Electronic Colloquium on Computational Complexity, Revision 4 of Report No. 54 (2017)

Lifting randomized query complexity to randomized communication complexity

Anurag Anshu[†] Naresh B. Goud[†] Rahul Jain [*] Srijita Kundu[†]
Priyanka Mukhopadhyay[†]

June 20, 2017

Abstract

We show that for any (partial) query function $f : \{0,1\}^n \to \{0,1\}$, the randomized communication complexity of f composed with Index_m^n (with $m = \text{poly}(n)$) is at least the randomized query complexity of f times $\log n$. Here $\text{Index}_m : [m] \times \{0,1\}^m \to \{0,1\}$ is defined as $\text{Index}_m(x,y) = y_x$ (the xth bit of y).

Our proof follows on the lines of Raz and Mckenzie [RM99] (and its generalization due to [GPW15]), who showed a lifting theorem for deterministic query complexity to deterministic communication complexity. Our proof deviates from theirs in an important fashion that we consider partitions of rectangles into many sub-rectangles, as opposed to a particular sub-rectangle with desirable properties, as considered by Raz and McKenzie. As a consequence of our main result, some known separations between quantum and classical communication complexities follow from analogous separations in the query world.

1 Introduction

Communication complexity and query complexity are two concrete models of computation which are very well studied. In the communication model there are two parties Alice, with input x and Bob, with input y, and they wish to compute a joint function $f(x,y)$ of their inputs. In the query model one party Alice tries to compute a function $f(x)$ by querying bits of a database string x. There is a natural way in which a query protocol can be viewed as a communication protocol between Alice, with no input, and Bob, with input x, in which the only communication allowed is queries to the bits of x and answers to these queries. Given this, we can (informally) view the query model as a "simpler" sub-model of the communication model. Indeed several results in query complexity are easier to argue and obtain than the corresponding results in communication complexity. One interesting technique that is often employed with great success is to first show a result in the query model and then "lift" it to a result in the communication model via some "lifting theorem".

[*]Centre for Quantum Technologies, National University of Singapore and MajuLab, UMI 3654, Singapore.
[†]Center for Quantum Technologies, National University of Singapore, Singapore.

ISSN 1433-8092

One of the first such lifting theorems was shown by Raz and McKenzie [RM99] (and its generalization by [GPW15]). For a (partial) query function $f : \{0,1\}^n \to \{0,1\}$ and a communication function $g : \{0,1\}^m \times \{0,1\}^k \to \{0,1\}$ let the composed function $f \circ g^n$ be defined as $f \circ g^n((x_1, y_1), \ldots, (x_n, y_n)) = f(g(x_1, y_1), \ldots, g(x_n, y_n))$. Raz and McKenzie [RM99] (and the generalization due to [GPW15]) showed that for every query function $f : \{0,1\}^n \to \{0,1\}$ the deterministic communication complexity of f composed with Index_m (with $m = \mathrm{poly}(n)$) is at least the deterministic query complexity of f times $\log n$. Here $\mathrm{Index}_m : [m] \times \{0,1\}^m \to \{0,1\}$ is defined as $\mathrm{Index}_m(x, y) = y_x$ (the xth bit of y). Subsequently several lifting theorems for different complexity measures have been shown, example lifting *approximate-degree* to *approximate-rank* [She11] and *approximate junta-degree* to *smooth-corruption-bound* [GLM+15] etc.

Our result

In this work we show lifting of (bounded error) randomized query complexity to (bounded error) randomized communication complexity. For a (partial) query function $f : \{0,1\}^n \to \{0,1\}$ let the randomized query complexity with (worst-case) error $\varepsilon > 0$ of f be denoted $\mathrm{R}_\varepsilon(f)$. Similarly for a communication function $g : \{0,1\}^m \times \{0,1\}^k \to \{0,1\}$, let the randomized communication complexity with (worst-case) error $\varepsilon > 0$ of g be denoted $\mathrm{R}_\varepsilon(g)$. We show the following.

Theorem 1. *For all (partial) functions $f : \{0,1\}^n \to \{0,1\}$,*

$$\mathrm{R}_{1/4}(f \circ \mathrm{Index}_m^n) = \Omega(\mathrm{R}_{1/3}(f) \cdot \log n),$$

where $m = \mathrm{poly}(n)$ [1].

On the other hand it is easily seen with a simple simulation of a query protocol using a communication protocol that $\mathrm{R}_{1/3}(f \circ \mathrm{Index}_m^n) = O(\mathrm{R}_{1/3}(f) \cdot \log m)$. This implies $\mathrm{R}_{1/3}(f \circ \mathrm{Index}_m^n) = \Theta(\mathrm{R}_{1/3}(f) \cdot \log n)$ with $m = \mathrm{poly}(n)$.

Our result implies a recent result of [ABBD+16] where they exhibited a power 2.5 separation between classical randomized and quantum communication complexities for a total function. It also implies exponential separation between two-round quantum communication complexity and randomized communication complexity first shown by [Raz99]

Our techniques

Our techniques are partly based on the techniques of Raz and McKenzie [RM99] as presented in [GPW15] with an important modification to deal with distributional error protocols instead of deterministic protocols. Let T be a deterministic communication protocol tree for $f \circ \mathrm{Index}_m^n$ (with $m = \mathrm{poly}(n)$). We use this to create a randomized query protocol Π (see Algorithm 2) for f. Let z be an input for which we are supposed to output $f(z)$. We start with the root of T and continue to simulate T (using randomness) till we find a co-ordinate $i \in [n]$ where T has *worked enough* so that $\mathrm{Index}_m(x_i, y_i)$ is becoming (only slightly) *determined*. Using the properties of Index_m we conclude that T must have communicated $O(\log n)$ bits by now. We go ahead a query z_i (the ith bit of z) and *synchronize* with z_i, that is go to the appropriate sub-event of the current node in T consistent with z_i.

To keep the unqueried bits z_i sufficiently *undetermined*, firstly we need the number of possible y_i in each of these locations to be large. Since the set of y is exponentially larger than the set of x to begin with, the communication protocol lets us do this for free. To make sure z_i is not too determined by x_i, we need to make some effort. Suppose the set of unqueried indices in z is $I \subseteq [n]$. The parameter we use to keep track of how much T has worked to determine x_i,

[1] We state our result for Boolean functions f, however it holds for general relations. Moreover, for convenience we use a partial version of the Index_m^n function defined on the subset Bal_m^n of $\{0,1\}^{mn}$ that only contains strings with an equal number of 0's and 1's in every block.

is the conditional probability (within the set A of all possible x at this point) of a particular value of x_i, given a particular value $x_{I\setminus\{i\}}$ at other unqueried locations. When this conditional probability $p_A(x_i|x_{I\setminus\{i\}})$ becomes too high for a sufficiently large number of strings, we conclude that we cannot maintain the conditional probability to be low for this i any more and a query must be made.

However, we only want to make a query in that part of A where the conditional probability violation takes place – this eventually lets us compare the number of queries we make with the number of bits communicated. So within a query subroutine, we first probabilistically split A into the strings HIGH where the conditional probability becomes too high, and $A \setminus$ HIGH, where this does not happen, according to their respective sizes. A query is then made in the HIGH part, and only the x_i and y_i that are consistent with the z_i that we learn from the query are retained – in such a way that they still form a rectangle. After we have done the HIGH, $A \setminus$ HIGH splitting and querying in all indices where the conditional probability $p_A(x_i|x_{I\setminus\{i\}})$ was too high for many strings, the conditional probability can be restored to a low enough value for the rest of the indices, and we can move on with communication steps.

As long as we have a bound $p_{\max}$ on the maximum $p_A(x_i|x_{I\setminus\{i\}})$ in A and the set B of all possible y is large, the unqueried z_i are sufficiently undetermined and we can move from node to node of T according to the "flow" of T, for every input z. We prove this in parts in the Partition Lemma 5 in section 2 and Lemma 11 in section 3. This lets our algorithm sample the leaves of T close to their original probabilities, and thus the correctness of T on (x, y) in expectation ensures the correctness of our algorithm on z on expectation.

During the course of our simulation, we may end up at some "bad" subevents, where we will not be able to maintain a sufficiently large number of (x, y) consistent with z. We need to abort the algorithm on such subevents. When we have a $p_{\max}$ bound, B is large and we are going with the "flow" of T, we can ensure that the probability of going to such bad subevents is small. But, if we need to do a series of queries in one go, we will not be able to maintain the requisite $p_{\max}$ bound in between queries. So it may be possible that when we do a query and try to synchronize x_i and y_i with z_i, we do not find any (or sufficiently many in the form of a rectangle) x_i and y_i that are consistent with z_i. In the technical Lemma 7 we show that there is a way around this, that in fact if we do some "preprocesing" on A before carrying out queries, the probability of this bad subevent happening is still small. Thus the algorithm does not abort with high probability, and ends up at the leaves with the correct probabilities.

2 Preliminaries

In this section, we present some notations and basic lemmas needed for the proof of our main result.

Let $f : \{0,1\}^n \to \{0,1\}$ be a (partial) function. Let $\varepsilon > 0$ be an error parameter. Let the randomized query complexity, denoted $\mathrm{R}_\varepsilon(f)$, be the maximum number of queries made by the best randomized query protocol computing f with error at most ε on any input $x \in \{0,1\}^n$. Let θ be a distribution on $\{0,1\}^n$. Let the distributional query complexity, denoted $\mathrm{D}^\theta_\varepsilon(f)$, be the maximum number of queries made by the best deterministic query protocol computing f with average error at most ε under θ. The distributional and randomized query complexities are related by the following Yao's Lemma.

Fact 2 (Yao's Lemma)**.** *Let $\varepsilon > 0$. We have $\mathrm{R}_\varepsilon(f) = \max_\theta \mathrm{D}^\theta_\varepsilon(f)$.*

Similarly, we can define randomized and distributional communication complexities with a similar Yao's Lemma relating them.

