasted" (as a leaf, l_0 must correspond to an answer to $f(Z_1, \ldots, Z_n)$, and this answer must be fully independent of Z_i).

By having chosen carefully the set of allowed generic actions in a tree, we are guaranteeing that it is "ontologically polarised" – thus avoiding the possibility of "wasted queries", as described above. On the other hand, having such "reduced instruction set" will demand from us a somewhat bigger effort in order to "mimic" the behaviour of the given protocol for $f \circ g^n$ when constructing a protocol for f. We will see next how to achieve that; as a result, the constructed protocol will compute $f(Z)$ with exactly the same accuracy that the original protocol achieves for $f \circ g^n(X_1, \ldots, X_n)$.

3 Constructing a protocol for f, given a protocol for $f \circ g^n$

Recall that the argument of our lower bound is based on transforming a given protocol for $f \circ g^n$ into a protocol for f, as accurate as the original one and whose query complexity will be low if that of the original protocol was low. In this part we describe the transformation and notice some basic properties of the constructed protocol for f, as summarised by Lemma 1 and Corollary 1 (used in Section 4 to obtain the desired bound).

Let μ_g over $\{0,1\}^m$ be a non-trivial[4] input distribution for g that is supported on legal input values only, and for $a \in \{0,1\}$ denote by μ_g^a the distribution of $Y \in \{0,1\}^m$ when $Y \sim \mu_g$, conditioned on $g(Y) = a$. For a binary string z of length k, denote by $z \circ \mu_g$ the distribution of $(X_1, \ldots, X_k)$, where $X_i \sim \mu_g^{z_i}$ for every $i \in [k]$. For a distribution ν on $\{0,1\}^k$, denote by

$$\nu \circ \mu_g$$

[3] When a tree contains generic nodes, by a *computational path* we mean not only the set of tree nodes that have been "visited", but also the information about the randomised decisions taken at each visited generic node. Accordingly, a number of distinct computational paths can lead to the same tree leaf.

[4] so that g is not constant on $\mathrm{supp}(\mu_g)$

the distribution of $X \in \{0,1\}^{k \cdot m}$, corresponding to choosing $Z \sim \nu$ followed by $X \sim Z \circ \mu_g$.

Let μ over $\{0,1\}^{n \cdot m}$ be input distribution for $f \circ g^n$, so that $\mu = \mu_f \circ \mu_g$ for some μ_f over $\{0,1\}^n$ – in other words, μ_f is the distribution of $(g(X_1), \ldots, g(X_n))$ when $X = (X_1, \ldots, X_n) \sim \mu$.

Let $\mathcal{P}$ be a (deterministic) protocol that solves $f \circ g^n$ with respect to μ with error ε. Next we construct a protocol $\mathcal{P}'$ that solves f with respect to μ_f with the same error ε. We will represent $\mathcal{P}'$ as a polarised protocol tree, which will be isomorphic to the tree of $\mathcal{P}$.

3.1 Constructing $\mathcal{P}'$: the inductive step

Let Z be a random variable taking values in $\{0,1\}^n$: in the context of computing $f \circ g^n(X_1, \ldots, X_n)$ we let $Z_i = g(X_i)$, and in the context of computing f we let Z_i be the i'th bit of input (this convention should not confuse us, as Z will denote the input to f in both cases).

To build a protocol for f, we repeatedly apply the following "local" mechanism that "translates" every node of the (given) tree for $f \circ g^n$ into a node of a polarised tree for f.

Starting from the root and selecting at each step a new non-leaf vertex whose predecessor in $\mathcal{P}'$ has already been constructed, we proceed as follows. Denote by v_0 be the current node in $\mathcal{P}$ and by v_0' the corresponding node in $\mathcal{P}'$. The action of v_0 is an X-query – let it query X_{i_0,j_0}. Let

$$p_{in} \stackrel{\text{def}}{=} \mathop{\mathbf{Pr}}_{X \sim \mu}\left[Z_{i_0} = 1 \middle| v_0\right],$$

where $[v_0]$ denotes the event that execution of $\mathcal{P}$ has reached the node v_0. Also let

$$p_< \stackrel{\text{def}}{=} \mathop{\mathbf{Pr}}_{\mu}\left[Z_{i_0} = 1 \middle| [v_0], X_{i_0,j_0} = a_0\right] \quad \text{and} \quad p_> \stackrel{\text{def}}{=} \mathop{\mathbf{Pr}}_{\mu}\left[Z_{i_0} = 1 \middle| [v_0], X_{i_0,j_0} = 1 - a_0\right],$$

where $a_0 \in \{0,1\}$ is such that $p_< \leq p_>$, and

$$\tau_< \stackrel{\text{def}}{=} \mathop{\mathbf{Pr}}_{\mu}\left[X_{i_0,j_0} = a_0 \middle| v_0\right] \quad \text{and} \quad \tau_> \stackrel{\text{def}}{=} \mathop{\mathbf{Pr}}_{\mu}\left[X_{i_0,j_0} = 1 - a_0 \middle| v_0\right] \quad \{= 1 - \tau_<\}.$$

The action of v_0' will be either a Z-node or a Z-mixer. We will associate the edge that leaves v_0' and is marked by "0" with the "a_0" edge of v_0, and the edge that leaves v_0' and is marked by "1" with the "$1 - a_0$" edge of v_0.

Informally, we want the action of v_0' to "mimic" that of v_0 with respect to the (conditional) distribution of Z: if we do that for every node in $\mathcal{P}$, then in the end we will obtain a protocol that solves $f(Z)$ with respect to μ_f with the same accuracy as $\mathcal{P}$ achieves for $f \circ g^n(X_1, \ldots, X_n)$ with respect to μ. In order to imitate the behaviour of $\mathcal{P}$ with respect to Z, the new protocol must (at least) imitate it at every v_0 with respect to the corresponding Z_{i_0}.

Technically, our assignment of action to v_0' will be such that if

$$\mathop{\mathbf{Pr}}_{\mu_f}\left[Z_{i_0} = 1 \middle| v_0'\right] = p_{in}, \tag{1}$$

then

$$\mathop{\mathbf{Pr}}_{\mu_f}\left[Z_{i_0} = 1 \middle| v_0', \text{answer "0"}\right] = p_< \quad \text{and} \quad \mathop{\mathbf{Pr}}_{\mu_f}\left[Z_{i_0} = 1 \middle| v_0', \text{answer "1"}\right] = p_> \tag{2}$$

and

$$\underset{\mu_f}{\mathbf{Pr}}\left[\text{answer ``0''}\middle|v'_0\right] = \tau_< \quad \text{and} \quad \underset{\mu_f}{\mathbf{Pr}}\left[\text{answer ``1''}\middle|v'_0\right] = \tau_>, \tag{3}$$

where [answer "b"] is the event that the generic action assigned to v'_0 returns the corresponding answer.

Note that (1) and (2) implies

$$\begin{aligned}\tau_< \cdot p_< + (1-\tau_<)\cdot p_> &= p_{in} \\ &= \underset{\mu_f}{\mathbf{Pr}}\left[\text{answer ``0''}\middle|v'_0\right]\cdot p_< + \left(1 - \underset{\mu_f}{\mathbf{Pr}}\left[\text{answer ``0''}\middle|v'_0\right]\right)\cdot p_>,\end{aligned}$$

and therefore (3) "almost always": the only exception is the "degenerate" case $[p_< = p_>]$ (which we will handle soon). In all other cases it will be enough to guarantee (2), assuming (1), and that will imply (3) as well.

Recall that all the predecessors of v'_0 in $\mathcal{P}'$ have already been constructed, so the value

$$q_{in} \stackrel{\text{def}}{=} \underset{\mu_f}{\mathbf{Pr}}\left[w_{i_0} \neq *\middle|v'_0\right]$$

is well-defined. Observe that

$$p_< \leq p_{in} \leq p_> \leq 1,$$

as $\tau_< \cdot p_< + \tau_> \cdot p_> = p_{in}$ and $\tau_< + \tau_> = 1$.

For the same reason, if $p_< = p_{in}$ or $p_{in} = p_>$, then $p_< = p_{in} = p_>$, which is the degenerate case mentioned above. To handle it, we let the action of v'_0 be a Z-mixer $(i_0, \tau_>, \tau_>)$ – it is easy to see that this choice satisfies both (2) and (3).

Now assume that

$$p_< < p_{in} < p_>. \tag{4}$$

As our tree is polarised, either $\mathbf{Pr}\left[w_{i_0} = 0\middle|v'_0\right] = q_{in}$ or $\mathbf{Pr}\left[w_{i_0} = 1\middle|v'_0\right] = q_{in}$ holds – without loss of generality, let us assume the latter (the other case is symmetric and treated similarly).

Let

$$p^*_{in} \stackrel{\text{def}}{=} \mathbf{Pr}\left[Z_{i_0} = 1\middle|[v'_0], w_{i_0} = *\right]$$

and observe that

$$p^*_{in}\cdot(1-q_{in}) + q_{in} \;=\; p_{in} \;=\; p_<\cdot\tau_< + p_>\cdot\tau_> \;=\; p_<\cdot\tau_< + p_>\cdot(1-\tau_<) \tag{5}$$

– these equalities will play their crucial role soon.

Note that the first equality in (5) implies

$$p^*_{in} = \frac{p_{in} - q_{in}}{1 - q_{in}} \leq p_{in}, \tag{6}$$

and so – by assumption (4) – either $p^*_{in} \leq p_< < p_>$ or $p_< < p^*_{in} < p_>$ holds. Both possibilities are valid and we will handle them differently: either with a Z-mixer or with a Z-node.

3.1.1 The case of $p^*_{in} \le p_<$: using a Z-mixer

Here we are assuming that

$$p^*_{in} \le p_< < p_> \le 1. \tag{7}$$

We will choose such $\alpha_0, \beta_0 \in [0,1]$ that making a Z-mixer (i_0, α_0, β_0) to be the action of v'_0 will satisfy (2).

Let γ_1 be such that

$$p_1(\gamma_1) \stackrel{\text{def}}{=} \frac{p^*_{in} \cdot (1 - q_{in}) + \gamma_1}{1 - q_{in} + \gamma_1} = p_<, \tag{8}$$

which exists and satisfies

$$\gamma_1 \in [0, q_{in}], \tag{9}$$

as $p_1(0) = p^*_{in} \le p_<$ by (7), $p_1(q_{in}) = p_{in} > p_<$ by (5) and (4), and $p_1(\cdot)$ is monotone on $[0, q_{in}]$, obviously. Then

$$\begin{aligned} p_< \cdot (1 - q_{in} + \gamma_1) + q_{in} - \gamma_1 = p^*_{in} \cdot (1 - q_{in}) + q_{in} = p_{in} \\ = p_< \cdot \tau_< + p_> \cdot (1 - \tau_<), \end{aligned}$$

where the first equality is (8) and the last two are (5). Since $p_> \le 1$ and $\gamma_1 \le q_{in}$,

$$p_< \cdot (1 - q_{in} + \gamma_1) + p_> \cdot (q_{in} - \gamma_1) \le p_< \cdot \tau_< + p_> \cdot (1 - \tau_<)$$

and

$$p_< \cdot (1 - q_{in} + \gamma_1 - \tau_<) \le p_> \cdot (1 - q_{in} + \gamma_1 - \tau_<)$$

– that is,

$$1 - q_{in} + \gamma_1 \ge \tau_<. \tag{10}$$

Let $\alpha_0, \beta_0 \in [0,1]$ be such that

$$1 - \beta_0 = \frac{\tau_<}{1 - q_{in} + \gamma_1}$$

and

$$(1 - \alpha_0) \cdot q_{in} = (1 - \beta_0) \cdot \gamma_1,$$

their existence follows from (10) and (9), respectively. We set the action of v'_0 to be a Z-mixer parametrised by (i_0, α_0, β_0) and we claim that (2) is satisfied.

