# Explicit separations between deterministic and randomized number-on-forehead communication

Talk by Kai Zheng

Notes by Sanjana Das

December 1, 2023

This is based on a recent paper by Kelley, Lovett, and Meka.

## §1 Introduction

### §1.1 The number-on-forehead model

Number-on-forehead (NOF) communication is a fairly well-studied communication model in theoretical computer science. We'll define a three-player number-on-forehead model, but the same setup can be used for more players as well.

Suppose we have three players, who we'll call $A$, $B$, and $C$. Each of these players has an input; we'll call their inputs $x$, $y$, and $z$ (respectively), and we'll suppose they're drawn from $[n]$. And there's a function $f\colon [n]^3 \to \{0,1\}$ that they wish to compute on their inputs — i.e., their goal is to calculate $f(x,y,z)$. But there's a twist — each player can only see the *other* two players' inputs.

> **Remark 1.1.** You can imagine that the three players are all in a room, and they each have their own input on their forehead — and they can see the other two players' foreheads, but not their own.

To compute $f(x,y,z)$, the players are allowed to perform some communication — they have a fixed communication protocol, and at the end of this protocol they should all be able to compute $f(x,y,z)$. (We can think of this communication as the players writing messages on a shared blackboard.) And we're interested in the minimum amount of communication this takes.

> **Question 1.2.** Given a function $f$, how many bits of communication are needed to compute $f$?

### §1.2 Deterministic vs. randomized protocols

There's a few different kinds of communication protocols we can work with — in particular, we can consider both *deterministic* and *randomized* protocols.

In a *deterministic* protocol, all the messages are fixed given the inputs, and the players must always succeed — i.e., they must always output the correct value of $f(x,y,z)$.

On the other hand, in a *randomized* protocol, the players' messages don't have to be fixed given their inputs — in this setting, the players have a shared source of randomness, and their communication can depend on this randomness. And instead of having to always succeed, we'll say that the players have to succeed with probabiltiy $\frac{2}{3}$, where the probability is over the shared randomness — so for all inputs, the players output the correct value of $f(x,y,z)$ with probability at least $\frac{2}{3}$ over the shared randomness.

**Remark 1.3.** Here we're using *public* randomness (where there's a shared random string everyone has access to). Private randomness is at most as powerful as public randomness (since the players could agree to split up the shared string into their own pieces), but it's actually *strictly* less powerful. (It might even be more similar to the deterministic model.)

**Definition 1.4.** We define the deterministic communication complexity of $f$, denoted $\mathrm{Det}^{\mathsf{cc}}(f)$, as the minimum number of bits in a deterministic protocol computing $f$. Similarly, we define the randomized communication complexity of $f$, denoted $\mathrm{Rand}^{\mathsf{cc}}(f)$, as the minimum number of bits in a randomized protocol computing $f$.

## §1.3 Some examples

We'll first see an example that only has two players (where there are two inputs $x$ and $y$, with one player seeing $x$ and the other seeing $y$, and the players' goal is to compute $f(x, y)$).

**Example 1.5**

Let $f(x, y) = \mathbf{1}[x = y]$ be the two-input equality function.

For a deterministic protocol, it turns out that the best thing we can do is just to have one player send over the entire input they see (i.e., the first player looks at $x$ and sends it over to the second, who checks whether it's equal to $y$ and announces the answer); so we have

$$\mathrm{Det}^{\mathsf{cc}}(f) = \log n.$$

On the other hand, the randomized communication complexity is a lot smaller.