Let λ be a *hard* distribution on $\{0,1\}^n$ such that $\mathrm{D}^\lambda_{1/3}(f) = \mathrm{R}_{1/3}(f)$, as guaranteed by Yao's Lemma. Let $m = n^{100}$ and let $\mathrm{Bal}_m \subset \{0,1\}^m$ be the set of all strings of length m with equal number of 0's and 1's. Observe that $|\mathrm{Bal}_m| = \binom{m}{m/2} \geq 2^m/\sqrt{m}$. The notation Bal^n_m

will refer to the set $\mathrm{Bal}_m \times \mathrm{Bal}_m \times \ldots \mathrm{Bal}_m$. Let $\mathrm{Index}_m : [m] \times \mathrm{Bal}_m \to \{0,1\}$ be defined as $\mathrm{Index}_m(x,y) = y_x$ (the xth bit of y). Consider the following *lifted* distribution for the composed function $f \circ \mathrm{Index}_m^n$: $\mu(x,y) = \lambda(G(x,y))/|G^{-1}(G(x,y))|$, where $G := \mathrm{Index}_m^n$. We observe for this distribution that $\mu(x)$ and $\mu(y)$ are uniform in their support.

Let Alice and Bob's inputs for the composed function be respectively $x = (x_1, \ldots, x_n) \in [m]^n$ and $y = (y_1, \ldots, y_n) \in \mathrm{Bal}_m^n$.

We use the following notation, some of which is adapted from notation used in [GPW15].

- For a node v in a communication protocol tree, let $R^v = X^v \times Y^v$ denote its associated rectangle. If Alice or Bob send the bit b at v, let v_b be the corresponding child of v and $X^{v,b} \subseteq X^v$ and $Y^{v,b} \subseteq Y^v$ be the set of inputs of Alice and Bob respectively on which they do this.
- For a string $x \in [m]^n$ and an interval $I \subset [n]$, let x_I be the restriction of x to the interval I. We use shorthand x_i for $x_{\{i\}}$. We use similar notation for string $y \in \mathrm{Bal}_m^n$.
- For a set $A \subset [m]^n$, let $A_I := \{x_I : x \in A\}$ be the restriction of A to the interval I and $A_{x_I} = \{x' \in A : x'_I = x_I\}$. Our convention is for an $x_I \notin A_I$, A_{x_I} is just the null set. We use similar notation for B.
- We say $B \subseteq \mathrm{Bal}_m^n$ is *large* with respect to $I \subseteq [n]$ if B is fixed outside of the interval I (ie, $|B_I| = |B|$) and $\frac{|B_I|}{|\mathrm{Bal}_m|^{|I|}} \geq 2^{-n^2(2n-|I|)}$.
- For $A \subseteq [m]^n$, an interval J, we define a uniform probability distribution on strings x in A and restriction of x to J in A as, $p_A(x_J) := \frac{|A_{x_J}|}{|A|}$. We use similar notation for B.
- For $J \subseteq [n]$ and $i \notin J$, we define the conditional probability of x_i given x_J in A such that $p_A(x_J) > 0$, as $p_A(x_i|x_J) := \frac{p_A(x_i \circ x_J)}{p_A(x_J)}$.
- For $A \subseteq [m]^n$, an index $i \in I$ and $x_{I\setminus\{i\}} \in [m]^{|I|-1}$, let

$$p_{\max}(A, x_{I\setminus\{i\}}) := \max_{x_i} p_A(x_i|x_{I\setminus\{i\}}).$$

  We say a $p_{\max}$ bound of α holds for A with respect to I, if $p_{\max}(A, x_{I\setminus\{i\}}) \leq \alpha$ for all $i \in I$ and all such $x_{I\setminus\{i\}}$.
- For $A \subseteq [m]^n, I \subseteq [n]$ and $i \in I$, let

$$\textsc{High}(A, \alpha, i) := \{x \in A : p_A(x_i|x_{I\setminus\{i\}}) > \alpha\}.$$

  Moreover, we use $\textsc{High}(A, \alpha, I)$ to denote $\cup_{i \in I} \textsc{High}(A, \alpha, i)$.
- For $y_i \in \mathrm{Bal}_m$ and $z_i \in \{0,1\}$, let $U(y_i, z_i) := \{x_i \in [m] : \mathrm{Index}_m(x_i, y_i) = z_i\}$. We use $A|_{U(y_i,z_i)}$ to denote $\cup_{x_i \in U(y_i,z_i)} A_{x_i}$.

Fact 3 (Chernoff bound)**.** *Let $X_1, X_2 \ldots X_t$ be independent random variables such that $0 < X_i < c$. Let $X = \sum_i X_i$ and $\mu = \mathbb{E}(X)$. Then for $\delta > 0$, it holds that*

$$\Pr[|X - \mu| \geq \mu\delta] \leq 2e^{-\frac{\delta^2\mu^2}{tc^2}}.$$

We shall use the sets $\textsc{Small}(A, A', I), \textsc{Bad}(A, B, i)$ and $\textsc{UnBal}(A, B, I)$ defined in the next Lemmas 4, 5 and 7 respectively in our algorithm and analysis.

Lemma 4. *For $A \subseteq [m]^n, I \subseteq [n]$ and $A' \subseteq A$, there exists $A'' \subseteq A'$ such that for all $i \in I$ and $x_{I\setminus\{i\}} \in A''_{I\setminus\{i\}}$, $|A''_{x_{I\setminus\{i\}}}| \geq \frac{1}{n^3}|A_{x_{I\setminus\{i\}}}|$. Further, if we denote $A' \setminus A''$ by $\textsc{Small}(A, A', I)$, then*

$$\sum_{x \in \textsc{Small}(A,A',I)} p_A(x) < \frac{1}{n^2}.$$

Algorithm 1: Decomposing $A' = A'' \cup \textsc{Small}(A, A', I)$

Initialize $A^0 = A', j = 0$
while $|A^j_{x_{I\setminus\{i\}}}| < \frac{1}{n^3}|A_{x_{I\setminus\{i\}}}|$ *for some* $i \in I$ *and* $x_{I\setminus\{i\}} \in A^j_{I\setminus\{i\}}$ **do**
  Pick such an $i \in I$ and $x_{I\setminus\{i\}}$
  Set $A^{j+1} = A^j \setminus \{x' \in A^j : x'_{I\setminus\{i\}} = x_{I\setminus\{i\}}\}$
end
Output $A'' = A^j$

Proof. The set A'' is obtained by running the following algorithm on A'. It is easy to see that the A'' obtained satisfies the property required.
To bound the size of $\textsc{Small}(A, A', I)$, let $(i_j, x^j_{I\setminus\{i_j\}})$ be the pair picked by the algorithm in the j-th iteration. Then we have, $|A^j_{x_{I\setminus\{i_j\}}}| < \frac{1}{n^3}|A_{x_{I\setminus\{i_j\}}}|$. Note that a particular $x^j_{I\setminus\{i_j\}}$ can only be removed once in the algorithm, so at most all $x_{I\setminus\{i\}}$ for all $i \in I$ can be removed. So the total strings removed is at most

$$\sum_{(i_j, x^j_{I\setminus\{i_j\}})} |A^j_{x^j_{I\setminus\{i_j\}}}| < \frac{1}{n^3} \sum_{(i_j, x^j_{I\setminus\{i_j\}})} |A_{x^j_{I\setminus\{i_j\}}}| \leq \frac{1}{n^3} \sum_{i\in I} \sum_{x_{I\setminus\{i\}}} |A_{x_{I\setminus\{i\}}}| = \frac{|I|}{n^3}|A| \leq \frac{1}{n^2}|A|.$$

This proves the lemma. □

Lemma 5 (Partition Lemma)**.** *For $A \subseteq [m]^n$ suppose a $p_{\max}$ bound of $m^{-8/10}$ holds with respect to $I \subseteq [n]$. For a $B \subseteq \mathrm{Bal}^n_m$ that is large with respect to I, we say a $y_i \in B_{\{i\}}$ is in $\textsc{Bad}(A, B, i)$ if it does not satisfy the property*

$$\sum_{x_i \in U(y_i, z_i)} p_A(x_i | x_{I\setminus\{i\}}) \in \left[\frac{1}{2} - n^{-5}, \frac{1}{2} + n^{-5}\right] \qquad \forall x_{I\setminus\{i\}} \in A_{I\setminus\{i\}} \quad \forall z_i \in \{0, 1\}$$

or B_{y_i} is not large with respect to $I \setminus \{i\}$. Then,

$$\sum_{y_i \in \textsc{Bad}(A,B,i)} p_B(y_i) \leq 2^{-n^2+1}.$$

Proof. For a fixed index $i \in I$ and fixed $z_i \in \{0, 1\}$ we say $y_i \in \mathrm{Bal}_m$ satisfy the property $P(x_{I\setminus\{i\}}, z_i)$ for set A if it holds that

$$\frac{1}{2} + \delta \geq \sum_{x_i \in U(y_i, z_i)} p_A(x_i | x_{I\setminus\{i\}}) \geq \frac{1}{2} - \delta \qquad \forall x_{I\setminus\{i\}} \in [m]^{|I|-1}.$$

Consider an indicator $\mathbb{1}_{g(x_i,y_i)=z_i}$ for the event $g(x_i, y_i) = z_i$ (say the function $(1 - (y_i)_{x_i} \oplus z_i)$). We can rewrite

$$\sum_{x_i \in U(y_i, z_i)} p_A(x_i | x_{I\setminus\{i\}}) = \sum_{x_i} \mathbb{1}_{g(x_i,y_i)=z_i} \cdot p_A(x_i | x_{I\setminus\{i\}}).$$