On the one hand,

$$\begin{aligned} \mathop{\mathbf{Pr}}_{\mu_f}\left[Z_{i_0} = 1 \middle| v'_0, \text{answer “0”}\right] &= \frac{(1 - q_{in}) \cdot (1 - \beta_0) \cdot p^*_{in} + q_{in} \cdot (1 - \alpha_0) \cdot 1}{(1 - q_{in}) \cdot (1 - \beta_0) + q_{in} \cdot (1 - \alpha_0)} \\ &= \frac{(1 - q_{in}) \cdot (1 - \beta_0) \cdot p^*_{in} + (1 - \beta_0) \cdot \gamma_1}{(1 - q_{in}) \cdot (1 - \beta_0) + (1 - \beta_0) \cdot \gamma_1} \end{aligned}$$

$$= \frac{(1-q_{in})\cdot p^*_{in} + \gamma_1}{(1-q_{in})+\gamma_1} \;=\; p_<,$$

where the last equality is (8). On the other,

$$\begin{aligned}\mathop{\mathbf{Pr}}_{\mu_f}\left[\text{answer ``0''}\middle|v'_0\right] &= (1-q_{in})\cdot(1-\beta_0) + q_{in}\cdot(1-\alpha_0)\\ &= (1-q_{in}+\gamma_1)\cdot(1-\beta_0) \;=\; \tau_<.\end{aligned}$$

Therefore,

$$\mathop{\mathbf{Pr}}_{\mu_f}\left[\text{answer ``1''}\middle|v'_0\right] = 1 - \mathop{\mathbf{Pr}}_{\mu_f}\left[\text{answer ``0''}\middle|v'_0\right] = 1-\tau_< = \tau_>$$

and

$$\begin{aligned}\mathop{\mathbf{Pr}}_{\mu_f}\left[Z_{i_0}=1\middle|v'_0,\text{answer ``1''}\right] &= \frac{p_{in} - \mathbf{Pr}\left[Z_{i_0}=1\middle|v'_0,\text{answer ``0''}\right]\cdot\mathbf{Pr}\left[\text{answer ``0''}\middle|v'_0\right]}{\mathbf{Pr}\left[\text{answer ``1''}\middle|v'_0\right]}\\ &= \frac{p_{in}-p_<\cdot\tau_<}{\tau_>} = p_>,\end{aligned}$$

as required.

3.1.2 The case of $p_< < p^*_{in}$: using a Z-node

Now assume that

$$p_< < p^*_{in} < p_> \le 1.$$

We will choose such $\alpha_0,\beta_0\in[0,1]$ that making a Z-node (i_0,α_0,β_0) to be the action of v'_0 will satisfy (2).

Let $\alpha'\in[0,1]$ be such that

$$\frac{(1-\alpha')\cdot p^*_{in}}{1-p^*_{in}} = \frac{p_<}{1-p_<}, \tag{11}$$

and denote

$$\gamma_2 \stackrel{\text{def}}{=} (1-\alpha' p^*_{in})(1-q_{in}) \quad\text{and}\quad \gamma_3 \stackrel{\text{def}}{=} \frac{\tau_<}{\gamma_2}.$$

From (11):

$$p_< = \frac{(1-\alpha')\cdot p^*_{in}}{1-\alpha' p^*_{in}}, \tag{12}$$

and so,

$$\begin{aligned}p_<\cdot\gamma_2 + 1 - \gamma_2 &= \frac{(1-\alpha')\cdot p^*_{in}}{1-\alpha' p^*_{in}}\cdot\gamma_2 + 1 - \gamma_2\\ &= (1-\alpha')\cdot p^*_{in}\cdot(1-q_{in}) + 1 - (1-\alpha' p^*_{in})(1-q_{in})\\ &= p^*_{in}\cdot(1-q_{in}) + q_{in}\end{aligned}$$

$$
\begin{aligned}
&= p_{in} \\
&= p_< \cdot \tau_< + p_> \cdot (1 - \tau_<),
\end{aligned}
$$

where the last two equalities follow, respectively, from (6) and (5). As $p_>, \tau_< \in [0,1]$,

$$p_< \cdot \gamma_2 + 1 - \gamma_2 \leq p_< \cdot \tau_< + 1 - \tau_<,$$

and it holds that $\gamma_2 \geq \tau_<$ and

$$\gamma_3 \in [0,1]. \tag{13}$$

Let

$$\alpha_0 = \gamma_3 \cdot \alpha' \quad \text{and} \quad \beta_0 = \frac{1 - \gamma_3}{1 - \gamma_3 \cdot \alpha'}.$$

It follows from $\alpha' \in [0,1]$ and (13) that $\alpha_0, \beta_0 \in [0,1]$. We set the action of v'_0 to be a Z-node parametrised by (i_0, α_0, β_0) and claim that (2) is satisfied.

On the one hand,

$$
\begin{aligned}
\mathop{\mathbf{Pr}}_{\mu_f}\left[\text{answer ``0''} \middle| v'_0\right] &= (1 - q_{in}) \cdot \big((1 - \alpha_0) \cdot (1 - \beta_0) + \alpha_0 \cdot (1 - p^*_{in})\big) \\
&= (1 - q_{in}) \cdot \big(1 - \alpha_0 \cdot p^*_{in} - \beta_0 \cdot (1 - \alpha_0)\big) \\
&= (1 - q_{in}) \cdot (\gamma_3 - \gamma_3 \cdot \alpha' \cdot p^*_{in}) \\
&= \tau_< \cdot \frac{(1 - q_{in}) \cdot (1 - \alpha' \cdot p^*_{in})}{\gamma_2} \;=\; \tau_<;
\end{aligned}
$$

on the other,

$$
\begin{aligned}
\mathop{\mathbf{Pr}}_{\mu_f}\left[Z_{i_0} = 1 \middle| v'_0, \text{answer ``0''}\right] &= \frac{(1 - \alpha_0) \cdot (1 - \beta_0) \cdot p^*_{in}}{(1 - \alpha_0) \cdot (1 - \beta_0) + \alpha_0 \cdot (1 - p^*_{in})} \\
&= \frac{(\gamma_3 - \alpha_0) \cdot p^*_{in}}{\gamma_3 - \alpha_0 \cdot p^*_{in}} \\
&= \frac{(1 - \alpha') \cdot p^*_{in}}{1 - \alpha' \cdot p^*_{in}} \;=\; p_<,
\end{aligned}
$$

where the last equality is (12). As in the case of $p^*_{in} \leq p_<$, from here it follows that (2) holds.

3.2 Constructing $\mathcal{P}'$: summing up

Thus far, we have assigned actions to the internal nodes in a polarised tree representing $\mathcal{P}'$, such that for every pair of mutually-corresponding non-leaves $v_0 \in \mathcal{P}$ and $v'_0 \in \mathcal{P}'$ that satisfy (1), both (2) and (3) must hold. Intuitively, it says that if the distribution of Z_{i_0} was "correct" upon reaching v'_0, then it is stays "correct" also after the operation performed by v'_0, where "correct" means being the same as in $\mathcal{P}$.

We would like to use this fact inductively in order to conclude that for every mutually-corresponding pair of vertices $v'_0 \in \mathcal{P}'$ and $v_0 \in \mathcal{P}$, the distribution of $Z \sim \mu_f$ conditioned on reaching v'_0 is the same as the distribution of $Z = g^n(X)$ for $X \sim \mu$ conditioned on reaching v_0. For that we need a somewhat stronger "step statement" than what we have above: namely,

we would like to say that if the distribution of the whole random vector Z is "correct" upon reaching v'_0, then it stays "correct" after leaving v'_0 as well.

In other words, we want to argue that it is enough to imitate by the action of v'_0 the behaviour of v_0 with respect to the "queried" coordinate Z_{i_0} (as we have done) in order to conclude that the behaviour with respect to full Z has been mimicked as well.

First we claim that the distribution of X, conditioned on reaching a certain node in $\mathcal{P}$, is always "reasonably localised" when $X \sim \mu$ (which holds due to the fact that μ has the "concatenated" structure of $\mu_f \circ \mu_g$).

Claim 1. *Let v be a vertex in $\mathcal{P}$ and $i \in [n]$, then*

$$\underset{X\sim\mu}{\mathbf{I}}\left[X_i : X_{[n]\setminus\{i\}}\middle|[v], Z_i\right] = 0.$$ [5]

Note that in general $\mathbf{I}\left[X_i : X_{[n]\setminus\{i\}}\middle|[v]\right]$ can be positive, as μ_f doesn't need to be a product distribution; however, in the above statement this possible dependence is "shielded" by conditioning on Z_i.

Proof. Let $a \in \{0,1\}$ and recall that $\mu = \mu_f \circ \mu_g$. By the nature of a query protocol, there exist sets A and B such that the distribution of $X \sim \mu$, conditioned on reaching v, is the same as the distribution of $X \sim \mu$, conditioned on $X_i \in A$ and $X_{[n]\setminus\{i\}} \in B$. Accordingly,

$$\begin{aligned}\underset{X\sim\mu}{\mathbf{I}}\left[X_i : X_{[n]\setminus\{i\}}\middle|[v], Z_i = a\right] &= \underset{\mu}{\mathbf{I}}\left[X_i : X_{[n]\setminus\{i\}}\middle|Z_i = a, X_i \in A, X_{[n]\setminus\{i\}} \in B\right] \\ &= \underset{\mu'}{\mathbf{I}}\left[X_i : X_{[n]\setminus\{i\}}\middle|X_i \in A, X_{[n]\setminus\{i\}} \in B\right],\end{aligned}$$

where $\mu' = \mu'_f \circ \mu_g$ for $Z \sim \mu'_f$ defined as $Z \sim \mu_f$, subject to $[Z_i = a]$. By our definition of concatenated distributions, $\mathbf{I}_{\mu'}\left[X_i : X_{[n]\setminus\{i\}}\right] = 0$, and therefore

$$\underset{\mu'}{\mathbf{I}}\left[X_i : X_{[n]\setminus\{i\}}\middle|X_i \in A, X_{[n]\setminus\{i\}} \in B\right] = 0$$

as well.

Finally, $\mathbf{I}_\mu\left[X_i : X_{[n]\setminus\{i\}}\middle|[v], Z_i\right]$ is a convex combination of $\mathbf{I}_\mu\left[X_i : X_{[n]\setminus\{i\}}\middle|[v], Z_i = 0\right]$ and $\mathbf{I}_\mu\left[X_i : X_{[n]\setminus\{i\}}\middle|[v], Z_i = 1\right]$, as μ_g is supported on legal input values for g. The result follows. ■*Claim 1*

Let v_0 be a node in $\mathcal{P}$ that queries X_{i_0,j_0} and let v'_0 be the corresponding node in $\mathcal{P}'$. Assume

$$\forall z \in \{0,1\}^n : \underset{Z\sim\mu_f}{\mathbf{Pr}}\left[Z = z\middle|v'_0\right] = \underset{X\sim\mu}{\mathbf{Pr}}\left[Z = z\middle|v_0\right] \tag{14}$$

– in particular, this means that (1) is satisfied, and therefore (2) and (3) hold with respect to v_0 and v'_0.

[5] Here "$[v]$" denotes the *event* that v is reached by the protocol and "Z_i" stands for conditioning on the *value* that the variable takes.