**Claim 1.6** — We have $\mathrm{Rand}^{\mathsf{cc}}(f) = O(1)$.

*Proof.* We'll describe a randomized protocol with only a constant amount of communication. First, both players treat the numbers $x$ and $y$ that they see as vectors in $\{0, 1\}^{\log n}$ (by taking their binary expansions). Then they jointly choose a random vector $v \in \{0, 1\}^{\log n}$, and they write down the single bits $\langle x, v\rangle$ and $\langle y, v\rangle$ (respectively) — the point is that if they write down *different* bits, then they automatically know their inputs are *not* equal, and so they can output $f(x, y) = 0$. We can repeat this a few times to boost the success probability over $\frac{2}{3}$ — so if $\langle x, v\rangle = \langle y, v\rangle$ on all the repetitions, then the players output $f(x, y) = 1$, while if this fails on any trial, then they output $f(x, y) = 0$. □

**Remark 1.7.** It might feel somewhat troubling that even though we're only communicating a constant number of bits here, we're actually using $\log n$ bits of randomness — does it really make sense to think of this as way more efficient than actually communicating $\log n$ bits? But one way to think about this (so that this does make sense) is that in the randomized protocol, the players aren't *telling* a lot to each other, which means each player only has to reveal a constant amount of information about the input they see to the other player.

We've looked at the equality function over two players; what about the equality function over three players?

**Example 1.8**

Let $f(x, y, z) = \mathbf{1}[x = y = z]$ be the three-input equality function.

Now we actually have $\mathrm{Det}^{\mathsf{cc}}(f) = O(1)$ as well — each player can just look at the other two players' inputs and say whether they're equal (e.g., player $A$ outputs whether $y = z$), and once all three players have done so, they'll know whether all three inputs are equal. And of course we always have $\mathrm{Rand}^{\mathsf{cc}}(f) \leq \mathrm{Det}^{\mathsf{cc}}(f)$ (any deterministic protocol can also be viewed as a randomized one), so $\mathrm{Rand}^{\mathsf{cc}}(f) = O(1)$ as well.

## §1.4 Separations

**Question 1.9.** How big can the gap between $\mathrm{Det}^{\mathsf{cc}}(f)$ and $\mathrm{Rand}^{\mathsf{cc}}(f)$ be?

Of course, we always have $\mathrm{Det}^{\mathsf{cc}}(f) \geq \mathrm{Rand}^{\mathsf{cc}}(f)$, since we can get a randomized protocol for $f$ by just ignoring the randomness and running our deterministic protocol. But we're interested in how *much* bigger $\mathrm{Det}^{\mathsf{cc}}(f)$ can be.

With two players, we've already seen that the two-input equality function (as in Example 1.5) has a gap of $\Omega(\log n)$. And this is the largest possible gap, since we trivially have $\mathrm{Det}^{\mathsf{cc}}(f) \leq \log n$ (each players can always communicate the entire input they see) and $\mathrm{Rand}^{\mathsf{cc}}(f) \geq 1$. So over two players, we've already satisfactorily answered this question — there's a simple function $f$ achieving the biggest possible gap.

But over three players, things are very different. We do know there's still a gap of $\Omega(\log n)$ — i.e., there exist functions with $\mathrm{Det}^{\mathsf{cc}}(f) = \Omega(\log n)$ and $\mathrm{Rand}^{\mathsf{cc}}(f) = O(1)$. But we don't know any actual examples of such functions — we know such functions *exist*, but we don't know explicitly what they are. (The proof that such functions exist is via a counting argument, due to Beame; so it doesn't give us actual constructions.)

The main theorem of the Kelley–Lovett–Meka paper gives an explicit function that achieves a fairly large separation.

**Theorem 1.10** (Kelley–Lovett–Meka)
Let $q$ be a prime, fix $k \geq 34$, and identify $[n]$ with $\mathbb{F}_q^k$ (so we view $x$, $y$, and $z$ as elements of $\mathbb{F}_q^k$). Let

$$f(x, y, z) = \mathbf{1}[\langle x, y\rangle = \langle y, z\rangle = \langle x, z\rangle].$$

Then we have $\mathrm{Det}^{\mathsf{cc}}(f) = \Omega((\log n)^{1/3})$ and $\mathrm{Rand}^{\mathsf{cc}}(f) = O(1)$.

(We think of $k$ as fixed and $q$ as large, chosen such that $q^k \approx n$. We certainly need $q$ to grow, because if $q$ is fixed then even $\mathrm{Det}^{\mathsf{cc}}(f)$ would be $O(1)$ — we can write down each inner product using $\log q$ bits.)