Note that $B_{\{i\}} \subseteq \mathrm{Bal}_m$ and when y_i is drawn uniformly from any subset of Bal_m, expectation value of the above quantity is $\frac{1}{2}$, as the expectation value of $\mathbb{1}_{g(x_i,y_i)=z_i}$ is $\frac{1}{2}$. Thus, using Chernoff bound (Fact 3) and the fact that for any $x_{I\setminus\{i\}}$, $0 \leq \mathbb{1}_{g(x_i,y_i)=z_i} \cdot p_A(x_i | x_{I\setminus\{i\}}) \leq p_A(x_i | x_{I\setminus\{i\}}) \leq p_{\max}(A, x_{I\setminus\{i\}}) \leq m^{-8/10}$, we have

$$\Pr_{y_i \sim \frac{1}{|B_{\{i\}}|}} \left[\left|\sum_{x_i} \mathbb{1}_{g(x_i,y_i)=z_i} \cdot p_A(x_i | x_{I\setminus\{i\}}) - \frac{1}{2}\right| > \delta\right] \leq 2e^{-\frac{\delta^2}{4m \cdot m^{-16/10}}} = 2e^{-\frac{m^{3/5} \cdot \delta^2}{4}}.$$

That is, the probability that y_i does not satisfy $P(x_{I\setminus\{i\}}, z_i)$ for any fixed $x_{I\setminus\{i\}}$ and z_i is at most $2e^{-\frac{m^{3/5}\cdot\delta^2}{4}}$. Thus the probability that y_i does not satisfy property $P(x_{I\setminus\{i\}}, z_i)$ for at least one $x_{I\setminus\{i\}}$ or one z_i is at most $4m^{|I|-1}\cdot e^{-\frac{m^{3/5}\cdot\delta^2}{4}} \leq m^n \cdot e^{-\frac{m^{3/5}\cdot\delta^2}{4}} = 2^{n\log m}\cdot e^{-\frac{m^{3/5}\cdot\delta^2}{4}}$. We say $y_i \in \textsc{Bad}_1(A,B,i)$ if it does not satisfy $P(x_{I\setminus\{i\}}, z_i)$ for at least one $x_{I\setminus\{i\}}$ or z_i. Taking $\delta = m^{-1/20}$, we get the number of $y_i \in \textsc{Bad}_1(A,B,i)$ to be at most $|B_{\{i\}}|\cdot 2^{n\log m}e^{-\frac{m^{1/2}}{4}} \leq |\mathrm{Bal}_m|\cdot 2^{-n^{50}}$. Since for any y_i, $|B_{y_i}| \leq |\mathrm{Bal}_m|^{|I|-1}$, we get,

$$\sum_{y_i\in\textsc{Bad}_1(A,B,i)} p_B(y_i) \leq \frac{|\mathrm{Bal}_m|\cdot 2^{-n^{50}}\cdot|\mathrm{Bal}_m|^{|I|-1}}{|\mathrm{Bal}_m|^{|I|}\cdot 2^{-n^2(2n-|I|)}} \leq 2^{-n^{45}}.$$

Now we say $y_i \in \textsc{Bad}_2(A,B,i)$ if $\frac{|(B_{y_i})_I|}{|\mathrm{Bal}_m|^{|I|-1}} < 2^{-n^2(2n-|I|+1)}$. Then,

$$\sum_{y_i\in\textsc{Bad}_2(A,B,i)} p_B(y_i) \leq \frac{|\mathrm{Bal}_m|\cdot|\mathrm{Bal}_m|^{|I|-1}2^{-n^2(2n-|I|+1)}}{|\mathrm{Bal}_m|^{|I|}\cdot 2^{-n^2(2n-|I|)}} \leq 2^{-n^2}.$$

Note that $\textsc{Bad}(A,B,i) = \textsc{Bad}_1(A,B,i)\cup\textsc{Bad}_2(A,B,i)$, which gives us the required bound on $\sum_{y_i\in\textsc{Bad}(A,B,i)} p_B(y_i)$. □

The following lemma follows from a result in [GPW17], which we state below. Here, $\chi(z) = (-1)^{\oplus z_i}$.

Lemma 6 ([GPW17], Equation 4)**.** *For an interval $I\subseteq[n]$, suppose $p_A(x_I) \leq m^{-0.7|I|}$ for all x_I (min entropy is at least $0.7|I|\log m$) and B is large. Then it holds that*

$$|\sum_{x_I,y_I} p_A(x_I)p_B(y_I)\chi(g^I(x_I,y_I))| \leq 2^{-5|I|\log n}.$$

Lemma 7. *For $A\subseteq[m]^n$ suppose a $p_{\max}$ bound of $m^{-8/10}$ holds with respect to $I\subseteq[n]$ and $B\subseteq\mathrm{Bal}_m^n$ is large with respect to I. We say an $x\in\textsc{UnBal}(A,B,I)$ if it does not satisfy the property*

$$\Pr_{y_I\sim\frac{|B_{y_I}|}{|B|}}[g^I(x_I,y_I)=z_I] \in \frac{1}{2^{|I|}}\left[1-n^{-3},1+n^{-3}\right] \qquad \forall z\in\{0,1\}^{|I|}.$$

Then,

$$p_A(\textsc{UnBal}(A,B,I)) \leq \frac{1}{n^8}.$$

Proof. Fix an interval $J\subseteq I$. Since a $p_{\max}$ bound of $m^{-8/10}$ holds for the given A, we have that for all x_J, $p_A(x_J)\leq m^{-0.8|J|}$. For any subset $A'_J\subseteq A_J$ such that $p_A(A'_J)\geq m^{-0.1|J|}$, we have that $p_{A'_J}(x_J)\leq m^{0.1|J|}m^{-0.8|J|}\leq m^{-0.7|J|}$. Thus, invoking Lemma 6, we obtain that

$$|\sum_{x_J,y_J} p_{A'_J}(x_J)p_B(y_J)\chi(g^J(x_J,y_J))| \leq 2^{-5|J|\log n}. \tag{1}$$

Let $\textsc{Bad}_J^{(1)}$ be the set of all $x_J\in A_J$ for which $\sum_{y_J} p_B(y_J)\chi(g^J(x_J,y_J)) \geq 2^{-5|J|\log n}$ and $\textsc{Bad}_J^{(0)}$ be the set of all $x_J\in A_J$ for which $\sum_{y_J} p_B(y_J)\chi(g^J(x_J,y_J)) \leq -2^{-5|J|\log n}$. Let $\textsc{Bad}_J := \textsc{Bad}_J^{(1)}\cup\textsc{Bad}_J^{(0)}$.We conclude that

$$\Pr_A(\textsc{Bad}_J) \leq \Pr_A(\textsc{Bad}_J^{(1)}) + \Pr_A(\textsc{Bad}_J^{(0)}) \leq 2\cdot m^{-0.1|J|},$$

where the last inequality follows from Equation 1.

Now, we put an x in $\textsc{UnBal}(A,B,I)$ if it holds that for all $J \subseteq I$, $x_J \in \textsc{Bad}_J$. The overall probability of $x \in \textsc{UnBal}(A,B,I)$ can be upper bounded as

$$p_A(\textsc{UnBal}(A,B,I)) \leq \sum_{J \subseteq I} \Pr_A(\textsc{Bad}_J) \leq 2 \sum_{r=1}^{|I|} \binom{|I|}{r} m^{-0.1r} \leq 2 \sum_{r=1}^{n} 2^{r \log n - 0.1 r \log m} \leq 3 \cdot 2^{-9 \log n}.$$

Fix an $x \notin \textsc{UnBal}(A,B,I)$. We have that $x_J \in \neg\textsc{Bad}_J$ for all J, which implies that

$$|\sum_{y_J} p_B(y_J) \chi(g^J(x_J, y_J))| \leq 2^{-5|J| \log n} \quad \text{for all } J.$$

Using Lemma 9 from [GPW17] (which appeared earlier in [GLM+15]), we conclude that

$$\Pr_{y_I \sim \frac{|B_{y_I}|}{|B|}} [g^I(x_I, y_I) = z_I] \in \frac{1}{2^{|I|}} \left[1 - n^{-3}, 1 + n^{-3}\right] \qquad \forall z \in \{0,1\}^{|I|}.$$

This establishes the desired property of $\textsc{UnBal}(A,B,I)$ as given in the statement of the lemma. This completes the proof. □

3 Proof of main result

We restate Theorem 1 and provide its proof below.

Theorem 8. *For all (partial) functions f, it holds that*

$$\mathrm{R}^{\mu}_{1/4}(f \circ \mathrm{Index}^n_m) = \Omega(\mathrm{R}_{1/3}(f) \cdot \log n),$$

where $m = \mathrm{poly}(n)$.