Let us see what happens at the vertices $v_0 \in \mathcal{P}$ and $v'_0 \in \mathcal{P}'$, conditioned upon $[Z_{i_0} = 1]$. First of all, from (2) and (3) it follows that

$$\mathop{\mathbf{Pr}}_{\mu_f}\left[\text{answer ``0''}\middle|[v'_0], Z_{i_0} = 1\right] = \mathop{\mathbf{Pr}}_{\mu}\left[X_{i_0,j_0} = a_0\middle|[v_0], Z_{i_0} = 1\right].$$

Conditioned on $[v_0] \wedge [Z_{i_0} = 1] \wedge [X_{i_0,j_0} = a_0]$, the distribution of Z is the same as conditioned only on $[v_0] \wedge [Z_{i_0} = 1]$, as follows from Claim 1 and the fact that $Z_{[n]\setminus\{i_0\}}$ is a function of $X_{[n]\setminus\{i_0\}}$. In other words,

$$\forall z \in \{0,1\}^n : \mathop{\mathbf{Pr}}_{\mu}\left[Z = z\middle|[v_0], Z_{i_0} = 1, X_{i_0,j_0} = a_0\right] = \mathop{\mathbf{Pr}}_{\mu}\left[Z = z\middle|[v_0], Z_{i_0} = 1\right]. \tag{15}$$

In the case of $\mathcal{P}'$ the following trivial analogue of Claim 1 holds:

$$\mathop{\mathbf{I}}_{Z\sim\mu_f}\left[\text{the answer of } v'_0 : Z_{[n]\setminus\{i_0\}}\middle|[v'_0], Z_{i_0}\right] = 0;$$

accordingly,

$$\forall z \in \{0,1\}^n : \mathop{\mathbf{Pr}}_{\mu_f}\left[Z = z\middle|[v'_0], Z_{i_0} = 1, \text{answer ``0''}\right] = \mathop{\mathbf{Pr}}_{\mu_f}\left[Z = z\middle|[v'_0], Z_{i_0} = 1\right].$$

By (14) and (15), this means that $\forall z \in \{0,1\}^n$:

$$\mathop{\mathbf{Pr}}_{\mu}\left[Z = z\middle|[v_0], Z_{i_0} = 1, X_{i_0,j_0} = a_0\right] = \mathop{\mathbf{Pr}}_{\mu_f}\left[Z = z\middle|[v'_0], Z_{i_0} = 1, \text{answer ``0''}\right].$$

By (2) and the symmetry with respect to the value of Z_{i_0} and the answer,

$$\begin{aligned}\forall z \in \{0,1\}^n : & \\ &\mathop{\mathbf{Pr}}_{\mu}\left[Z = z\middle|[v_0], X_{i_0,j_0} = a_0\right] = \mathop{\mathbf{Pr}}_{\mu_f}\left[Z = z\middle|[v'_0], \text{answer ``0''}\right] \\ \text{and} & \\ &\mathop{\mathbf{Pr}}_{\mu}\left[Z = z\middle|[v_0], X_{i_0,j_0} = 1 - a_0\right] = \mathop{\mathbf{Pr}}_{\mu_f}\left[Z = z\middle|[v'_0], \text{answer ``1''}\right].\end{aligned} \tag{16}$$

So, for every pair of mutually-corresponding vertices $v_0 \in \mathcal{P}$ and $v'_0 \in \mathcal{P}'$ that satisfy (14), both (16) and (3) must hold. This is precisely the statement that we want to use for our inductive argument, as described in the beginning of this part.

Formally, the full argument goes like that: Let $v_{root} \in \mathcal{P}$ and $v'_{root} \in \mathcal{P}'$ be the roots and assume that the initial distribution of Z is the same – that is, (14) is satisfied at the roots. Let $v_1 \in \mathcal{P}$ and $v'_1 \in \mathcal{P}'$ be mutually-corresponding sons of the roots; by the above statement, both (16) and (3) hold at v_{root} and v'_{root}. Note that (16) with respect to the roots implies that (14) is satisfied with respect to v_1 and v'_1. Continuing inductively, we conclude that both (16) and (3) hold (unconditionally) for all mutually-corresponding non-leaves of $\mathcal{P}$ and $\mathcal{P}'$.

Thus we have shown the following:

Lemma 1. *Let $g : \{0,1\}^m \to \{0,1,*\}$. Let $\mathcal{P}$ be a deterministic protocol that queries bits of $X = (X_1, \ldots, X_n) \in \{0,1\}^{n\cdot m}$. Let μ_g be a distribution over $\{0,1\}^m$, supported on legal input values for g, and $\mu = \mu_f \circ \mu_g$ be a distribution over $\{0,1\}^{n\cdot m}$, so that μ_f is the distribution of $(g(X_1, \ldots, g(X_n))$ when $X \sim \mu$.*

Then there exists a protocol $\mathcal{P}'$ that queries bits of $Z \in \{0,1\}^n$, such that an isomorphism $\mathcal{M}$ maps the protocol tree of $\mathcal{P}$ to a polarised tree representing $\mathcal{P}'$. Moreover, it holds that

- *for every vertex v in the tree of $\mathcal{P}$, the probability of reaching it under $X \sim \mu$ is the same as the probability of reaching $\mathcal{M}(v)$ in $\mathcal{P}'$ under $Z \sim \mu_f$;*
- *for every vertex v in the tree of $\mathcal{P}$, the distribution of $(g(X_1), \ldots, g(X_n)$ taken with respect to $X \sim \mu$ and conditioned upon reaching v is the same as the distribution of $(Z_1, \ldots, Z_n)$ taken with respect to $Z \sim \mu_f$ and conditioned upon reaching $\mathcal{M}(v)$ in $\mathcal{P}'$.*

The protocol $\mathcal{P}'$ is, in the first place, a protocol for computing $f(Z)$. From the above lemma it follows that for every leaf l of $\mathcal{P}$

$$\mathop{\mathbf{Pr}}_{Z\sim\mu_f}\left[f(Z)=1\middle|\mathcal{M}(l)\right] = \mathop{\mathbf{Pr}}_{X\sim\mu}\left[f\circ g^n(X)=1\middle|l\right].$$

Therefore, if we label every $\mathcal{M}(l)$ by the same answer as appears on l, we get a protocol that computes f over μ_f as accurately as $\mathcal{P}$ computes $f \circ g^n$ over μ.

Now let us assume for a moment that μ_f is supported on the whole $\{0,1\}^n$ and revisit Lemma 1. According to its statement, if $X \sim \mu$ and $Z \sim \mu_f$, then for every leaf l of $\mathcal{P}$ the distribution of $(g(X_1), \ldots, g(X_n)$ conditioned upon reaching l is the same as the distribution of Z conditioned upon reaching $\mathcal{M}(l)$ in $\mathcal{P}'$; moreover, the probabilities of reaching l and of reaching $\mathcal{M}(l)$ are the same. Therefore, for every $z_0 \in \{0,1\}^n$:

$$\begin{aligned}\mathop{\mathbf{Pr}}_{X\sim z_0\circ\mu_g}[l] &= \mathop{\mathbf{Pr}}_{\mu_f\circ\mu_g}\left[l\middle|Z=z_0\right] \\ &= \mathop{\mathbf{Pr}}_{\mu_f\circ\mu_g}\left[Z=z_0\middle|l\right]\cdot\frac{\mathbf{Pr}_{\mu_f\circ\mu_g}[l]}{\mathbf{Pr}_{\mu_f\circ\mu_g}[Z=z_0]} \\ &= \mathop{\mathbf{Pr}}_{Z\sim\mu_f}\left[Z=z_0\middle|\mathcal{M}(l)\right]\cdot\frac{\mathbf{Pr}_{\mu_f}[\mathcal{M}(l)]}{\mathbf{Pr}_{\mu_f}[Z=z_0]} = \mathbf{Pr}\left[\mathcal{M}(l)\middle|Z=z_0\right]\end{aligned}$$

(note that the rightmost probability only depends on the "internal randomness" of $\mathcal{P}'$, and not on the distribution of Z).

In other words, the distribution of the leaf that $\mathcal{P}'(Z)$ reaches when $Z = z_0$ is the same as the distribution of $\mathcal{M}(L)$, where L is the leaf reached by $\mathcal{P}(X)$ when $X \sim z_0 \circ \mu_g$.[6] Accordingly, the protocols $\mathcal{P}$ and $\mathcal{P}'$ "perform identically" (in the sense of Lemma 1) even if conditioned upon the value of Z, and therefore also with respect to *any* distribution $Z \sim \nu$ (corresponding to $X \sim \nu \circ \mu_g$). In other words, the construction of $\mathcal{P}'$ can be done independently of μ_f: it is enough to know the protocol $\mathcal{P}$, the function g and the distribution μ_g.

To conclude:

Corollary 1. *Let $f \subseteq \{0,1\}^n \times \Xi$ and $g : \{0,1\}^m \to \{0,1,*\}$. Let $\mathcal{P}$ be a deterministic protocol that queries bits of $X = (X_1, \ldots, X_n) \in \{0,1\}^{n\cdot m}$. Let μ_g be a distribution over $\{0,1\}^m$, supported on legal input values for g.*

Then there exists a protocol $\mathcal{P}'$ that queries bits of $Z \in \{0,1\}^n$, such that an isomorphism $\mathcal{M}$ maps the protocol tree of $\mathcal{P}$ to a polarised tree representing $\mathcal{P}'$. Moreover, for every distribution ν over $\{0,1\}^n$ it holds that

- *the error of $\mathcal{P}'$ in computing $f(Z)$ when $Z \sim \nu$ is the same as that of $\mathcal{P}$ in computing $f \circ g^n(X)$ when $X \sim \nu \circ \mu_g$;*

[6] Recall that $\mathcal{P}'$ is randomised, and so the leaf reached by the protocol is not necessarily determined by the input value. Note also that $z_0 \circ \mu_g$ is the distribution of $X \sim \mu_f \circ \mu_g$, conditioned upon $Z = (g(X_1), \ldots, g(X_n)) = z_0$.

- *for every z_0 in the support of ν, the distribution of $\mathcal{M}(L)$, where L is the leaf reached by $\mathcal{P}$ conditioned upon $(g(X_1), \ldots, g(X_n)) = z_0$ is the same as the distribution of the leaf reached by $\mathcal{P}'$ conditioned upon $Z = z_0$;*
- *for every vertex v in the tree of $\mathcal{P}$, the probability of reaching it under $X \sim \nu \circ \mu_g$ is the same as the probability of reaching $\mathcal{M}(v)$ in $\mathcal{P}'$ under $Z \sim \nu$;*
- *for every vertex v in the tree of $\mathcal{P}$, the distribution of $(g(X_1), \ldots, g(X_n))$ taken with respect to $X \sim \nu \circ \mu_g$ and conditioned upon reaching v is the same as the distribution of Z taken with respect to $Z \sim \nu$ and conditioned upon reaching $\mathcal{M}(v)$ by $\mathcal{P}'$.*

4 Comparing the complexities of the protocols

In Section 3 we were given a query protocol $\mathcal{P}$ for computing $f \circ g^n$ and used it to construct a protocol $\mathcal{P}'$ for computing f. Now let us analyse the properties of $\mathcal{P}'$ (as summarised in Corollary 1) in order to argue that it has low query complexity if that of $\mathcal{P}$ was low.

Let μ_f be such input distribution for f that $\mathcal{R}_{\mu_f,\varepsilon}(f) \in \Omega(\mathcal{R}(f))$ for some fixed $\varepsilon > 0$. We will assume that μ_f is *non-fixing* in the following sense: for any $i_0 \in [n]$ and $s_0 \in \{0,1\}^{n-1}$, the distribution of Z_{i_0} when $Z \sim \mu_f$ and $Z|_{[n]\setminus\{i_0\}} = s_0$ has positive entropy.[7] Let μ_g be input distribution for g, supported on legal input values (to be chosen in Section 4.1.3; it will be hard in a certain "error-independent" sense).

We apply Corollary 1 with respect to this μ_g and the given protocol $\mathcal{P}$, letting $\mathcal{P}'$ and $\mathcal{M}$ be as guaranteed by the statement.

4.1 Protocols as Z_i-predictors

Informally, in our analysis we will look at the "knowledge" of a given leaf of a protocol tree about Z_{i_0}. To formalise this, we consider the behaviour of a protocol with respect to the uniform distribution of Z, which corresponds to $X \sim \mu_g^{\mathcal{U}} \stackrel{\text{def}}{=} \mathcal{U}_{\{0,1\}^n} \circ \mu_g$ in the case of $\mathcal{P}$.