**Remark 1.11.** For this function, do we think that $(\log n)^{1/3}$ is the right deterministic complexity, or do we think the truth is larger? We think that the true deterministic complexity is probably $\log n$ — there's no obvious protocol that does better.

In fact, the authors more generally show that any 'random-looking' function (in a certain sense) achieves the bound $\mathrm{Det}^{\mathsf{cc}}(f) = \Omega((\log n)^{1/3})$, and it's plausible that these random-looking functions might actually have $\mathrm{Det}^{\mathsf{cc}}(f) = \Omega(\log n)$.

The proof that $\mathrm{Rand}^{\mathsf{cc}}(f) = O(1)$ is not too hard.

*Proof of randomized complexity.* We can use the following randomized protocol for $f$. First, each player can compute one of these inner products on their own — $A$ knows $\langle y, z\rangle$, $B$ knows $\langle x, z\rangle$, and $C$ knows $\langle x, y\rangle$. Then they can split into pairs, and each pair runs the randomized 2-player equality protocol (from Claim 1.6) on the inner products they have (e.g., $A$ and $B$ use the protocol to figure out whether $\langle y, z\rangle = \langle x, z\rangle$). Finally, the players all output the results of these protocols, which tell us whether $\langle x, y\rangle = \langle y, z\rangle = \langle x, z\rangle$. □

Lower-bounding the deterministic complexity is much more difficult (this is generally the case in these sorts of problems). To do so, the authors define some new notions of pseudorandomness and show any function satisfying these notions has high deterministic complexity; and then they show this function $f$ does satisfy these notions. Today we'll focus on the parts of the proof where they define these notions and use them to get high deterministic complexity; so we're not actually going to work with this specific function $f$.

## §2 Ideas behind the proof

### §2.1 A method for bounding deterministic complexity

To motivate what comes next, we'll first talk about a general method for lower-bounding deterministic communication complexity (and we'll see a combinatorial way of looking at it).

**Definition 2.1.** We say a set $T \subseteq [n]^3$ is a cylinder intersection if we can write it as

$$T = \{(x, y, z) \mid (x, y) \in S_1,\ (x, z) \in S_2,\ (y, z) \in S_3\}$$

for some sets $S_1, S_2, S_3 \subseteq [n]^3$.

Here we think of the sets $\{(x, y, z) \mid (x, y) \in S_1\}$, $\{(x, y, z) \mid (x, z) \in S_2\}$, and $\{(x, y, z) \mid (y, z) \in S_3\}$ as *cylinders* — the point is that for each of these sets, there's one input that doesn't affect membership in the set (e.g., membership in the first set doesn't depend on $z$).

**Example 2.2**

- The set $\{(x, x, x)\}$ is a cylinder intersection (as it's the intersection of the three cylinders defined by $x = y$, $y = z$, and $x = z$).
- The set $\{(x, y, z) \mid x + y = z\}$ is not a cylinder intersection — this is because all pairs $(x, y)$ with $x + y \leq n$ appear in this set, as do all pairs $(x, z)$ with $x \leq z$ and all pairs $(y, z)$ with $y \leq z$, so $S_1$, $S_2$, and $S_3$ must then contain all such pairs (respectively); but most ways of combining such pairs don't satisfy $x + y = z$.

The reason cylinder intersections are useful is because of the following result (which we won't prove).

**Theorem 2.3**

If a function $f$ has $\mathrm{Det}^{\mathsf{cc}}(f) = b$, then there are $2^b$ cylinder intersections $T_1, \ldots, T_{2^b}$ such that

$$\{(x, y, z) \mid f(x, y, z) = 1\} = T_1 \cup \cdots \cup T_{2^b}.$$

(The cylinder intersections $T_i$ don't have to be disjoint — they're allowed to overlap.)