Proof. For a given function f, recall the definition of λ (hard distribution for f) and μ (lifted distribution for $f \circ \mathrm{Index}^n_m$) from Section 2. Let T be a deterministic communication tree for f achieving $\mathrm{D}^{\mu}_{1/4}(f \circ \mathrm{Index}^n_m)$. Let $c := \mathrm{D}^{\mu}_{1/4}(f \circ \mathrm{Index}^n_m)$ be the depth of T. Using our algorithm Π given in Algorithm 2 and described in the form of a flowchart in Figure 1, we get a randomized query protocol for f which makes an error of at most $\frac{1.1}{4}$ under λ (as implied by Lemma 9) and makes at most $O(c/\log n)$ expected number of queries (as implied by Lemma 15). This can be converted into an algorithm with $O(c/\log n)$ number of queries (in the worst case) and distributional error $\frac{1}{3}$, using standard application of Markov's inequality. This shows that

$$\mathrm{R}_{\frac{1}{3}}(f) = \mathrm{D}^{\lambda}_{\frac{1}{3}}(f) \leq O\left(\frac{c}{\log n}\right).$$

□

For an input z, we construct a tree $\mathcal{T}$ which represents the evolution of the algorithm Π, depending on the random choices made by it in steps 4, 9, 16, 19 and the Filter steps. All the nodes of the tree are labelled by unique triplets $(A \times B, I, v)$ where $I \subseteq [n]$ is the current interval, $A \subseteq [m]^n, B \subseteq \mathrm{Bal}^n_m$ are the current parts of the rectangle held by Alice and Bob respectively, and v is the current node of T. The root node is $([m]^n \times \mathrm{Bal}^n_m, [n], r)$ where r is the root of T, and the children of any node are all the nodes that can be reached from it depending on random choices made. Each edge is labelled by the conditional probability of the algorithm reaching the child node, conditioned on it reaching the parent node for that z. The overall probability of the algorithm reaching a node $(A \times B, I, v)$ on input z, denoted by $\Pr_{\mathcal{T},z}[(A \times B, I, v)]$ is obtained by multiplying all the conditional probabilities along the path from the root to $(A \times B, I, v)$.

Note that there are at most $O(n \log n)$ communication steps in T and additionally at most n query steps in Π, and constant number of operations for each of these steps in the algorithm.

Algorithm 2: Randomized query algorithm Π for f

Input: $z \in \{0,1\}^n$

Initialize v as root of the protocol tree T, $I = [n]$, $A = [m]^n$ and $B = \mathrm{Bal}_m^n$
while *v is not a leaf* **do**
  if *Bob sends a bit at v* **then**
    Pick $b \in \{0,1\}$ with probability $|B \cap Y^{v,b}|/|B|$
    if $|B \cap Y^{v,b}| < 2^{-n^2(2n-|I|)} \cdot |\mathrm{Bal}_m|^{|I|}$ *for the picked b* **then** Abort
    Set $v \leftarrow v_b$ and $B \leftarrow B \cap Y^{v,b}$
  end
  else if *Alice sends a bit at v* **then**
    For $b \in \{0,1\}$ pick $A' = A \cap X^{v,b}$ with probability $|A \cap X^{v,b}|/|A|$
    if $|A'| < \frac{1}{n^2}|A|$ *for the picked b* **then** Abort
    Filter$(A', \mathrm{Small}(A, A', I))$
    Set $v \leftarrow v_b$ and $A \leftarrow A'$
    if *there is an i such that* $|\mathrm{High}(A, m^{-93/100}, i)| > \frac{1}{n^3}|A|$ **then**
      Filter $(A, \mathrm{UnBal}(A, B, I) \cup \mathrm{Small}(A, A \setminus \mathrm{UnBal}(A, B, I), I))$
      while $|\mathrm{High}(A, m^{-93/100}, i)| > \frac{1}{n^3}|A|$ *for some $i \in I$* **do**
        Pick $A' = \mathrm{High}(A, m^{-93/100}, i)$ or $A \setminus \mathrm{High}(A, m^{-93/100}, i)$ with probability $|\mathrm{High}(A, m^{-93/100}, i)|/|A|$ or $1 - |\mathrm{High}(A, m^{-93/100}, i)|/|A|$ respectively
        if $A' = \mathrm{High}(A, m^{-93/100}, i)$ *is picked* **then**
          Query z_i
          Pick $y_i \in \{0,1\}^m$ with probability $|B_{y_i}|/|B|$
          if $|A'|_{U(y_i,z_i)}| < \frac{1}{n^3}|A'|$ *or* $|B_{y_i}| < 2^{-n^2(2n-|I|+1)} \cdot |\mathrm{Bal}_m|^{|I|-1}$ *for the picked y_i* **then** Abort
          Set $A \leftarrow A'|_{U(y_i,z_i)}$, $B \leftarrow B_{y_i}$ and $I \leftarrow I \setminus \{i\}$
        end
        Set $A \leftarrow A'$
      end
    end
    Filter $\big(A, \mathrm{High}(A, m^{-93/100}, I) \cup \mathrm{Small}(A, A \setminus \mathrm{High}(A, m^{-93/100}, I), I)\big)$
  end
end
Output as T does on the leaf v.

Procedure 3: Filter(A, S)

Input: $A \subseteq [m]^n$ and $S \subseteq A$

Pick $A' = S$ or $A \setminus S$ with probability $|S|/|A|$ or $1 - |S|/|A|$ respectively
if *$A' = S$ is picked* **then** Abort
Set $A \leftarrow A'$

Error analysis of algorithm Π

In this section we shall prove the following main lemma.

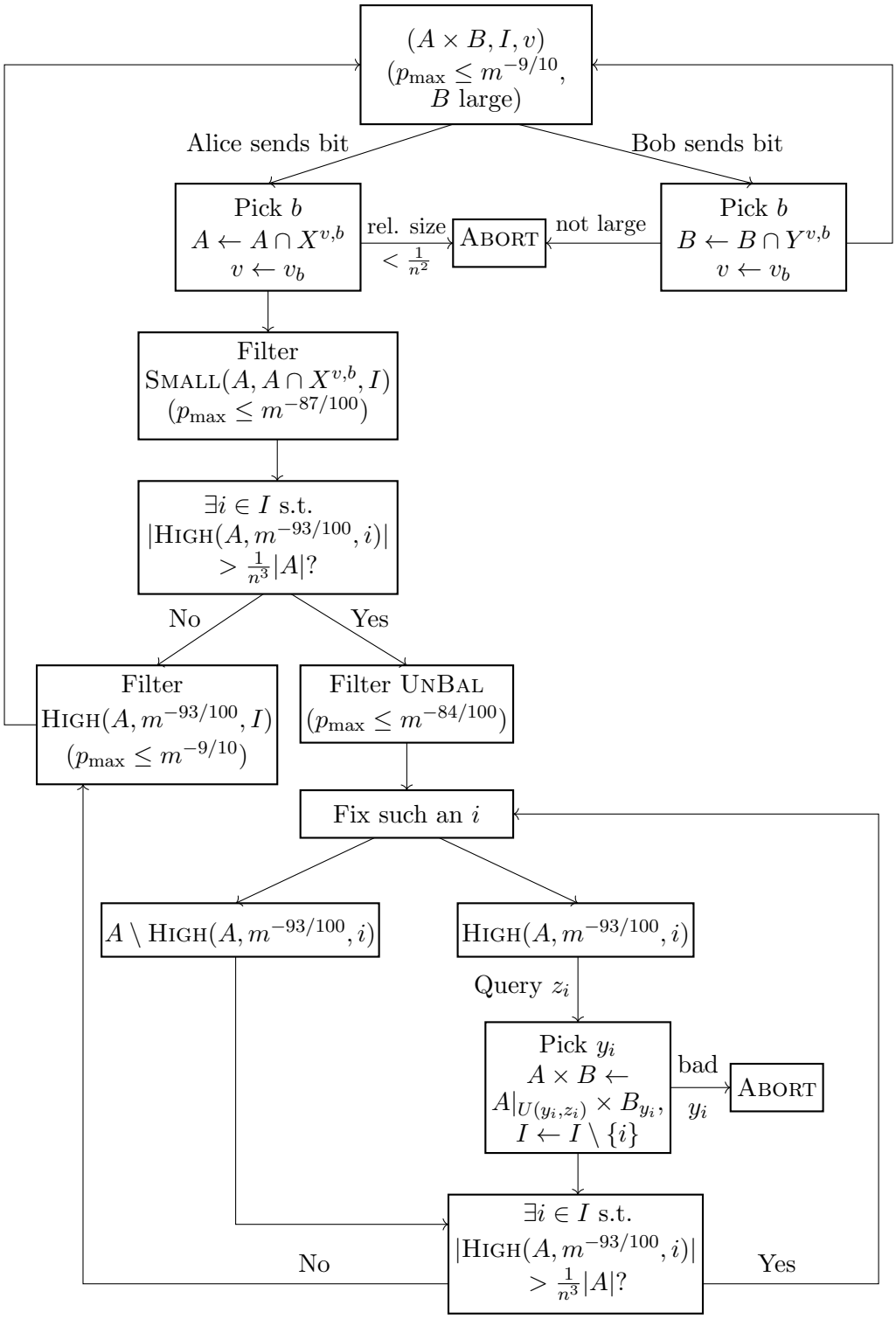

Figure 1: A flowchart description of the algorithm

Lemma 9. *The algorithm* Π *makes an error of at most* $1/4+O(\log n/n)$ *when input* z *is drawn from* λ.

The proof of this lemma has the following components. First we prove an Invariance Lemma which will show that the conditions required for the the Partition Lemma 5 hold. Second, using the Partition Lemma, we show a claim related to the size of $|(A\times B)\cap G(z)|$ relative to $|A\times B|$ at any node for any input z to Π. This lets us do two things: first we can show that the transition probabilities of Π are almost the same for any z, and second, we can show that the probability of the algorithm going to an aborted node is small. Finally, due to the transition probabilities being independent of z, the probability of Π reaching a leaf is close to the probability of the original protocol T reaching that leaf. This lets us show that the expected value of the error of Π over the leaves is close to the expected value of the error of T over the leaves, which is small.

Lemma 10 (Invariance Lemma)**.** *Throughout the execution of* Π*, we show the following invariants:*

1. $p_{\max}$ *bounds of* $m^{-87/100}$, $m^{-84/100}$ *and* $m^{-9/10}$ *for the current* A *with respect to the current interval* I *hold after steps 12, 14 and 26 respectively;*
2. B *is large with respect to the current interval* I.