Let $\mathcal{T}$ denote the protocol tree of $\mathcal{P}$. For every leaf $l \in \mathcal{T}$, let $\lambda_{i_0}(l) \in \{0,1\}$ be a most likely value of $Z_{i_0} = g(X_{i_0})$ when $X \sim \mu_g^{\mathcal{U}}$, conditioned on reaching l.[8] Let

$$\delta_{i_0}(l) \stackrel{\text{def}}{=} \underset{X \sim \mu_g^{\mathcal{U}}}{\mathbf{Pr}} \left[g(X_{i_0}) = \lambda_{i_0}(l) \middle| l \right] - \frac{1}{2} \quad \left\{ \in \left[0, \frac{1}{2}\right] \right\}$$

and

$$\delta_{i_0}(\mathcal{T}) \stackrel{\text{def}}{=} \underset{L}{\mathbf{E}} \left[\delta_{i_0}(L) \right],$$

where L is distributed as the leaf of $\mathcal{T}$ reached by $\mathcal{P}$ when $X \sim \mu_f \circ \mu_g$ (note the "mixture of distributions": $\lambda_{i_0}(l)$ and $\delta_{i_0}(l)$ are defined relative to $X \sim \mu_g^{\mathcal{U}}$, but L in the definition of $\delta_{i_0}(\mathcal{T})$ is sampled with respect to $X \sim \mu_f \circ \mu_g$).

[7] If μ_f is fixing, we let $\varepsilon' \stackrel{\text{def}}{=} \varepsilon/2$, $\mu_f' \stackrel{\text{def}}{=} \frac{\mu_f + \varepsilon' \cdot \mathcal{U}_{\{0,1\}^n}}{1+\varepsilon'}$ and take (μ_f', ε') instead of (μ_f, ε): the resulting μ_f' is non-fixing and $\mathcal{R}_{\mu_f',\varepsilon'}(f) \geq \mathcal{R}_{\mu_f,\varepsilon}(f) \in \Omega(\mathcal{R}(f))$.

[8] Recall that μ_g is supported only on legal input values for g, so $\mathbf{Pr}\left[g(X_{i_0}) = 0 \middle| l\right] + \mathbf{Pr}\left[g(X_{i_0}) = 1 \middle| l\right] = 1$ always.

Let $\mathcal{T}'$ denote the protocol tree of $\mathcal{P}'$. For every leaf $l' \in \mathcal{T}'$, let $\lambda_{i_0}(l') \in \{0,1\}$ be a most likely value of Z_{i_0} when $Z \sim \mathcal{U}_{\{0,1\}^n}$, conditioned on reaching l'. Let

$$\delta_{i_0}(l') \stackrel{\text{def}}{=} \underset{Z\sim\mathcal{U}_{\{0,1\}^n}}{\mathbf{Pr}}\left[Z_{i_0} = \lambda_{i_0}(l')\middle|l'\right] - \frac{1}{2} \quad \left\{\in \left[0, \frac{1}{2}\right]\right\}$$

and

$$\delta_{i_0}(\mathcal{T}') \stackrel{\text{def}}{=} \underset{L'}{\mathbf{E}}\left[\delta_{i_0}(L')\right],$$

where L' is the distributed as the leaf of $\mathcal{T}'$ reached by $\mathcal{P}'$ when $Z \sim \mu_f$ (note the "mixture of distributions": $\lambda_{i_0}(l)$ and $\delta_{i_0}(l)$ are defined relative to $Z \sim \mathcal{U}_{\{0,1\}^n}$, but L' in the definition of $\delta_{i_0}(\mathcal{T}')$ is sampled with respect to $Z \sim \mu_f$).

Note several important symmetries in the above definitions with respect to the isomorphism $\mathcal{M}$. First, for every leaf l of $\mathcal{P}$

$$\lambda_{i_0}(l) = \lambda_{i_0}(\mathcal{M}(l)) \quad \text{and} \quad \delta_{i_0}(l) = \delta_{i_0}(\mathcal{M}(l)).$$

Second, the distribution of $\mathcal{M}(L)$, where L is the leaf of $\mathcal{T}$ reached by $\mathcal{P}$ when $X \sim \mu_f \circ \mu_g$ is the same as the distribution of the leaf of $\mathcal{T}'$ reached by $\mathcal{P}'$ when $Z \sim \mu_f$. Accordingly,

$$\delta_{i_0}(\mathcal{T}) = \delta_{i_0}(\mathcal{T}'). \tag{17}$$

These symmetries hold due to the fact that the construction of Corollary 1 only depends on μ_g – in particular, its guarantees hold both in the case of $X \sim \mu_g^{\mathcal{U}}$ and in the case of $X \sim \mu_f \circ \mu_g$.

4.1.1 The case of $\mathcal{P}'$

First we take a closer look at $\mathcal{T}'$, the tree of $\mathcal{P}'$. It is polarised, so we "statically" define $w_{i_0}(l') \in \{0,1\}$ such that conditioned on reaching the leaf $l' \in \mathcal{T}'$, if $w_{i_0} \neq *$, then $w_{i_0} = w_{i_0}(l')$. Let

$$q_{i_0}(l') \stackrel{\text{def}}{=} \underset{Z\sim\mathcal{U}_{\{0,1\}^n}}{\mathbf{Pr}}\left[w_{i_0} \neq *\middle|l'\right] = \underset{Z\sim\mathcal{U}_{\{0,1\}^n}}{\mathbf{Pr}}\left[Z_{i_0} \text{ has been queried by } \mathcal{P}'(Z)\middle|l'\right].$$ [9]

Obviously, under $Z \sim \mathcal{U}_{\{0,1\}^n}$ the best guess for the value of Z_{i_0} conditioned on reaching l' would be $w_{i_0}(l')$; therefore,

$$\lambda_{i_0}(l') = w_{i_0}(l').$$

The bits of Z are both unbiased and mutually independent under $\mathcal{U}_{\{0,1\}^n}$; accordingly, if protocol $\mathcal{P}'$ "knows something" about Z_{i_0}, then $w_{i_0} \neq *$ and the protocol knows that value with certainty:

$$\delta_{i_0}(l') = \frac{1}{2} \cdot (1 - q_{i_0}(l')) + q_{i_0}(l') - \frac{1}{2} = \frac{q_{i_0}(l')}{2}$$

[9] Note that if $q_{i_0}(l') = 0$, then $w_{i_0}(l') \in \{0,1\}$ can be defined arbitrarily – this is similar to the situation when $\delta_{i_0}(l) = 0$ in the case of $l \in \mathcal{P}$.

and

$$\delta_{i_0}(\mathcal{T}') = \mathop{\mathbf{E}}_{L'}\left[\delta_{i_0}(L')\right] = \frac{1}{2}\cdot\mathop{\mathbf{E}}_{L'}\left[q_{i_0}(L')\right], \tag{18}$$

where L' is distributed as the leaf of $\mathcal{T}'$ reached by $\mathcal{P}'$ when $Z \sim \mu_f$.

Next we want to use $\delta_{i_0}(\mathcal{T}')$ as an upper bound on the number of queries made by $\mathcal{P}'(Z)$ to Z_{i_0} under $Z \sim \mu_f$. The main obstacle here is the fact that $q_{i_0}(l')$ is defined with respect to $Z \sim \mathcal{U}_{\{0,1\}^n}$.[10]

Claim 2. *For every $z \in \{0,1\}^n$ such that $\mathcal{P}'(z)$ reaches l' with positive probability, it holds that*

$$\mathop{\mathbf{Pr}}_{Z\sim\mathcal{U}_{\{0,1\}^n}}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z)\middle|[l'], Z_{i_0} = z_{i_0}\right] = \mathbf{Pr}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(z)\middle|l'\right].$$

That is, the probability that Z_{i_0} is queried doesn't depend on $Z_{[n]\setminus\{i_0\}}$. Note that the right-hand side of the above equality only depends on the "internal randomness" of $\mathcal{P}'$ (and not on the distribution of Z).

Proofof Claim 2. Note that

$$\mathop{\mathbf{Pr}}_{\mathcal{U}_{\{0,1\}^n}}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z)\middle|[l'], Z_{i_0} = z_{i_0}\right] = \mathop{\mathbf{E}}_{\substack{Z'\sim\mathcal{U}_{\{0,1\}^n}\\ Z'_{i_0}=z_{i_0}}}\left[\mathbf{Pr}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z')\middle|l'\right]\right].$$

We claim that for every $z' \in \{0,1\}^n$, the value of $\mathbf{Pr}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(z')\middle|l'\right]$ is a function of z'_{i_0} – in particular, this means that the expectation on the right-hand side of the above equality is over a constant value equal to $\mathbf{Pr}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(z)\middle|l'\right]$ (quod erat demonstrandum).

Let $v_1, \ldots, v_t$ be the Z-nodes on the path from the root of $\mathcal{T}'$ to l' that may query Z_{i_0}, listed in order of appearance. For $j \in [t]$ let $a_j \in \{0,1\}$ be the "answer" of v_j on the path to l' and let $\mathbf{e}_j$ denote the event that Z_{i_0} is queried by $\mathcal{P}'(z')$ in v_j. Let v_j be a Z-node, parametrised by (i_0, α, β). Note that conditional on reaching v_j, the events $[\wedge_{k=1}^{j-1}(\neg\mathbf{e}_k)]$ and $[w_{i_0} = *]$ coincide.

If $a_j \neq z'_{i_0}$, then $\mathbf{Pr}\left[\mathbf{e}_j\right] = 0$.

If $a_j = z'_{i_0} = 0$, then

$$\mathbf{Pr}\left[a_j \text{ is answered}, \mathbf{e}_j\middle|v_j \text{ is reached}, \wedge_{k=1}^{j-1}(\neg\mathbf{e}_k)\right] = \alpha,$$

$$\mathbf{Pr}\left[a_j \text{ is answered}, \neg\mathbf{e}_j\middle|v_j \text{ is reached}, \wedge_{k=1}^{j-1}(\neg\mathbf{e}_k)\right] = (1-\alpha)\cdot(1-\beta)$$

$$\implies \quad \mathbf{Pr}\left[\mathbf{e}_j\middle|v_j \text{ is reached}, a_j \text{ is answered}, \wedge_{k=1}^{j-1}(\neg\mathbf{e}_k)\right] = \frac{\alpha}{\alpha + (1-\alpha)\cdot(1-\beta)}.$$

If $a_j = z'_{i_0} = 1$, then

$$\mathbf{Pr}\left[a_j \text{ is answered}, \mathbf{e}_j\middle|v_j \text{ is reached}, \wedge_{k=1}^{j-1}(\neg\mathbf{e}_k)\right] = \alpha,$$

$$\mathbf{Pr}\left[a_j \text{ is answered}, \neg\mathbf{e}_j\middle|v_j \text{ is reached}, \wedge_{k=1}^{j-1}(\neg\mathbf{e}_k)\right] = (1-\alpha)\cdot\beta$$

$$\implies \quad \mathbf{Pr}\left[\mathbf{e}_j\middle|v_j \text{ is reached}, a_j \text{ is answered}, \wedge_{k=1}^{j-1}(\neg\mathbf{e}_k)\right] = \frac{\alpha}{\alpha + (1-\alpha)\cdot\beta}.$$

[10] Note that the value of $\mathbf{Pr}_{Z\sim\nu}\left[Z_{i_0} \text{ has been queried by } \mathcal{P}'(Z)\middle|l'\right]$ is, in general, not "ν-independent" – in spite of the fact that the parameters of the generic nodes in $\mathcal{T}'$ are ν-independent, and therefore known.