### §2.2 A high-level overview of the proof

Theorem 2.3 means that one way to prove a lower bound on deterministic communication complexity is to show that the set $D = \{(x, y, z) \mid f(x, y, z) = 1\}$ *can't* be expressed as a union of a small number of cylinder intersections. In particular, in order to get an explicit function $f$ with $\mathrm{Det}^{\mathsf{cc}}(f) = \Omega((\log n)^{1/3})$, we want to construct a set $D \subseteq [n]^3$ such that $D$ cannot be written as $T_1 \cup \cdots \cup T_{2^b}$ for $b = (\log n)^{1/3}$. If we can construct such a set, then we can take $f = \mathbf{1}[D]$, and since $D$ can't be written as a union of a small number of cylinder intersections, Theorem 2.3 will give the desired lower bound on its deterministic communication complexity. (Theorem 1.10 already specifies $f$ (and therefore $D$), but imagine for now that we don't yet know this, and we're just trying to construct *some* nice function with high deterministic complexity.)

At a high level, the proof works by imposing some nice properties on $D$, and showing that if $D$ satisfies these properties, then any cylinder intersection with low density *also* has low density *inside* $D$. More precisely, letting $F$ be the indicator function of our cylinder intersection, we show that if $\mathbb{E}_{(x,y,z)\in[n]^3}[F(x,y,z)]$ — the density of the cylinder intersection in the *entire* space — is small, then $\mathbb{E}_{(x,y,z)\in D}[F(x,y,z)]$ — the density of the cylinder intersection in $D$ — is small as well.

Then we construct $D$ to be a *sparse* (i.e., low-density) set satisfying these nice properties, and suppose that we can write $D = T_1 \cup \cdots \cup T_{2^b}$ as a union of cylinder intersections. Then since $D$ itself has low density (in the entire space), so does each cylinder intersection $T_i$. But this means the density of each $T_i$ inside $D$ is also small; and this means we'll need a lot of these cylinder intersections to cover $D$.

(The set $D$ that the authors construct is the one in Theorem 1.10 — i.e., $\{(x,y,z) \mid \langle x,y\rangle = \langle y,z\rangle = \langle x,z\rangle\}$ — and they show that it satisfies these nice properties. We won't discuss this part of the proof, though — we'll just focus on what the nice properties are and how they imply this statement about density.)

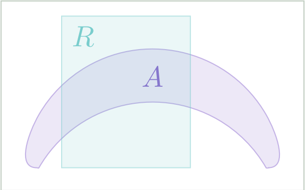
## §2.3 Some notions of pseudorandomness

To capture the properties of $D$ that we want, we'll define some notions of pseudorandomness for sets (or more generally functions).

**Definition 2.4** (Spreadness)**.** Let $A\colon X \times Y \to [0,1]$ be a function, and let $r \geq 1$ and $\varepsilon \in (0,1)$. Then we say $A$ is $(r,\varepsilon)$-spread if for any rectangle $R = X' \times Y' \subseteq X \times Y$ of size $|R| \geq 2^{-r}\,|X \times Y|$, we have

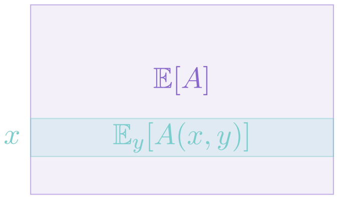
$$\mathbb{E}_{(x,y)\in R}[A(x,y)] \leq (1+\varepsilon)\mathbb{E}[A].$$

We use $\mathbb{E}[A]$ to mean $\mathbb{E}_{(x,y)\in X\times Y}[A(x,y)]$. This condition says that if we look at the density of $A$ restricted to any reasonably large rectangle $R$, it's not too much bigger than the original density of $A$ (where $r$ represents what we mean by 'reasonably large,' and $\varepsilon$ represents what we mean by 'not too much bigger'). (This is a *one*-sided pseudorandomness condition; this property is also sometimes called *upper regularity.*)



**Definition 2.5** (Left-lower boundedness)**.** For $\varepsilon \in (0,1)$, we say a function $A\colon X \times Y \to [0,1]$ is $\varepsilon$-left-lower bounded if for every $x \in X$, we have $\mathbb{E}_{y\in Y}[A(x,y)] \geq (1-\varepsilon)\mathbb{E}[A]$.