Proof. 1. **After step 12:** If ABORT does not happen here, A is set to $A' = A^b\backslash A^b|_{\textsc{Small}(A,A^b,I)}$ (where we use A^b to denote $A\cap X^{v,b}$). A' only contains $x_{I\setminus\{i\}}\notin \textsc{Small}(A,A^b,I)$. For these prefixes, $|A'_{x_{I\setminus\{i\}}}|\geq \frac{1}{n^3}\cdot|A_{x_{I\setminus\{i\}}}|$ and obviously $|A'_{x_{I\setminus\{i\}}\circ x_i}|\leq|A_{x_{I\setminus\{i\}}\circ x_i}|$ for every x_i. Since a $p_{\max}$ bound of $m^{-9/10}$ holds for A, we have,

$$\max_{x_i}\frac{|A'_{x_{I\setminus\{i\}}\circ x_i}|}{|A'_{x_{I\setminus\{i\}}}|}=\max_{x_i}\frac{n^3\cdot|A_{x_{I\setminus\{i\}}\circ x_i}|}{|A_{x_{I\setminus\{i\}}}|}\leq m^{-87/100}.$$

Hence, the $p_{\max}$ bound of $m^{-87/100}$ holds for A'.

After step 14: Here an A for which a $p_{\max}$ bound of $m^{-87/100}$ holds is set to A'. We have for every remaining prefix $x_{I\setminus\{i\}}$, $|A'_{x_{I\setminus\{i\}}}|\geq\frac{1}{n^3}|A_{x_{I\setminus\{i\}}}|$. $|A'_{x_{I\setminus\{i\}}\circ x_i}|$ can only be at most $|A_{x_{I\setminus\{i\}}\circ x_i}|$. So for any $x_{I\setminus\{i\}}$,

$$\max_{x_i}\frac{|A'_{x_{I\setminus\{i\}}\circ x_i}|}{|A'_{x_{I\setminus\{i\}}}|}\leq\max_{x_i}\frac{n^3\cdot|A_{x_{I\setminus\{i\}}\circ x_i}|}{|A_{x_{I\setminus\{i\}}}|}\leq m^{-84/100}$$

due to the $p_{\max}$ bound on A.

After step 26: A similar argument holds here. Since the strings in both $\textsc{High}(A,m^{-93/100},I)$ and $\textsc{Small}(A,A\setminus\textsc{High}(A,m^{-93/100},I),I)$ are removed, we have for the remaining strings in the set A',

$$\max_{x_i}\frac{|A'_{x_{I\setminus\{i\}}\circ x_i}|}{|A'_{x_{I\setminus\{i\}}}|}\leq\max_{x_i}\frac{n^3\cdot|(A\setminus\textsc{High}(A,m^{-93/100},I))_{x_{I\setminus\{i\}}\circ x_i}|}{|A_{x_{I\setminus\{i\}}}|}\leq m^{-90/100}$$

by the definition of $\textsc{High}(A,m^{-93/100},I)$.

2. B is changed in steps 6 and 21. In step 21, I is changed to to $I\setminus\{i\}$ and B is correspondingly changed to B_{y_i}, so B always has a single string outside of the current interval I. The choice of how to update B is made in steps 4 and 19 and a choice that would lead to B not satifying the condition is aborted in steps 5 and 20.

□

Lemma 11. *Let $(A \times B, I, v)$ be a node of $\mathcal{T}$ at which a $p_{\max}$ bound of $m^{-8/10}$ holds for A with respect to I, and B is large with respect to I. Then for any $z \in \{0,1\}^n$ on which Π reaches $(A \times B, I, v)$, the number of inputs (x, y) in $A \times B$ consistent with z, denoted by $\rho_{(A\times B,I)}(z) = |(A \times B) \cap G^{-1}(z)|$, satisfies*

$$\rho_{(A\times B,I)}(z) \in \frac{1}{2^{|I|}} \left[1 - 4n^{-4}, 1 + 4n^{-4}\right] \cdot |A| \cdot |B|.$$

Proof. Without loss of generality, we assume $I = \{1, \ldots, l\}$, which means that the bits of z that have been queried till now are $l+1, \ldots n$. Since Π reaches $(A \times B, I, v)$ on z, we must have that $g^{n-l}(x_{[n]\setminus[l]}, y_{[n]\setminus[l]}) = z_{[n]\setminus[l]}$. Hence,

$$\begin{aligned}\rho_{(A\times B,I)}(z) &= \sum_{(x,y)\in A\times B} \mathbb{1}_{g(x_1,y_1)=z_1} \cdot \ldots \cdot \mathbb{1}_{g(x_l,y_l)=z_l} \\ &= \sum_{y\in B} \sum_{x_1 \in U(y_1,z_1)} \cdots \sum_{x_l \in U(y_l,z_l)} |A_{x_1 \ldots x_l}|.\end{aligned}$$

Now note that B contains only a single string outside of the interval I, so $|B| = |B_I|$ and B_I satisfies the condition required in the Partition Lemma 5. Additionally, since the $p_{\max}$ bound of $m^{-8/10}$ holds for A with respect to I, we can apply the Partition Lemma, to $A \times B$ on this interval. By the Lemma, we have that for at most $2 \cdot 2^{-n^2}$ fraction of $y_1 \in \textsc{Bad}(A, B, 1)$, and for all $y_1 \notin \textsc{Bad}(A, B, 1)$,

$$\begin{aligned}\sum_{x_1 \in U(y_1,z_1)} |A_{x_1 \ldots x_l}| &\in \left[\frac{1}{2} - n^{-5}, \frac{1}{2} + n^{-5}\right] |A_{x_2 \ldots x_l}| \\ &= \left[\frac{1}{2} - n^{-5}, \frac{1}{2} + n^{-5}\right] \sum_{x_1} |A_{x_1 x_2 \ldots x_l}| \quad \forall x_2 \ldots x_l \end{aligned} \tag{2}$$

and moreover B_{y_1} is large for $[l] \setminus \{1\}$. Moreover note that, for a fixed x_1, the $p_{\max}$ bound also holds for A_{x_1} with respect to $[l] \setminus \{1\}$, since it holds for A with respect to $[l]$. So taking $B' = \bigcup_{y_1 \notin \textsc{Bad}(A,B,1)} B_{y_1}$ we can apply the Partition Lemma again on $A_{x_1} \times B'$ for an index in $\{2, \ldots, l\}$. By the Partition Lemma, at most $2 \cdot 2^{-n^2}$ fraction of $y_2 \in \textsc{Bad}(A, B', 2)$, and for all $y_2 \notin \textsc{Bad}(A, B', 2)$,

$$\begin{aligned}\sum_{x_2 \in U(y_2,z_2)} |(A_{x_1})_{x_2 \ldots x_l}| &\in \left[\frac{1}{2} - n^{-5}, \frac{1}{2} + n^{-5}\right] |A_{x_1 x_3 \ldots x_l}| \\ &= \left[\frac{1}{2} - n^{-5}, \frac{1}{2} + n^{-5}\right] \sum_{x_2} |A_{x_1 x_2 \ldots x_l}| \quad \forall x_1 \, \forall x_3 \ldots x_l \end{aligned} \tag{3}$$

and B'_{y_2} is large for $[l] \setminus \{1, 2\}$. For $y_2 \notin \textsc{Bad}(A, B', 2)$, both conditions (2) and (3) hold, and for $y_2 \in \textsc{Bad}(A, B', 2)$,

$$\sum_{x_1 \in U(y_1,z_1)} \sum_{x_2 \in U(y_2,z_2)} |A_{x_1 \ldots x_l}| \leq \sum_{x_1} \sum_{x_2} |A_{x_1 \ldots x_l}|.$$

So similarly defining $B'' = \bigcup_{y_2 \notin \textsc{Bad}(A,B',2)} B'_{y_2}$ and combining conditions (2) and (3) we get, for

all $x_3 \dots x_l$,

$$\begin{aligned}\sum_{y\in B}\sum_{x_1\in U(y_1,z_1)}\sum_{x_2\in U(y_2,z_2)}|A_{x_1x_2\dots x_l}| &\geq \sum_{y\in B''}\sum_{x_1\in U(y_1,z_1)}\sum_{x_2\in U(y_2,z_2)}|A_{x_1x_2\dots x_l}| \\ &\geq (1-2\cdot 2^{-n^2})|B'|\sum_{x_1\in U(y_1,z_1)}\left(\frac{1}{2}-n^{-5}\right)\sum_{x_2}|A_{x_1\dots x_l}| \\ &= (1-2\cdot 2^{-n^2})\left(\frac{1}{2}-n^{-5}\right)|B'|\sum_{x_2}\sum_{x_1\in U(y_1,z_1)}|A_{x_1\dots x_l}| \\ &\geq (1-2\cdot 2^{-n^2})^2\left(\frac{1}{2}-n^{-5}\right)^2|B|\sum_{x_1}\sum_{x_2}|A_{x_1\dots x_l}|\end{aligned}$$

where in the last inequality we have used the fact that condition (2) holds for all $x_2 \dots x_l$ and hence all x_2. Similarly, for all $x_3 \dots x_l$,

$$\begin{aligned}\sum_{y\in B}\sum_{x_1\in U(y_1,z_1)}\sum_{x_2\in U(y_2,z_2)}|A_{x_1x_2\dots x_l}| \leq &(1-2\cdot 2^{-n^2})^2\left(\frac{1}{2}+n^{-5}\right)^2|B|\sum_{x_1}\sum_{x_2}|A_{x_1\dots x_l}| \\ &+4\cdot 2^{-n^2}|B|\sum_{x_1}\sum_{x_2}|A_{x_1\dots x_l}|\end{aligned}$$

Proceeding in this manner by successively applying the Partition Lemma on indices through to l and going to subsets which are not bad for these indices, we get the lower bound

$$\begin{aligned}\rho_{(A\times B,I)}(z) &\geq \left(1-2^{-n^2}\right)^l\left(\frac{1}{2}-n^{-5}\right)^l|B|\sum_{x_1}\dots\sum_{x_l}|A_{x_1\dots x_l}| \\ &\geq \frac{1}{2^l}\left(1-2^{-n^2}\right)^l(1-l\cdot n^{-5})|A|\cdot|B| \geq \frac{1}{2^l}(1-3n^{-4})|A|\cdot|B|\end{aligned}$$

and the upper bound

$$\rho_{(A\times B,I)}(z) \leq \frac{1}{2^l}(1+3n^{-4})|A|\cdot|B| + 2l\cdot 2^{-n^2}|A|\cdot|B| \leq \frac{1}{2^l}(1+4n^{-4})|A|\cdot|B|.$$

□

The lemma has the following immediate corollary.