As a_j, α and β are constants,

$$\mathbf{Pr}\left[\mathbf{e}_j \middle| v_j \text{ is reached, } a_j \text{ is answered, } \wedge_{k=1}^{j-1}(\neg \mathbf{e}_k)\right]$$

is a function of z'_{i_0}, as well as

$$\mathbf{Pr}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(z') \middle| l'\right] = \sum_{j=1}^{t} \mathbf{Pr}\left[\mathbf{e}_j \middle| v_j \text{ is reached, } a_j \text{ is answered, } \wedge_{k=1}^{j-1}(\neg \mathbf{e}_k)\right],$$

and the result follows. ■*Claim 2*

Let us decompose

$$\begin{aligned} q_{i_0}(l') &= \mathop{\mathbf{Pr}}_{Z\sim\mathcal{U}_{\{0,1\}^n}}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z) \middle| l'\right] \\ &= \mathop{\mathbf{E}}_{a\in\{0,1\}}\left[\mathop{\mathbf{Pr}}_{Z\sim\mathcal{U}_{\{0,1\}^n}}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z) \middle| [l'], Z_{i_0}=a\right]\right]. \end{aligned}$$

For every $z \in \{0,1\}^n$ such that $\mathcal{P}'(z)$ reaches l' with positive probability:

$$\begin{aligned} q_{i_0}(l') &\geq \frac{1}{2} \cdot \mathop{\mathbf{Pr}}_{Z\sim\mathcal{U}_{\{0,1\}^n}}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z) \middle| [l'], Z_{i_0}=z_{i_0}\right] \\ &= \frac{1}{2} \cdot \mathbf{Pr}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(z) \middle| l'\right], \end{aligned}$$

where the equality follows from Claim 2. So, for every distribution ν it holds that

$$\mathop{\mathbf{Pr}}_{Z\sim\nu}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z) \middle| l'\right] = \mathop{\mathbf{E}}_{Z'\sim\nu}\left[\mathbf{Pr}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z') \middle| l'\right]\right] \leq 2 \cdot q_{i_0}(l').$$

By (18),

$$\begin{aligned} \mathop{\mathbf{Pr}}_{Z\sim\mu_f}\left[Z_{i_0} \text{ is queried by } \mathcal{P}'(Z)\right] &= \mathop{\mathbf{E}}_{L'}\left[\mathbf{Pr}\left[Z_{i_0} \text{ is queried} \middle| L'\right]\right] \\ &\leq 2 \cdot \mathop{\mathbf{E}}_{L'}\left[q_{i_0}(L')\right] = 4 \cdot \delta_{i_0}(\mathcal{T}'), \end{aligned}$$

where L' is distributed as the leaf of $\mathcal{T}'$ reached by $\mathcal{P}'(Z)$ when $Z \sim \mu_f$. Then

$$\mathop{\mathbf{E}}_{Z\sim\mu_f}\left[\text{number of queries made by } \mathcal{P}'(Z)\right] \leq 4 \cdot \sum_{i=1}^{n} \delta_i(\mathcal{T}').$$

So,

$$\sum_{i=1}^{n} \delta_i(\mathcal{T}') \geq \frac{1}{4} \cdot \mathcal{R}_{\mu_f,\varepsilon}(f) \in \Omega(\mathcal{R}(f)). \tag{19}$$

4.1.2 The case of $\mathcal{P}$

From (19) and (17) we have:

$$\sum_{i=1}^{n} \delta_i(\mathcal{T}) \in \Omega(\mathcal{R}(f)), \tag{20}$$

where $\mathcal{T}$ is the tree of the given protocol $\mathcal{P}(X)$. Let us analyse $\mathcal{P}$, trying to obtain a lower bound on $\sum \delta_i(\mathcal{T})$.

For $x \in \{0,1\}^{n \cdot m}$, denote by $l(x)$ the leaf of $\mathcal{T}$ that is reached by $\mathcal{P}(x)$.[11] Let $\mu = \mu_f \circ \mu_g$, then

$$\delta_{i_0}(\mathcal{T}) = \mathop{\mathbf{E}}_{X \sim \mu} [\delta_{i_0}(l(X))] = \mathop{\mathbf{E}}_{X' \sim \mu} \left[\mathop{\mathbf{E}}_{X \sim \mu} \left[\delta_{i_0}(l(X)) \middle| X_{[n]\setminus\{i_0\}} = X'_{[n]\setminus\{i_0\}} \right] \right]$$

(note that $X_{[n]\setminus\{i_0\}}$ contains $(n-1)\cdot m$ bits). Let

$$\delta_{i_0}^{(x)}(\mathcal{T}) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{X \sim \mu} \left[\delta_{i_0}(l(X)) \middle| X_{[n]\setminus\{i_0\}} = x_{[n]\setminus\{i_0\}} \right],$$

then

$$\delta_{i_0}(\mathcal{T}) = \mathop{\mathbf{E}}_{X' \sim \mu} \left[\delta_{i_0}^{(X')}(\mathcal{T}) \right]. \tag{21}$$

Let $d_\mu(\mathcal{T})$ denote the expected number of oracle queries that $\mathcal{P}(X)$ makes when $X \sim \mu$ (this is the "expected depth" of $\mathcal{T}$). Let $d_\mu^{(i_0)}(\mathcal{T})$ denote the expected (total) number of queries to bits of X_{i_0} by $\mathcal{T}(X)$ when $X \sim \mu$, and let $d_\mu^{(i_0,x)}(\mathcal{T})$ denote the same expectation, conditioned upon $[X_{[n]\setminus\{i_0\}} = x_{[n]\setminus\{i_0\}}]$.

Obviously,

$$d_\mu(\mathcal{T}) = \sum_{i=1}^{n} d_\mu^{(i)}(\mathcal{T}) \quad \text{and} \quad d_\mu^{(i)}(\mathcal{T}) = \mathop{\mathbf{E}}_{X' \sim \mu} \left[d_\mu^{(i,X')}(\mathcal{T}) \right].$$

Since we can assume that $d_\mu(\mathcal{T}) \in O(\mathcal{R}(f \circ g^n))$,

$$\sum_{i=1}^{n} \mathop{\mathbf{E}}_{X' \sim \mu} \left[d_\mu^{(i,X')}(\mathcal{T}) \right] \in O(\mathcal{R}(f \circ g^n)). \tag{22}$$

Restricting and trimming $\mathcal{P}$ Now we only miss an upper bound on $\delta_{i_0}^{(x)}(\mathcal{T})$ in terms of $d_\mu^{(i_0,x)}(\mathcal{T})$ in order to be able to put together (20), (21) and (22).

In this part we construct a "restriction" of protocol $\mathcal{P}$, which will compute $g(\cdot)$ and whose accuracy and complexity will be closely related to $\delta_{i_0}^{(x)}(\mathcal{T})$ and $d_\mu^{(i_0,x)}(\mathcal{T})$, respectively. It will remain to state that if its accuracy is "noticeable", then the expected number of queries that it makes cannot be "negligible" – that will be done in Section 4.1.3 via choosing a suitable distribution μ_g.

[11] Note that $l(\cdot)$ is well-defined, as $\mathcal{P}$ is deterministic.

Recall the definitions of μ_g^0 and μ_g^1 from Section 3.2. Our ultimate μ_g will be such that

$$\mu_g = \frac{\mu_g^0 + \mu_g^1}{2} \tag{23}$$

– i.e., g will be unbiased with respect to it. For $i_0 \in [n]$ and $x \in \{0,1\}^{n \cdot m}$, denote by $\mu_g^{(i_0,x)}$ the distribution of $X_{i_0} \in \{0,1\}^m$ when $X \sim \mu = \mu_f \circ \mu_g$, conditioned upon $[X_{[n]\setminus\{i_0\}} = x_{[n]\setminus\{i_0\}}]$. Note that $\mu_g^{(i_0,x)}$ is a convex combination of μ_g^0 and μ_g^1.

We can easily turn $\mathcal{T}$ into a protocol for computing $g(X_{i_0})$: Label each leave l of the new protocol by $\lambda_{i_0}(l)$, as defined in the beginning of Section 4.1. When $\mathcal{T}$ queries a bit of X_{i_0}, the new protocol does the same, and whenever $\mathcal{T}$ queries a bit of $X_{[n]\setminus\{i_0\}}$, the new protocol "locally" substitutes the corresponding bit of some fixed $x \in \{0,1\}^{n \cdot m}$ (so, the value of x may affect protocol's behaviour). Denote this new (deterministic) protocol by $\mathcal{P}^{(i_0,x)}$, let us have a closer look at some of its properties.

Assume that $X_{i_0} \sim \mu_g$. The protocol $\mathcal{P}^{(i_0,x)}$ queries (only) bits of X_{i_0} and computes $g(X_{i_0})$ with some accuracy. We would like to use $\delta_{i_0}^{(x)}(\mathcal{T})$ as a "measure of accuracy" of $\mathcal{P}^{(i_0,x)}$ and $d_\mu^{(i_0,x)}(\mathcal{T})$ as its "measure of complexity".

Note that the value of $\delta_{i_0}^{(x)}(\mathcal{T})$ does not directly attest the accuracy of $\mathcal{P}^{(i_0,x)}$ under $X_{i_0} \sim \mu_g$, as $\delta_{i_0}^{(x)}(\mathcal{T})$ has been defined relative to $\mu_g^{(i_0,x)}$. Nevertheless,

$$\begin{aligned}
\delta_{i_0}^{(x)}(\mathcal{T}) &= \mathop{\mathbf{E}}_{\substack{X_{i_0} \sim \mu_g^{(i_0,x)} \\ X_{[n]\setminus\{i_0\}} = x_{[n]\setminus\{i_0\}}}} [\delta_{i_0}(l(X))] \\
&= \sum_{a \in \{0,1\}} \mathop{\mathbf{Pr}}_{X_{i_0} \sim \mu_g^{(i_0,x)}} [g(X_{i_0}) = a] \cdot \mathop{\mathbf{E}}_{\substack{X_{i_0} \sim \mu_g^{(i_0,x)} \\ X_{[n]\setminus\{i_0\}} = x_{[n]\setminus\{i_0\}}}} \left[\delta_{i_0}(l(X)) \middle| g(X_{i_0}) = a\right] \\
&\leq 2 \cdot \sum_{a \in \{0,1\}} \frac{1}{2} \cdot \mathop{\mathbf{E}}_{\substack{X_{i_0} \sim \mu_g^{(i_0,x)} \\ X_{[n]\setminus\{i_0\}} = x_{[n]\setminus\{i_0\}}}} \left[\delta_{i_0}(l(X)) \middle| g(X_{i_0}) = a\right] \\
&= 2 \cdot \mathop{\mathbf{E}}_{\substack{X_{i_0} \sim \mu_g \\ X_{[n]\setminus\{i_0\}} = x_{[n]\setminus\{i_0\}}}} [\delta_{i_0}(l(X))],
\end{aligned}$$

where the equality follows from (23) and the fact that $\delta_{i_0}(l) \geq 0$ always. Therefore,

$$\mathop{\mathbf{Pr}}_{X_{i_0} \sim \mu_g} \left[\mathcal{P}^{(i_0,x)}(X_{i_0}) = g(X_{i_0})\right] \geq \frac{1}{2} + \frac{\delta_{i_0}^{(x)}(\mathcal{T})}{2}. \tag{24}$$

Relating the query complexity of $\mathcal{P}^{(i_0,x)}$ to the value of $d_\mu^{(i_0,x)}(\mathcal{T})$ is more interesting, as the latter can be much smaller than the expected number of queries made by the protocol under μ_g.[12] In order to use both $\delta_{i_0}^{(x)}(\mathcal{T})$ and $d_\mu^{(i_0,x)}(\mathcal{T})$ as intended, we "trim" $\mathcal{P}^{(i_0,x)}$.