Corollary 12. *For any node $(A_1\times B_1, I_1, v_1)$ and its successor node $(A_2\times B_2, I_2, v_2)$ such that a $p_{\max}$ bound of $m^{-8/10}$ holds at both these nodes and B_1 and B_2 are both large with respect to their respective intervals I_1 and I_2, the probability that the algorithm Π reaches $(A_2\times B_2, I_2, v_2)$, conditioned on it reaching $(A_1\times B_1, I_1, v_1)$ on any input z, lies in the range*

$$[1-8n^{-4}, 1+8n^{-4}]\cdot 2^{-|I_1\setminus I_2|}\cdot\frac{|A_2|\cdot|B_2|}{|A_1|\cdot|B_1|}.$$

Proof. Note that the transitions in $\mathcal{T}$ from $(A_1\times B_1, I_1, v_1)$ to $(A_1'\times B_1, I_1', v_1')$ and so on until to $(A_2\times B_2, I_2, v_2)$ happen according to the relative sizes of the rectangles. Hence the probability of these transitions on z are given by $\rho_{(A_1'\times B_1', I_1)}(z)/\rho_{(A_1\times B_1, I_1)}$ and so on until some $\rho_{(A_2\times B_2, I_2)}(z)/\rho_{(A_2'\times B_2', I_2')}(z)$. So,

$$\Pr_{\mathcal{T},z}[(A_2\times B_2, I_2, v_2)|(A_1\times B_1, I_1, v_1)] = \frac{\rho_{(A_1'\times B_1', I_1')}(z)}{\rho_{(A_1\times B_1, I_1)}(z)}\cdot\dots\cdot\frac{\rho_{(A_2\times B_2, I_2)}(z)}{\rho_{(A_2'\times B_2', I_2')}(z)} = \frac{\rho_{(A_2\times B_2, I_2)}(z)}{\rho_{(A_1\times B_1, I_1)}(z)}$$

Since the $p_{\max}$ bound of $m^{-8/10}$ holds at both these nodes and B_1 and B_2 are large, we can apply Lemma 11 on them to obtain that for every z,

$$\rho_{(A_1\times B_1,I_1)}(z) \in \frac{1}{2^{|I_1|}}[1-4n^{-4}, 1+4n^{-4}]\cdot|A_1|\cdot|B_1|$$

and a similar result holds for $(A_2 \times B_2, I_2, v_2)$. Plugging in the upper and lower bounds for both these quantities, we get the required result.

□

For bounding the probability of the algorithm aborting, we need the following claim.

Claim 13. *Consider a tree τ representing a random process with directed edges weighed by the conditional probability of going to a child node conditioned on being in a parent node. Some of the nodes are marked as aborted nodes, and we have that for any node, the sum of weights of the edges going to aborted children be at most δ. If the depth of τ is d, then the overall probability of the random process reaching an aborted node is at most $\delta \cdot d$.*

Proof. We construct a new tree τ' in which nodes which are not aborted at a particular level are coarse-grained into a single node and the aborted nodes are coarse grained into another node (which we again call abort node). For τ', the probability of a node having an aborted child is still at most δ and the overall probability of reaching an aborted node is at least as large as in τ. The probability of reaching an aborted node in τ' is given by

$$\delta + (1-\delta)\cdot\delta + (1-\delta)^2\cdot\delta \ldots + (1-\delta)^{d-1}\delta \leq d\delta$$

which gives us the required bound for the probability of reaching an aborted node in τ. □

Lemma 14. *The overall probability of the algorithm Π aborting on any input $z \in \{0,1\}^n$ is $O(\log n/n)$.*

Proof. The algorithm aborts on steps 5, 10, 20 and the FILTER steps. We shall consider each of these separately.

On steps 10 and 11: We first consider the conditional abort probability on a communication sub-routine of Alice starting from step 8 and ending at step 11, conditioned on being at a node $(A \times B, I)$ at the beginning of this subroutine at step 8. This gives us the conditional probability of not aborting at either step 10 or in the FILTER procedure in step 11. Note that a $p_{\max}$ bound of $m^{-9/10}$ holds at A and a $p_{\max}$ bound of $m^{-87/100}$ holds for all possible non-aborting A_j obtained from it at step 11. B is large at the beginning and does not change in these steps. Hence by Corollary 12,

$$\sum_{\text{non-aborting } j} \frac{\rho_{(A_j\times B,I)}(z)}{\rho_{(A\times B,I)}(z)} \geq (1-8n^{-4}) \sum_{\text{non-aborting } j} \frac{|A_j|}{|A|}.$$

Note that at first A is partitioned into two subsets A^0 and A^1 according to the picked b in step 9. At most one of A^0 and A^1 could have lead to an abort and because of our aborting condition we have

$$\sum_{\text{non-aborting } b\in\{0,1\}} \frac{|A^b|}{|A|} \geq 1 - \frac{1}{n^2}.$$

Moreover, from Lemma 4,

$$\sum_{x\in\textsc{Small}(A,A^b,I)} p_A(x) = \sum_{x\in\textsc{Small}(A,A^b,I)} \frac{|A^b_x|}{|A|} \leq \frac{1}{n^2}$$

which gives us

$$\sum_{\text{non-aborting } j} \frac{|A_j|}{|A|} = \sum_{\text{non-aborting } b\in\{0,1\}} \frac{|A^b \setminus \textsc{Small}(A, A^b, I)|}{|A|}$$
$$\geq \sum_{\text{non-aborting } b\in\{0,1\}} \frac{1}{|A|}\left(|A^b| - \sum_{x\in\textsc{Small}(A,A^b,I)} |A^b_x|\right)$$
$$\geq \left(1 - \frac{1}{n^2}\right) - \frac{2}{n^2} \geq 1 - \frac{3}{n^2}.$$

So finally we get,

$$\sum_{\text{non-aborting } j} \frac{\rho_{(A_j\times B,I)}(z)}{\rho_{(A\times B,I)}(z)} \geq (1 - 8n^{-4})(1 - 3n^{-2}) \geq 1 - 4n^{-2}.$$

Hence, the probability of abort conditioned on reaching this node is at most $4n^{-2}$.

On steps 14 and 26 (no queries): We can do very similar calculations for the probability of abort on steps 14, conditioned on the A after step 11. Note that a $p_{\max}$ bound better than $m^{-8/10}$ holds for both the parent node A in step 11 and the non-aborting child node A' in step 14. Hence $\rho_{(A'\times B,I)}(z)/\rho_{(A\times B,I)}(z) \geq (1 - 8n^{-4}) \cdot \frac{|A'|}{|A|}$ as before. Now in A' the strings $\textsc{UnBal}(A, B, I)$ and the strings $\textsc{Small}(A, A \setminus \textsc{UnBal}(A, B, I), I)$ are removed. By Lemma 7, the total probability loss due to removal of the strings in $\textsc{UnBal}(A, B, I)$ is n^{-50} and the total probability loss due to removal of the strings in $\textsc{Small}(A, A \setminus \textsc{UnBal}(A, B, I), I)$ is n^{-2} by Lemma 4. Hence the total conditional probability of aborting is again upper bounded by $O(n^{-2})$. A similar argument holds for the abort probability in step 26 if there are no queries carried out (we deal with the case where this abort happens after some queries are carried out in the next part).

Note that each of the aborts whose conditional probabilities we have calculated so far happen once after Alice communicates a bit. Since there are at most $O(n \log n)$ bits communicated, by Claim 13, the overall probability of abort in these steps is at most $O(\log n/n)$.

On steps 20 and 26 (at least one query): Now assuming at least one query happens, we calculate the probability of abort on steps 20 and 26, conditioned on being at a node $(A\times B, I, v)$ before the while loop began. A $p_{\max}$ bound of $m^{-84/100}$ holds for A and B is large with respect to I, which suppose is $\{1, \ldots, l\}$, by the Invariance Lemma. By Lemma 11 we can say,

$$\rho_{(A\times B,[l])}(z) \leq \frac{1 + 4n^{-4}}{2^l}|A| \cdot |B|$$

Consider the simplest case where the while loop has only one iteration, querying say z_1. In the while loop, first A is split into $A'(= \textsc{High}(A, m^{-93/100}, 1))$ and $A \setminus A'$. $A \setminus A'$ exits the while loop without any queries being done, and then a $\textsc{Filter}$ step is carried out on it, after which a $p_{\max}$ bound of $m^{-9/10}$ holds by the Invariance Lemma. Suppose the part that is not aborted in the $\textsc{Filter}$ step is $A \setminus A''$, then $|A \setminus A''| \geq (1 - 2n^{-2})|A \setminus A'|$, since at most n^{-2} fraction is removed in $\textsc{High}$ and $\textsc{Small}$ parts each. So again by Lemma 11 we have,

$$\rho_{((A\setminus A'')\times B,[l])}(z) \geq \frac{1 - 4n^{-4}}{2^l}|A \setminus A''| \cdot |B| \geq \frac{(1 - 4n^{-4})(1 - 2n^{-2})}{2^l}|A \setminus A'| \cdot |B|.$$