Define $\mathcal{P}_{tr}^{(i_0,x)}$ as the protocol obtained from (the tree of) $\mathcal{P}^{(i_0,x)}$, where every vertex v such that $\mathbf{Pr}_{\mu_g}\left[g(X_{i_0}) = a \middle| v\right] > 3/4$ for some $a \in \{0,1\}$ is replaced by a leaf labelled with

[12] E.g., if $\mathbf{Pr}\left[g(X_{i_0}) = 0\right] = 1/m$ under $X_{i_0} \sim \mu_g^{(i_0,x)}$, and $\mathcal{P}^{(i_0,x)}$ makes $\Omega(m)$ queries when $g(X_{i_0}) = 0$ and $O(1)$ queries when $g(X_{i_0}) = 1$, then $d_\mu^{(i_0,x)}(\mathcal{T}) \in O(1)$ but $\mathcal{P}^{(i_0,x)}$ makes $\Omega(m)$ expected queries under μ_g.

"a" (the sub-trees that were under these vertices are dropped). Like $\mathcal{P}$ and $\mathcal{P}^{(i_0,x)}$, $\mathcal{P}_{tr}^{(i_0,x)}$ is deterministic.

First we analyse the accuracy of $\mathcal{P}_{tr}^{(i_0,x)}$ under $X_{i_0} \sim \mu_g$. Here the "worst case" would be if we have trimmed sub-trees that correctly computed the value of $g(X_{i_0})$, in which case the accuracy in those vertices has reduced from $1/2 + 1/2$ to $1/2 + 1/4$. From (24),

$$\mathop{\mathbf{Pr}}_{X_{i_0}\sim\mu_g}\left[\mathcal{P}_{tr}^{(i_0,x)}(X_{i_0}) = g(X_{i_0})\right] \geq \frac{1}{2} + \frac{\delta_{i_0}^{(x)}(\mathcal{T})}{4}. \tag{25}$$

From the definitions, $d_\mu^{(i_0,x)}(\mathcal{T})$ equals the expected number of queries made by $\mathcal{P}^{(i_0,x)}$ when $X_{i_0} \sim \mu_g^{(i_0,x)}$. Therefore, under the same input distribution the expected number of queries made by $\mathcal{P}_{tr}^{(i_0,x)}$ is at most $d_\mu^{(i_0,x)}(\mathcal{T})$.

Assume without loss of generality that $\mathbf{Pr}_{\mu_g^{(i_0,x)}}[g(X_{i_0}) = 1] \geq \frac{1}{2}$. Let v be a non-leaf vertex of $\mathcal{P}_{tr}^{(i_0,x)}$ and $\mathbf{Pr}[v]$ be the probability that it is "visited" by $\mathcal{P}_{tr}^{(i_0,x)}$ on input X_{i_0}. From (23) it follows that

$$\begin{aligned}
&\frac{\mathbf{Pr}_{\mu_g^0}[v]}{\mathbf{Pr}_{\mu_g^1}[v]} = \frac{\mathbf{Pr}_{\mu_g}[v \text{ and } g(X_{i_0}) = 0]}{\mathbf{Pr}_{\mu_g}[v \text{ and } g(X_{i_0}) = 1]} = \frac{\mathbf{Pr}_{\mu_g}\left[g(X_{i_0}) = 0 \middle| v\right]}{\mathbf{Pr}_{\mu_g}\left[g(X_{i_0}) = 1 \middle| v\right]} \leq 3;\\
&\mathop{\mathbf{Pr}}_{\mu_g}[v] = \frac{\mathbf{Pr}_{\mu_g^0}[v] + \mathbf{Pr}_{\mu_g^1}[v]}{2} \leq 2 \cdot \mathop{\mathbf{Pr}}_{\mu_g^1}[v];\\
&\mathop{\mathbf{Pr}}_{\mu_g^{(i_0,x)}}[v] \geq \mathop{\mathbf{Pr}}_{\mu_g^{(i_0,x)}}[g(X_{i_0}) = 1] \cdot \mathop{\mathbf{Pr}}_{\mu_g^1}[v] \geq \frac{1}{2} \cdot \mathop{\mathbf{Pr}}_{\mu_g^1}[v];\\
&\mathop{\mathbf{Pr}}_{\mu_g^{(i_0,x)}}[v] \geq \frac{1}{4} \cdot \mathop{\mathbf{Pr}}_{\mu_g}[v].
\end{aligned}$$

The leaves of $\mathcal{P}_{tr}^{(i_0,x)}$ make no queries, so the expected number of queries made by the protocol under $X_{i_0} \sim \mu_g$ is at most 4 times that number under $\mu_g^{(i_0,x)}$. In other words,

$$\mathop{\mathbf{E}}_{X_{i_0}\sim\mu_g}\left[\text{number of queries made by } \mathcal{P}_{tr}^{(i_0,x)}(X_{i_0})\right] \leq 4 \cdot d_\mu^{(i_0,x)}(\mathcal{T}). \tag{26}$$

4.1.3 Choosing a suitable μ_g

We have seen so far that under our assumptions there existed a deterministic protocol that made $O\left(d_\mu^{(i_0,x)}(\mathcal{T})\right)$ expected queries and computed $g(X_{i_0})$ with accuracy $1/2 + \Omega\left(\delta_{i_0}^{(x)}(\mathcal{T})\right)$ under $X_{i_0} \sim \mu_g$. It remains to choose μ_g that would make $g(\cdot)$ hard to compute with *any* non-trivial advantage over randomly guessing the answer.

Lemma 2. *Let $g : \{0,1\}^m \to \{0,1,*\}$. There exists a distribution μ_g, such that for any $\delta > 0$, any deterministic protocol computing $g(Y)$ with accuracy $1/2 + \delta$ under $Y \sim \mu_g$ makes*

$$\Omega\left(\delta^2 \cdot \mathcal{R}(g)\right)$$

expected queries.

For us the order of quantifiers in this lemma is crucial: the claim holds for the same μ_g with respect to any δ. In particular, this means that g is balanced perfectly with respect to μ_g.[13] Accordingly, a non-trivial *deterministic* protocol cannot make less than 1 expected query (otherwise it would never make a query and had accuracy $1/2$). Therefore:

Corollary 2. *Let $g : \{0,1\}^m \to \{0,1,*\}$. There is a distribution μ_g, such that for any $\delta > 0$, any deterministic protocol computing $g(Y)$ with accuracy $1/2 + \delta$ under $Y \sim \mu_g$ makes*

$$\Omega\big(\delta^2 \cdot \mathcal{R}(g)\big) + 1$$

expected queries. In particular, $\mathbf{Pr}\left[g(Y) = 0\right] = \mathbf{Pr}\left[g(Y) = 1\right] = 1/2$.

If we choose such μ_g to be the "g-part" of the input distribution $X \sim \mu_f \circ \mu_g$, then from (25) and (26):

$$d_{\mu}^{(i_0,x)}(\mathcal{T}) \in \Omega\left(\left(\delta_{i_0}^{(x)}(\mathcal{T})\right)^2 \cdot \mathcal{R}(g) + 1\right) \subseteq \Omega\Big(\delta_{i_0}^{(x)}(\mathcal{T}) \cdot \sqrt{\mathcal{R}(g)}\Big). \tag{27}$$

Proofof Lemma 2. Let α be such that for every distribution ν, such that g is balanced with respect to it, there exists $\delta_\nu > 0$ and a (deterministic) query protocol ρ_ν that makes at most $\alpha \cdot \delta_\nu^2 \cdot \mathcal{R}(g)$ expected queries and computes $g(Y)$ with accuracy at least $1/2 + \delta_\nu$ when $Y \sim \nu$. We need to show that $\alpha \in \Omega(1)$. Assume $\alpha \leq 1$.

Let d_ν denote the expected number of queries that ρ_ν makes when $Y \sim \nu$. Obviously, we can assume that $d_\nu \in O(\mathcal{R}(g))$. If g is balanced with respect to ν, then any protocol computing it with accuracy $1/2 + \delta_\nu$ must make at least δ_ν expected queries; accordingly,

$$\alpha \cdot \delta_\nu^2 \cdot \mathcal{R}(g) \geq d_\nu \geq \delta_\nu \quad \Longrightarrow \quad d_\nu,\, \delta_\nu \geq \frac{1}{\alpha \cdot \mathcal{R}(g)} \geq \frac{1}{\mathcal{R}(g)}. \tag{28}$$

Let μ'_g be such that $\mathcal{R}_{\mu'_g, \frac{1}{3}}(g) \in \Omega(\mathcal{R}(g))$ and assume without loss of generality that g is balanced with respect to μ'_g. We will use our assumptions to build a (deterministic) protocol tree $\mathcal{T}_g$ for computing $g(Y)$ with high accuracy under $Y \sim \mu'_g$. Every non-leaf vertex $v \in \mathcal{T}_g$ will correspond to running a "weak protocol" that computes $g(Y)$ with accuracy $1/2 + \delta_v$ with respect to certain distribution ν_v (the outgoing edges will be labelled by the answer returned by that protocol). Every leaf will be labelled by the answer that $\mathcal{T}_g$ returns upon reaching it.

The tree is constructed inductively, where at every step we handle one vertex – that is, we decide whether it will corresponds to a leaf in $\mathcal{T}_g$, and if not, then we assign a weak protocol to this vertex. A *non-handled* vertex can only appear as a son of a vertex that has already been handled (and therefore all his predecessors starting from the root have been handled too).

Let $\mathcal{T}_g^{(i)}$ denote the partial protocol tree constructed at step i ($\mathcal{T}_g^{(0)}$ contains only the root v_{root}). Here is the i'th step of our construction:

(a) Let $v \in \mathcal{T}_g^{(i-1)}$ be a closest to the root non-handled vertex. Let ν'_v be the distribution of $Y \sim \mu'_g$, conditioned on reaching v by the protocol described by $\mathcal{T}_g^{(i-1)}$ (this is well-defined, as the actions of the v's predecessors are known).
If the entropy of $g(Y)$ when $Y \sim \nu'_v$ is at most $1/2$, let v be a leaf in $\mathcal{T}_g^{(i)}$ and label it by the more likely value of $g(Y)$.

[13] Which, in turn, means that μ_g is supported only on legal input values for g.

Otherwise, let ν_v be the "balanced version" of ν'_v, defined as $\frac{\nu_v^{(0)}+\nu_v^{(1)}}{2}$ where $\nu_v^{(a)}$ is the distribution of $Y \sim \nu'_v$, conditioned on $[g(Y) = a]$. Let ρ_v be a protocol that makes d_v expected queries and computes $g(Y)$ with accuracy at least $1/2 + \sqrt{\frac{d_v}{\alpha \cdot \mathcal{R}(g)}}$ when $Y \sim \nu_v$: its existence follows from (28). Let the action of v in $\mathcal{T}_g^{(i)}$ be ρ_v, and add to $\mathcal{T}_g^{(i)}$ two (non-handled) sons of v, corresponding to the possible answers given by ρ_v.

(b) If there is no non-handled vertices in $\mathcal{T}_g^{(i)}$, stop the construction and let $\mathcal{T}_g \stackrel{\text{def}}{=} \mathcal{T}_g^{(i)}$.

(c) Let $\widetilde{\mathcal{T}}_g^{(i)}$ be a modification of $\mathcal{T}_g^{(i)}$, where each non-handled vertex becomes a leaf labelled by a most likely value of $g(Y)$ conditioned on reaching that vertex when $Y \sim \mu'_g$. If

$$\mathop{\mathbf{Pr}}_{Y \sim \mu'_g}\left[\widetilde{\mathcal{T}}_g^{(i)}(Y) \neq g(Y)\right] \leq \frac{1}{3}, \tag{29}$$

stop the construction and let $\mathcal{T}_g \stackrel{\text{def}}{=} \widetilde{\mathcal{T}}_g^{(i)}$.

First we claim that if this construction halts, then

$$\mathop{\mathbf{Pr}}_{Y \sim \mu'_g}\left[\mathcal{T}_g(Y) \neq g(Y)\right] \leq \frac{1}{3}. \tag{30}$$

It is obviously the case if condition (29) has been satisfied; if not, then the construction has aborted at *(b)*, which means that all the leaves of $\mathcal{T}_g$ have been created at *(a)*. Then the entropy at every leaf of $\mathcal{T}_g$ is at most $1/2$ and $\mathbf{Pr}\left[\mathcal{T}_g(Y) \neq g(Y)\right] < 1/9$ (as $h_2(x) \leq 1/2 \Rightarrow x \notin [1/9, 8/9]$, where $h_2(\cdot)$ denotes the binary entropy function).

Let us argue that the construction halts and analyse the query complexity of $\mathcal{T}_g(Y)$ under $Y \sim \mu'_g$. For a (deterministic) protocol tree $\mathcal{T}$ that queries bits of Y, let

$$H_g(\mathcal{T}) \stackrel{\text{def}}{=} \mathop{\mathbf{E}}_{Y \sim \mu'_g}\left[\mathop{H}_{Y' \sim \mu'_g}\left(g(Y')\middle| Y' \in l_{\mathcal{T}}(Y)\right)\right],$$

where $H(\cdot)$ denotes the entropy and $l_{\mathcal{T}}(y)$ is the set of all $y' \in \{0,1\}^m$ that reach the same leaf of $\mathcal{T}$ as y does. For $i \geq 1$, let v_i be the vertex handled at step i and "$[v_i]$" denote the event $[v_i$ is reached by $\widetilde{\mathcal{T}}_g^{(i)}(Y)]$. Assume that v_i is not a leaf in $\mathcal{T}_g$, then [14]

$$\begin{aligned} H_g(\widetilde{\mathcal{T}}_g^{(i-1)}) - H_g(\widetilde{\mathcal{T}}_g^{(i)}) &= \mathop{\mathbf{Pr}}_{\mu'_g}[v_i] \cdot \left(\mathop{H}_{\mu'_g}\left(g(Y)\middle|[v_i]\right) - \mathop{H}_{\mu'_g}\left(g(Y)\middle|[v_i], \rho_{v_i}(Y)\right)\right) \\ &= \mathop{\mathbf{Pr}}_{\mu'_g}[v_i] \cdot \underbrace{\left(\mathop{H}_{\nu'_{v_i}}\left(g(Y)\right) - \mathop{H}_{\nu'_{v_i}}\left(g(Y)\middle|\rho_{v_i}(Y)\right)\right)}_{(*)}. \end{aligned} \tag{31}$$

Let us estimate $(*)$. Let $1/2 + \delta_{v_i}$ be the accuracy of $\rho_{v_i}(Y)$ in computing $g(Y)$ over ν_{v_i}. Denote $\alpha_a = \mathbf{Pr}_{\nu_{v_i}^{(a)}}[\rho_{v_i}(Y) = 1]$ for $a \in \{0,1\}$, then

$$\frac{(1-\alpha_0)+\alpha_1}{2} = \frac{1}{2} + \delta_{v_i} \quad \Longrightarrow \quad \alpha_1 - \alpha_0 = 2 \cdot \delta_{v_i}.$$

[14] Let $\widetilde{\mathcal{T}}_g^{(0)}$ consist of a single root-leaf vertex v_{root} labelled by "1".

Note that

$$\underset{\nu'_{v_i}}{H}\left(g(Y)\right) = h_2\left(\underset{A}{\mathbf{E}}\left[\alpha_A\right]\right) \quad \text{and} \quad \underset{\nu'_{v_i}}{H}\left(g(Y)\middle|\rho_{v_i}(Y)\right) = \underset{A}{\mathbf{E}}\left[h_2\left(\alpha_A\right)\right],$$

where A is a Boolean random variable distributed like $g(Y)$ under $Y \sim \nu'_{v_i}$.

Let us use Hölder's simple yet useful "defect estimation" for Jensen's inequality, as given in [Bec12]:

Fact 1 (*Hölder's estimation*)**.** *If $f : [a,b] \to \mathbb{R}$ is twice continuously differentiable and X is a (discrete) random variable taking values on $[a,b]$, then*

$$\mathbf{E}\left[f(X)\right] - f\left(\mathbf{E}\left[X\right]\right) = \frac{f''(x_0)}{2} \cdot \left(\mathbf{E}\left[X^2\right] - \left(\mathbf{E}\left[X\right]\right)^2\right) \quad \left\{= \frac{f''(x_0)}{2} \cdot \mathbf{Var}\left[X\right]\right\}$$

for some $x_0 \in [a,b]$.

Applying it with $h_2\left(\cdot\right)$ and α_A, we get:[15]

$$\underset{\nu'_{v_i}}{H}\left(g(Y)\right) - \underset{\nu'_{v_i}}{H}\left(g(Y)\middle|\rho_{v_i}(Y)\right) \geq \frac{\inf_{x\in(0,1)}\left\{-h_2''\left(x\right)\right\}}{2} \cdot \mathbf{Var}\left[\alpha_A\right] > 2 \cdot \mathbf{Var}\left[\alpha_A\right].$$

As v_i is not a leaf in $\mathcal{T}_g$,

$$\underset{\nu'_{v_i}}{H}\left(g(Y)\right) > \frac{1}{2} \quad \Longrightarrow \quad \frac{1}{10} < \underset{\nu'_{v_i}}{\mathbf{Pr}}\left[g(Y)=1\right] < \frac{9}{10}$$

and

$$\mathbf{Var}\left[\alpha_A\right] = \left(\alpha_0 - \alpha_1\right)^2 \cdot \left(1 - \underset{\nu'_{v_i}}{\mathbf{Pr}}\left[g(Y)=1\right]\right) \cdot \underset{\nu'_{v_i}}{\mathbf{Pr}}\left[g(Y)=1\right] > \frac{\delta_{v_i}^2}{3}.$$

So, (31) leads to

$$H_g(\widetilde{\mathcal{T}}_g^{(i-1)}) - H_g(\widetilde{\mathcal{T}}_g^{(i)}) > \frac{\mathbf{Pr}_{\mu'_g}\left[v_i\right] \cdot \delta_{v_i}^2}{2} \geq \frac{\mathbf{Pr}_{\mu'_g}\left[v_i\right] \cdot d_{v_i}}{2 \cdot \alpha \cdot \mathcal{R}(g)} \geq \frac{\mathbf{Pr}_{\mu'_g}\left[v_i\right]}{2 \cdot \left(\mathcal{R}(g)\right)^2}, \tag{32}$$

where the last two inequalities follow from (28).

Next we apply (31) to see that our construction of $\mathcal{T}_g$ always halts. Note that at *(a)* we always choose a non-handled vertex closest to the root, so at step i there are at most two "layers" of $\mathcal{T}_g^{(i)}$ that contain non-handled vertices. For $k \in \mathbb{N}$, define the k'th *stage* of construction as the collection of all steps that handle a vertex at depth k (observe that the steps of a stage always form an uninterrupted sequence).

Let i_0 and i_1 be the first and the last steps of stage k', assume that k' was not the last stage of the construction (recall that now we are proving halting). Let $V_1 \subseteq \mathcal{T}_g^{(i_1)}$ be the set of vertices (at depth k') where our construction has assigned a protocol during one of the steps of

[15] Formally speaking, $h_2''\left(\cdot\right)$ is continuous only on $(0,1)$; if $\alpha_0 = 0$ or $\alpha_1 = 1$, this violates the condition of Hölder's estimation, as stated above. However, the requirements of Fact 1 can, obviously, be relaxed by letting f be twice continuously differentiable on (a,b) only and continuous on $[a,b]$.

stage k', and let L_1 be the leaves of $\mathcal{T}_g^{(i_1)}$ at depth k' (i.e., these are vertices with conditional entropy of $g(Y)$ at most $1/2$). Observe that $V_1 \cup L_1 \subseteq \mathcal{T}_g^{(i_1)}$ is the set of vertices at depth k'. Then

$$H_g(\widetilde{\mathcal{T}}_g^{(i_0-1)}) - H_g(\widetilde{\mathcal{T}}_g^{(i_1)}) = \sum_{v \in V_1} \mathop{\mathbf{Pr}}_{\mu'_g}[v] \cdot \left(\mathop{H}_{\nu'_v}(g(Y)) - \mathop{H}_{\nu'_v}\left(g(Y) \middle| \rho_{v_i}(Y)\right) \right) \tag{33}$$
$$\geq \frac{\mathbf{Pr}_{\mu'_g}[\text{computation of } \mathcal{T}_g(Y) \text{ goes through } V_1]}{2 \cdot (\mathcal{R}(g))^2},$$

where the inequality is (32).

On the other hand, from the assumption that k' was not the last stage of the construction it follows that

$$\mathop{\mathbf{Pr}}_{Y \sim \mu'_g}\left[\widetilde{\mathcal{T}}_g^{(i_1)}(Y) \neq g(Y)\right] > \frac{1}{3}.$$

Since for every $l \in L_1$ it holds that $\mathbf{Pr}\left[\widetilde{\mathcal{T}}_g^{(i_1)}(Y) \neq g(Y) \middle| l\right] < 1/9$ (as $h_2(x) \leq 1/2 \Rightarrow x \notin [1/9, 8/9]$),

$$\begin{aligned}\mathop{\mathbf{Pr}}_{Y \sim \mu'_g}\left[\widetilde{\mathcal{T}}_g^{(i_1)}(Y) \neq g(Y)\right] &\leq \mathop{\mathbf{Pr}}_{\mu'_g}[\text{computation of } \mathcal{T}_g(Y) \text{ goes through } V_1] \cdot \frac{1}{2} \\ &\quad + \mathop{\mathbf{Pr}}_{\mu'_g}[\text{computation of } \mathcal{T}_g(Y) \text{ goes through } L_1] \cdot \frac{1}{9} \\ &= \frac{1}{9} + \mathop{\mathbf{Pr}}_{\mu'_g}[\text{computation of } \mathcal{T}_g(Y) \text{ goes through } V_1] \cdot \frac{7}{18},\end{aligned}$$

and therefore,

$$\mathop{\mathbf{Pr}}_{\mu'_g}[\text{computation of } \mathcal{T}_g(Y) \text{ goes through } V_1] > \frac{4}{7}.$$

From (33),

$$H_g(\widetilde{\mathcal{T}}_g^{(i_0-1)}) - H_g(\widetilde{\mathcal{T}}_g^{(i_1)}) > \frac{2}{7 \cdot (\mathcal{R}(g))^2}$$

and our construction halts after finitely-many steps (as every stage is obviously finite).