On A', z_1 is queried and A' is set to $A'|_{U(y_1,z_1)}$ depending on the choice of y_1 in step 19. Some of these y_1 lead to abort in step 20, let us denote this set by Ab. Note that $\text{Ab} = \text{Ab}_1 \cup \text{Ab}_2$, where Ab_1 is the set of y_1 for which B_{y_1} is not large with respect to $[l] \setminus \{1\}$, and Ab_2 is the set of y_1 for which $|A'|_{U(y_1,z_1)}| \leq \frac{1}{n^3}|A'|$. The non-aborting part then goes through another $\textsc{Filter}$

step, after which at most $2n^{-2}$ fraction of $A'|_{U(y_1,z_1)}$ is removed, and it has a $p_{\max}$ bound with respect to $[l] \setminus \{1\}$. So if we let $A_j \times B_j$ denote the rectangles on which a query happens and which are not aborted on steps 20 or 26, then by Lemma 11,

$$\begin{aligned}
&\sum_j \rho_{(A_j \times B_j, [l]\setminus\{1\})}(z) \\
&\geq \frac{(1-4n^{-4})(1-2n^{-2})}{2^{l-1}} \sum_{y_1 \notin \mathrm{Ab}} |A'|_{U(y_1,z_1)}| \cdot |B_{y_1}| \\
&= \frac{(1-4n^{-4})(1-2n^{-2})}{2^{l-1}} \left(\sum_{y_1} |A'|_{U(y_1,z_1)}| \cdot |B_{y_1}| - \sum_{y_1 \in \mathrm{Ab}_1} |A'|_{U(y_1,z_1)}| \cdot |B_{y_1}| \right. \\
&\quad \left. - \sum_{y_1 \in \mathrm{Ab}_2 \setminus \mathrm{Ab}_1} |A'|_{U(y_1,z_1)}| \cdot |B_{y_1}| \right)
\end{aligned}$$

We bound each of the summations in the above expression separately. For the first term, note that

$$\begin{aligned}
\sum_{y_1} |A'|_{U(y_1,z_1)}| \cdot |B_{y_1}| &= |B| \sum_{x \in A'} \sum_{y_1} p_B(y_1) \mathbb{1}_{g(x_1,y_1)=z_1} \\
&= |B| \sum_{x \in A'} \Pr_{y_1 \sim \frac{|B_{y_1}|}{|B|}} [g(x_1, y_1) = z_1] \\
&\geq |B| \sum_{x \in A'} \frac{1}{2}(1 - 3n^{-4}) = \frac{1}{2}(1-3n^{-4})|A'| \cdot |B| \qquad (4)
\end{aligned}$$

where the inequality holds due to Lemma 7, because A' does not contain any unbalanced strings. For the second and third terms we have,

$$\begin{aligned}
&\sum_{y_1 \in \mathrm{Ab}_1} |A'|_{U(y_1,z_1)}| \cdot |B_{y_1}| + \sum_{y_1 \in \mathrm{Ab}_2 \setminus \mathrm{Ab}_1} |A'|_{U(y_1,z_1)}| \cdot |B_{y_1}| \\
&\leq |A'| \cdot |B| \sum_{y_1 \in \mathrm{Ab}_1} p_B(y_1) + \frac{1}{n^3}|A'| \cdot |B| \sum_{y_1 \in \mathrm{Ab}_2 \setminus \mathrm{Ab}_1} p_B(y_1) \\
&\leq |A'| \cdot |B| \cdot 2^{-n^2} + |A'| \cdot |B| \cdot n^{-3} \leq |A'| \cdot |B| \cdot 2n^{-3}. \qquad (5)
\end{aligned}$$

This gives us

$$\sum_q \rho_{(A_j \times B_j, [l]\setminus\{1\})}(z) \geq \frac{(1-4n^{-4})(1-2n^{-2})}{2^l}(1-5n^{-3})|A'| \cdot |B|.$$

So the total probability of not aborting is at least

$$\begin{aligned}
&\frac{\sum_j \rho_{(A_j \times B_j, [l]\setminus\{1\})}(z) + \rho_{((A\setminus A'') \times B, [l])}(z)}{\rho_{(A \times B, [l])}(z)} \\
&\geq (1-2n^{-2}) \cdot \frac{1-4n^{-4}}{1+4n^{-4}} \left(\frac{|A'| + |A \setminus A'|}{|A|} - 5n^{-3} \cdot \frac{|A'|}{|A|} \right) \\
&\geq (1-2n^{-2})(1-6n^{-3}) \geq 1 - 4n^{-2}.
\end{aligned}$$

So the conditional probability of aborting in this step is at most $4n^{-2}$.

For a larger number of queries, there are more possible divisions of A, but the calculations are similar, applying different cases of Lemma 7. There are different sequences of queries different

partitions of the rectangle $A \times B$ at the beginning of the while loop. Let $A_i^s \times B_i^s$ denote all the sub-rectangles that were queried in sequence s and were not aborted in step 20 in any iteration. Let the unqueried set left after the query sequence s be J_s. Step 26 takes A_i^s to $A_i'^s$, removing at most $2n^{-2}$ fraction of the strings for each subrectangle, so we have for all non-aborting subrectangles after step 26,

$$\sum_s \sum_{A_i'^s \times B_i'^s} \rho_{(A_i'^s \times B_i'^s, J_s)}(z) \geq (1-4n^{-4})(1-2n^{-2}) \sum_s \sum_{A_i^s \times B_i^s} \frac{1}{2^{|J_s|}} |A_i^s| \cdot |B_i^s|. \tag{6}$$

Now each $A_i^s \times B_i^s$ is formed successively projecting onto the HIGH or $A \setminus$ HIGH part for each index that is queried and then projecting Alice's part to $U(y_i, z_i)$ and Bob's part to the y_i picked in step 19. However, note that all subrectangles that are queried in a particular sequence have the same sequence of falling into the HIGH or $A \setminus$ HIGH parts. Once we know what the HIGH and $A \setminus$ HIGH choices are for each sequence s, we may rearrange the branching process so that the projections to HIGH or $A \setminus$ HIGH are done first and then the projections to $U(y_i, z_i)$ and y_i are done. Suppose for the query sequence s, A^{*s} is the subset of A (Alice's part of the rectangle before the query sequence began) obtained by doing the projections to HIGH or $A \setminus$ HIGH corresponding to s first. Now fix an $s = (i_1, \ldots, i_k)$. Similar to the calculation in (4), by applying Lemma 7 with $J = I \setminus J_s$ we can say that

$$\sum_{y_{I \setminus J_s}} |A^{*s}|_{U(y_{i_1}, z_{i_1})} \cdots |_{U(y_{i_k}, z_{i_k})}| \cdot |B_{y_{I \setminus J_s}}| \geq \frac{1-3n^{-4}}{2^{|I \setminus J_s|}} |A^{*s}| \cdot |B|.$$

And similar to the calculation in (5), we can say that the choices of of y_i that lead to abort somewhere in the sequence satisfy

$$\sum_{y_{I \setminus J_s} \in \mathrm{Ab}} |A^{*s}|_{U(y_{i_1}, z_{i_1})} \cdots |_{U(y_{i_k}, z_{i_k})}| \cdot |B_{y_{I \setminus J_s}}| \leq \frac{|I|(1+3n^{-4})}{2^{|I \setminus J_s|-2} \cdot n^3} |A^{*s}| \cdot |B|.$$

This gives us

$$\sum_{A_i^s \times B_i^s} |A_i^s| \cdot |B_i^s| \geq \frac{1-3n^{-4}}{2^{|I \setminus J_s|}} \left(1 - \frac{4(1+3n^{-4})}{n^2(1-3n^{-4})}\right) |A^{*s}| \cdot |B| \geq \frac{1-6n^{-2}}{2^{|I \setminus J_s|}} |A^{*s}| \cdot |B|.$$

Plugging this into (6) we get,

$$\sum_s \sum_{A_i'^s \times B_i'^s} \rho_{(A_i'^s \times B_i'^s, J_s)}(z) \geq \frac{(1-4n^{-4})(1-2n^{-2})(1-6n^{-2})}{2^{|I|}} \sum_s |A^{*s}| \cdot |B| \geq \frac{1-20n^{-2}}{2^{|I|}} |A| \cdot |B|.$$

Now using upper bound $\rho_{(A \times B, I)}(z)$, we get the abort probability to be $O(n^{-2})$ for the case of multiple iterations of the while loop as well. Since there can be at most n queries, the total probability of aborting is at most $O(n^{-1})$.

On step 5: Since the abort condition here is with respect to the whole of $\mathrm{Bal}_m^{|I|}$ rather than the parent, we calculate the abort probability conditioned on the root. Conditioned on the root, by a similar argument as above, the probability of one such abort happening on $(A \times B, I)$ is at most $O(n^{-4}) \frac{|B|}{|\mathrm{Bal}_m|^n}$. Now note that everytime we change I, B is restricted to having only one string outside of I. So for the aborting sets,

$$\frac{|B|}{|\mathrm{Bal}_m|^n} = \frac{|B_I|}{|\mathrm{Bal}_m|^n} \leq \frac{|B_I|}{|\mathrm{Bal}_m|^{|I|}} \leq 2^{-n^2}.$$

Now the depth of $\mathcal{T}$ is at most $O(n \log n)$, resulting in at most $2^{O(n \log n)}$ subsets of Bal_m^n which can abort. Thus the overall probability of abort of this form is at most $O(n^{-4}) \cdot 2^{O(n \log n)} \cdot 2^{-n^2} = O(2^{-n})$.

This gives the total probability of the algorithm aborting at any step to be at most $O(\log n/n)$. □

Now we are in a position to do error analysis for the algorithm.

Proof of Lemma 9. The probability that Π makes an error is at most the sum of the probability that Π aborts, given by Lemma 14, and the probability that it makes an error on a leaf. We know by Lemma 14 that the overall probability of abort on any z is at most $O(\log n/n)$, hence the overall probability of abort when z is drawn from λ to also be at most $O(\log n/n)$.