It remains to analyse the query complexity of $\mathcal{T}_g$ under μ'_g. Let $V \subset \mathcal{T}_g$ be the set of non-leaves and $S \subset \mathbb{N}$ be the steps of the construction where the elements of V were handled ($|V| = |S|$). Then

$$H_g(\widetilde{\mathcal{T}}) = H_g(\widetilde{\mathcal{T}}_g^{(0)}) - \sum_{i \in S}\left(H_g(\widetilde{\mathcal{T}}_g^{(i-1)}) - H_g(\widetilde{\mathcal{T}}_g^{(i)})\right) \leq 1 - \frac{1}{2 \cdot \alpha \cdot \mathcal{R}(g)} \cdot \sum_{v \in V} \mathop{\mathbf{Pr}}_{\mu'_g}[v] \cdot d_v,$$

where the inequality is (32), and

$$\sum_{v \in V} \mathop{\mathbf{Pr}}_{\mu'_g}[v] \cdot d_v \leq 2 \cdot \alpha \cdot \mathcal{R}(g).$$

The left-hand side of this inequality is the expected number of queries that $\mathcal{T}_g(Y)$ makes when $Y \sim \mu'_g$. The result follows from (30) and the assumption that $\mathcal{R}_{\mu'_g, \frac{1}{3}}(g) \in \Omega(\mathcal{R}(g))$. ■*Lemma 2*

4.2 Summing up: the complexities

From (20) and (21),

$$\sum_{i=1}^{n} \mathop{\mathbf{E}}_{X\sim\mu}\left[\delta_i^{(X)}(\mathcal{T})\right] \in \Omega(\mathcal{R}(f)).$$

From (27), for all $i \in [n]$ and $x \in \{0,1\}^{n\cdot m}$:

$$d_\mu^{(i,x)}(\mathcal{T}) \in \Omega\Big(\delta_i^{(x)}(\mathcal{T}) \cdot \sqrt{\mathcal{R}(g)}\Big).$$

Accordingly,

$$\sum_{i=1}^{n} \mathop{\mathbf{E}}_{X\sim\mu}\left[d_\mu^{(i,X)}(\mathcal{T})\right] \in \Omega\Big(\sqrt{\mathcal{R}(g)}\Big) \cdot \sum_{i=1}^{n} \mathop{\mathbf{E}}_{X\sim\mu}\left[\delta_i^{(X)}(\mathcal{T})\right] \subseteq \Omega\Big(\mathcal{R}(f)\cdot\sqrt{\mathcal{R}(g)}\Big).$$

By (22), this implies that

$$\Omega\Big(\mathcal{R}(f)\cdot\sqrt{\mathcal{R}(g)}\Big) \cap O(\mathcal{R}(f\circ g^n)) \neq \emptyset.$$

To conclude:

Theorem 1. *Let $f \subseteq \{0,1\}^n \times \Xi$ and $g : \{0,1\}^m \to \{0,1,*\}$. Then*

$$\mathcal{R}(f\circ g^n) \in \Omega\Big(\mathcal{R}(f)\cdot\sqrt{\mathcal{R}(g)}\Big).$$

5 Tightness: $\mathcal{R}(f\circ g^n) \in O\Big(\mathcal{R}(f)\cdot\sqrt{\mathcal{R}(g)}\Big)$ is possible

We construct a relation $f_0 \subseteq \{0,1\}^n \times \{0,1\}^n$ (i.e., $\Xi = \{0,1\}^n$) and a promise function $g_0 : \{0,1\}^n \to \{0,1,*\}$ (i.e., $m = n$), such that $\mathcal{R}(f_0) \in \Theta(\sqrt{n})$, $\mathcal{R}(g_0) \in \Theta(n)$ and $\mathcal{R}(f_0 \circ g_0^n) \in \Theta(n)$.

Let

$$f_0(z) \stackrel{\text{def}}{=} \left\{a \,\middle|\, |a+z| \le \frac{n}{2} - \sqrt{n}\right\}$$

and

$$g_0(x) \stackrel{\text{def}}{=} \begin{cases} 0 & \text{if } |x| \le n/2 - \sqrt{n}; \\ 1 & \text{if } |x| \ge n/2 + \sqrt{n}; \\ * & \text{otherwise.} \end{cases}$$

Claim 3. $\mathcal{R}(f_0) \in \Omega(\sqrt{n})$.

Proof. Assume that a deterministic protocol of cost k solves f_0 with respect to the uniform input distribution with error at most $1/4$. Such protocol partitions $\{0,1\}^n$ into (at most) 2^k sub-cubes, each marked by some "answer" (an element from $\{0,1\}^n$). In particular, more than 2^{n-1} points belong to sub-cubes of size at least 2^{n-k-1} – in other words, to sub-cubes of

co-dimension at most $k+1$. As more than half of all points belong to such sub-cubes and the total protocol error is at most $1/4$, there exists at least one sub-cube of co-dimension at most $k+1$, on which the protocol errs with probability less than $1/2$.

The symmetry in the definition of f_0 allows us to assume without loss of generality that the sub-cube is the set $\tau \stackrel{\text{def}}{=} 0^{k+1} \circ \{0,1\}^{n-k-1}$, where "$\circ$" denotes string concatenation. It is easy to see that the "answer" that would minimise the error probability with respect to this sub-cube can be any binary string starting with "0^{k+1}", so let us assume that the actual label is 0^n. Then

$$\mathbf{Pr}\left[error \middle| Z \in \tau\right] = \mathop{\mathbf{Pr}}_{Z' \in \{0,1\}^{n-k-1}}\left[\left|Z'\right| \leq \frac{n}{2} - \sqrt{n}\right] < \frac{1}{2},$$

which implies that $k+1 \geq 2\sqrt{n}$, as a uniformly-random binary string of length more than $n - 2\sqrt{n}$ would have more than $n/2 - \sqrt{n}$ "ones" with probability at least $1/2$. ■*Claim 3*

Claim 4. $\mathcal{R}(g_0) \in \Omega(n)$.

Proof. A randomised query protocol of cost k for g_0 would trivially imply existence of a randomised communication protocol of cost at most $2k$ for the bipartite problem *Gap-Hamming-Distance*:

$$GHD(X,Y) \stackrel{\text{def}}{=} \begin{cases} 0 & \text{if } |X \oplus Y| \leq n/2 - \sqrt{n}; \\ 1 & \text{if } |X \oplus Y| \geq n/2 + \sqrt{n}; \\ * & \text{otherwise,} \end{cases}$$

and it has been demonstrated by Chakrabarti and Regev [CR11] that the complexity of this problem is $\Omega(n)$. ■*Claim 4*

Claim 5. $\mathcal{R}_\varepsilon(f_0 \circ g_0^n) \in O\left(n \cdot \sqrt{\log(1/\varepsilon)}\right)$.

Proof. Consider the following protocol for computing $f_0\left(g_0(x_1), \ldots, g_0(x_n)\right)$, where $x_i \in \{0,1\}^n$: For every $i \in [n]$, let $a_i = x_i(j_i)$, where $j_i \in_R [n]$ – that is, a_i is a uniformly-random bit of x_i. Then $\left|\left\{i | a_i = g_0(x_i)\right\}\right|$ – the expected number of "correctly guessed" a_i's is at least $n/2 + \sqrt{n}$; intuitively, this means that the probability that $a_1, \ldots, a_n$ is a right answer to $f_0\left(g_0(x_1), \ldots, g_0(x_n)\right)$ is "non-trivially high" – to "boost" this probability, we will use several "probes" from every x_i and take their majority vote.

Protocol: For an odd integer t_ε as defined next, independently choose $j_{i,k} \in_R [n]$ for $i \in [n]$ and $k \in [t_\varepsilon]$. Let $a_i \stackrel{\text{def}}{=} maj\left(x_i(j_{i,1}), \ldots, x_i(j_{i,t_\varepsilon})\right)$ and output "$a_1, \ldots, a_n$".

To analyse it, we consider for every $i \in [n]$:

$$\begin{aligned}
&\mathbf{Pr}\left[a_i = g_0(x_i)\right] - \mathbf{Pr}\left[a_i \neq g_0(x_i)\right] \\
&\geq \sum_{i=0}^{\frac{t_\varepsilon - 1}{2}} \binom{t_\varepsilon}{i} \cdot \left(\left(\frac{1}{2} - \frac{1}{\sqrt{n}}\right)^i \left(\frac{1}{2} + \frac{1}{\sqrt{n}}\right)^{t_\varepsilon - i} - \left(\frac{1}{2} - \frac{1}{\sqrt{n}}\right)^{\frac{t_\varepsilon+1}{2}+i} \left(\frac{1}{2} + \frac{1}{\sqrt{n}}\right)^{\frac{t_\varepsilon-1}{2}-i}\right) \\
&= \left(1 - \left(\frac{1 - 2/\sqrt{n}}{1 + 2/\sqrt{n}}\right)^{\frac{t_\varepsilon+1}{2}}\right) \cdot \sum_{i=0}^{\frac{t_\varepsilon-1}{2}} \binom{t_\varepsilon}{i} \cdot \left(\frac{1}{2} - \frac{1}{\sqrt{n}}\right)^i \left(\frac{1}{2} + \frac{1}{\sqrt{n}}\right)^{t_\varepsilon - i},
\end{aligned}$$

where the equality occurs when $|x_i| - n/2 = \pm\sqrt{n}$. As

$$\sum_{i=0}^{\frac{t_\varepsilon-1}{2}} \binom{t_\varepsilon}{i} \cdot \left(\frac{1}{2} - \frac{1}{\sqrt{n}}\right)^i \left(\frac{1}{2} + \frac{1}{\sqrt{n}}\right)^{t_\varepsilon - i} = \mathbf{Pr}\left[a_i = g_0(x_i) \middle| |x_i| - \frac{n}{2} = \pm\sqrt{n}\right] > \frac{1}{2},$$

we get

$$\begin{aligned}&\mathbf{Pr}\left[a_i = g_0(x_i)\right] - \mathbf{Pr}\left[a_i \neq g_0(x_i)\right] \\ &\quad > \frac{1}{2} \cdot \left(1 - \left(\frac{1 - 2/\sqrt{n}}{1 + 2/\sqrt{n}}\right)^{\frac{t_\varepsilon+1}{2}}\right) > \frac{1}{2} \cdot \left(1 - \left(1 - \frac{2}{\sqrt{n}}\right)^{t_\varepsilon/2}\right) \geq \min\left\{\frac{t_\varepsilon}{4\sqrt{n}}, \frac{1}{4}\right\}.\end{aligned}$$

Our t_ε will be small enough to guarantee that $\frac{t_\varepsilon}{4\sqrt{n}} \leq \frac{1}{4}$, so we can write

$$\mathbf{Pr}\left[a_i = g_0(x_i)\right] > \frac{1}{2} + \frac{t_\varepsilon}{8\sqrt{n}}. \tag{34}$$

Now let us estimate the probability that $a_1, \ldots, a_n$ is a wrong answer to $f_0\left(g_0(x_1), \ldots, g_0(x_n)\right)$: This occurs only if $\left|\left\{i | a_i = g_0(x_i)\right\}\right| < n/2 + \sqrt{n}$, so by the Chernoff bound (in a form given in [DM05]),

$$\mathbf{Pr}\left[\text{the protocol errs}\right] < \exp\left(-\frac{1}{2} \cdot \left(\frac{t_\varepsilon}{8} - 1\right)^2\right),$$

so that choosing $t_\varepsilon \in \Theta\left(\sqrt{\log(1/\varepsilon)}\right)$ would suffice for our needs and the result follows. ■*Claim 5*

From Theorem 1 and Claims 3, 4 and 5:

Theorem 2. *For f_0 and g_0 as defined above,*

$$\mathcal{R}(f_0) \in \Theta\left(\sqrt{n}\right),\ \mathcal{R}(g_0) \in \Theta(n) \text{ and } \mathcal{R}(f_0 \circ g_0^n) \in \Theta(n).$$

6 Conclusions

We have seen that $\mathcal{R}(f \circ g^n) \in \Omega\left(\mathcal{R}(f) \cdot \sqrt{\mathcal{R}(g)}\right)$ for every relation f and promise function g, and this can be tight.

One may attempt to prove a more general lower bound by allowing g to be a relation as well (although the corresponding definition of the composed problem looks somewhat artificial).

On the other hand, it may be interesting to analyse the tightness of this lower bound in the following two more restricted cases:

- when both f and g are promise functions;
- when both f and g are total functions.

Depending on the answer, addressing this question may require either proving a stronger lower bound on the complexity of the composed problem or finding a tightness-witnessing example that would use more restricted type of computational problems than what we have seen in Section 5 (or – somewhat less likely – both).

References

[BDK16] S. Ben-David and R. Kothari. Randomized Query Complexity of Sabotaged and Composed Functions. *Proceedings of the 43rd International Colloquium on Automata, Languages and Programming*, pages 60:1–60:14, 2016.

[Bec12] R. Beckner. The Variance Drain and Jensen's Inequality. *CAEPR Working Paper*, 2012.

[CR11] A. Chakrabarti and O. Regev. An Optimal Lower Bound on the Communication Complexity of Gap-Hamming-Distance. *Proceedings of the 43rd Symposium on Theory of Computing*, pages 51–60, 2011.

[DM05] E. Drukh and Y. Mansour. Concentration Bounds for Unigram Language Models. *Journal of Machine Learning Research 6*, pages 1231–1264, 2005.