To bound the error at a leaf, let us denote the ouput of a leaf L of T by b^L and probability that T on input (x, y) drawn uniformly from $G^{-1}(z)$ for a fixed z, reaches leaf L by q_z^L. By correctness of T on the distribution μ we have,

$$\Pr_{(x,y)\sim\mu}[T(x,y) = f \circ g^n(x,y)] = \mathbb{E}_{z\sim\lambda}\left[\sum_{L:f(z)=b^L} q_z^L\right] \geq \frac{3}{4}. \tag{7}$$

Let us further denote the probability of Π reaching a leaf on a fixed input z by $q_z'^L$. We will lower bound

$$\Pr_{z\sim\lambda}[\Pi(z) = f(z)] = \mathbb{E}_{z\sim\lambda}\left[\sum_{L:f(z)=b^L} q_z'^L\right].$$

Due to (7), it is enough to show that q_z^L and $q_z'^L$ are close. Let the rectangle associated with the leaf L of T be denoted as $A^L \times B^L$. Since T has no internal randomness, the probability when an input drawn uniformly from $G^{-1}(z)$ reaches L is given only by the relative number of $(x, y) \in A^L \times B^L$ that are consistent with $G^{-1}(z)$. That is,

$$q_z^L = \frac{|(A^L \times B^L) \cap G^{-1}(z)|}{|G^{-1}(z)|}.$$

Now there are many nodes $(A_i \times B_i, I_i, L)$ in $\mathcal{T}$ corresponding to the node L and we have that $A^L = (\cup_i A_i) \cup A^L_{\text{ABORT}}$, $B^L = (\cup_i B_i) \cup B^L_{\text{ABORT}}$. Moreover, we know from Corollary 12 that the probability of Π going to node $(A_i \times B_i, I_i, L)$ is proportional to $|(A_i \times B_i) \cap G^{-1}(z)|$, which is independent of z (the non-aborting leaf nodes all have a $p_{\max}$ bound and large B). So,

$$\begin{aligned} q_z'^L &= \frac{\sum_{(A_i\times B_i, I_i, L)\in\mathcal{T}} |(A_i \times B_i) \cap G^{-1}(z)|}{|G^{-1}(z)|} \\ &= \frac{|(A^L \times B^L) \cap G^{-1}(z)| - |(A^L_{\text{ABORT}} \times B^L_{\text{ABORT}}) \cap G^{-1}(z)|}{|G^{-1}(z)|} \\ &\geq q_z^L - O\left(\frac{\log n}{n}\right) \end{aligned}$$

where we have used the abort probability calculation to upper bound $|(A^L_{\text{ABORT}} \times B^L_{\text{ABORT}}) \cap G^{-1}(z)|/|G^{-1}(z)|$. This gives us the probability of the algorithm making an error on a leaf to be

$$1 - \mathbb{E}_{z\sim\lambda}\left[\sum_{L:f(z)=b^L} q_z'^L\right] \leq 1 - \mathbb{E}_{z\sim\lambda}\left[\sum_{L:f(z)=b^L} q_z^L\right] + O\left(\frac{\log n}{n}\right) \leq \frac{1}{4} + O\left(\frac{\log n}{n}\right).$$

□

Expected number of queries of Π

Lemma 15. *The algorithm Π makes at most $\frac{2c}{\log n}$ expected number of queries, where c is the number of bits communicated in T in the worst case.*

Proof. Consider a leaf node L of T, whose corresponding rectangle in T is $A^L \times B^L$. As noted before, $A^L = (\cup_i A_i) \cup A^L_{\text{ABORT}}$, where $(A_i \times B_i, I_i, L)$ are different nodes corresponding to leaf L in $\mathcal{T}$. Note that at either a communication or query step of Π, at most $O\left(\frac{1}{n^2}\right)$ fraction of $|A|$ is aborted, and since there are $O(n \log n)$ communication steps and $O(n)$ query steps, we can at most have $|A^L_{\text{ABORT}}| \leq O\left(\frac{\log n}{n}\right)|A^L|$. The different A_i correspond to different choices made by the algorithm in steps 16 and 19. For each query performed, step 16 reduces the size of Alice's half of the rectangle by at most n^3 (by the condition on which queries are carried out) and step 19 reduces the size by at most n^3 (otherwise we abort on step 20). Since $(n - |I_i|)$ queries are performed on $(A_i \times B_i, I_i, L)$, we must have,

$$|A_i| \geq n^{6(|I_i|-n)}|A^L \setminus A^L_{\text{ABORT}}| \geq n^{6(|I_i|-n)}\left(1 - O\left(\frac{\log n}{n}\right)\right)|A^L| \geq \frac{1}{2} \cdot n^{6(|I_i|-n)} \cdot |A^L|. \quad (8)$$

Now consider a string $x \in A_i$ and suppose for convenience $I_i = \{1, \ldots, l\}$ without loss of generality. We have that,

$$\frac{1}{|A_i|} = p_{A_i}(x) = p_{A_i}(x_{[l]}) \cdot p_{A_i}(x_{l+1}|x_{[l]}) \cdot \ldots \cdot p_{A_i}(x_n|x_{[l]} \circ x_{l+1} \circ \ldots \circ x_{n-1}).$$

Note that there is at least one $x_{[l]}$ in A_i such that $p_{A_i}(x_{[l]}) \geq m^{-l}$. Without loss of generality, let us assume the queries happened in the order $z_n, z_{n-1}, \ldots, z_{l+1}$. Suppose the query z_{l+1} happened on an ancestor A'_i of A_i. In order for the query to have happened, for any x_{l+1} that is retained in A_i we must have had $p_{A'_i}(x_{l+1}|x_{[l]}) > m^{-93/100}$. Now A_i is obtained by projecting $\textsc{High}(A'_i, m^{-93/100}, l+1)$ to $U(y_{l+1}, z_{l+1})$ for some y_{l+1} and then filtering out some $x_{[l]}$. This means that for all $x_{[l]} \circ x_{l+1}$ present in A_i, we must have $|(A_i)_{x_{[l]} \circ x_{l+1}}| = |(A'_i)_{x_{[l]} \circ x_{l+1}}|$ and obviously $|(A_i)_{x_{[l]}}| \leq |(A'_i)_{x_{[l]}}|$ for all $x_{[l]}$. So we have,

$$p_{A_i}(x_{l+1}|x_{[l]}) = \frac{|(A_i)_{x_{[l]} \circ x_{l+1}}|}{|(A_i)_{x_{[l]}}|} \geq \frac{|(A'_i)_{x_{[l]} \circ x_{l+1}}|}{|(A'_i)_{x_I}|} \geq m^{-93/100}.$$

Similarly we get $p_{A_i}(x_{l+2}|x_{[l]} \circ x_{l+1}) \geq m^{-93/100}$ and so on. This gives us, for at least one $x_{[l]}$,

$$\frac{1}{|A_i|} \geq m^{-|I_i|} \cdot \left(m^{93/100}\right)^{|I_i|-n} = n^{-7|I_i|} \cdot m^{-93n/100} \geq m^{-n} \cdot n^{7(n-|I_i|)}. \quad (9)$$

So from equations (8) and (9) we get for any $(A_i \times B_i, I_i, L)$,

$$m^n \cdot n^{7(|I_i|-n)} \geq \frac{1}{2} \cdot n^{6(|I_i|-n)}|A^L|.$$

Rearranging and taking logarithm on both sides we get,

$$\log\left(\frac{m^n}{|A^L|}\right) \geq (n - |I_i|)\log n - 1 \qquad \Rightarrow \quad n - |I_i| \leq \frac{2}{\log n} \cdot \log\left(\frac{m^n}{|A^L|}\right)$$

Now $(n - |I_i|)$ is the number of queries done on this branch of $\mathcal{T}$ and $\log\left(\frac{m^n}{|A^L|}\right)$ is the number of bits communicated by Alice on this branch of $\mathcal{T}$, which is at least the total number of bits communicated at leaf L of the communication protocol. Hence the expected number of queries made is at most $2/\log n$ times the expected number of bits communicated, which is at most the number of bits communicated in the worst case. □

Acknowledgement

This work is supported by the Singapore Ministry of Education and the National Research Foundation, also through the Tier 3 Grant Random numbers from quantum processes MOE2012-T3-1-009.

References

[ABBD+16] A. Anshu, A. Belovs, S. Ben-David, M. Gs, R. Jain, R. Kothari, T. Lee, and M. Santha. Separations in communication complexity using cheat sheets and information complexity. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 555–564, Oct 2016.

[GLM+15] Mika Göös, Shachar Lovett, Raghu Meka, Thomas Watson, and David Zuckerman. Rectangles are nonnegative juntas. In *Proceedings of the Forty-seventh Annual ACM Symposium on Theory of Computing*, STOC '15, pages 257–266, New York, NY, USA, 2015. ACM.

[GPW15] Mika Göös, Toniann Pitassi, and Thomas Watson. Deterministic communication vs. partition number. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 1077–1088, Oct 2015.

[GPW17] Mika Göös, Toniann Pitassi, and Thomas Watson. Query-to-communication lifting for bpp. 2017.

[Raz99] Ran Raz. Exponential separation of quantum and classical communication complexity. In *Proceedings of the Thirty-first Annual ACM Symposium on Theory of Computing*, STOC '99, pages 358–367, New York, NY, USA, 1999. ACM.

[RM99] Ran Raz and Pierre McKenzie. Separation of the monotone nc hierarchy. *Combinatorica*, 19(3):403–435, 1999.

[She11] Alexander A. Sherstov. The pattern matrix method. *SIAM Journal on Computing*, 40(6):1969–2000, 2011.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il