Intuitively, we can think of $A$ as a matrix (with rows indexed by $X$ and columns by $Y$, and entries $A(x,y)$); then this condition states that the average of each row of $A$ is at least $1-\varepsilon$ of the global average of $A$ (i.e., the average of the entire matrix).



**Definition 2.6** (Near-uniformity)**.** For $\varepsilon \in (0,1)$ and $k \in \mathbb{N}$, we say $A$ is $(k,\varepsilon)$-near uniform if

$$\mathbb{P}_{(x,y)\in X\times Y}[(1-\varepsilon)\mathbb{E}[A] \le A(x,y) \le (1+\varepsilon)\mathbb{E}[A]] \ge 1-2^{-k}.$$

Intuitively, this states that nearly all entries of $A$ are close to the global average of $A$ (where 'close' is given by $\varepsilon$, and 'nearly all' by $k$).

All three of these conditions can be interpreted as regularity conditions — they measure how close $A$ is to a constant function in some sense (though the senses in which they measure this are different — for example, the first two conditions can be satisfied by a $\{0,1\}$-valued function, while the third cannot be).

## §2.4 The main analytical theorem

The main theorem we'll prove today (which we can think of as the main analytical theorem of the paper), regarding these pseudorandomness conditions, is as follows.

**Theorem 2.7**

Let $A\colon X\times Z \to [0,1]$ and $B\colon Y\times Z\to[0,1]$ be functions, and let $d,k\ge 1$ and $\varepsilon\in(0,1)$. Suppose that:

(1) $\mathbb{E}[A]$ and $\mathbb{E}[B]$ are both at least $2^{-d}$.

(2) $A$ and $B$ are both $(r,\varepsilon)$-spread for $r=\Omega(dk/\varepsilon)$.

(3) $A$ and $B$ are both $\varepsilon$-left-lower bounded.

Then $A\circ B\colon X\times Y\to[0,1]$, defined as $(A\circ B)(x,y)=\mathbb{E}_z[A(x,z)B(y,z)]$, is $(k,320\varepsilon)$-near uniform.

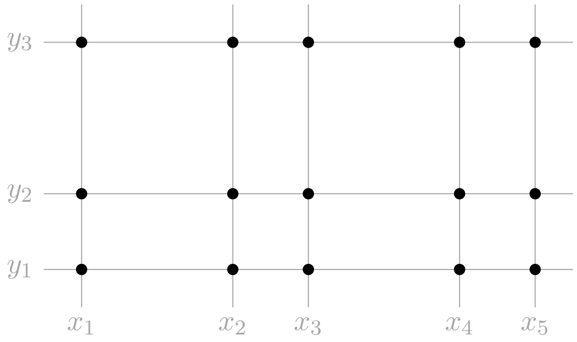
If we think of $A$ and $B$ as matrices, then $A\circ B$ corresponds to the matrix product $AB^\intercal$. In particular, for intuition, we can think of $A$ and $B$ as $\{0,1\}$-valued matrices representing bipartite graphs (between $X$ and $Z$ for $A$, and $Y$ and $Z$ for $B$); then Theorem 2.7 says that if we take two graphs with good regularity properties (specifically, the density between any two large vertex subsets shouldn't be too large, and no vertex on the right should have too small degree), then their matrix product is roughly constant.

To prove Theorem 2.7, we'll need some machinery — we'll introduce a seminorm called the *grid norm* — and prove two lemmas about it.

**Definition 2.8.** For a function $f$ on $X\times Y$ and $\ell,k\in\mathbb{N}$, we define

$$U_{(\ell,k)}(f) = \mathbb{E}_{x_1,\dots,x_\ell\in X,y_1,\dots,y_k\in Y}\left[\prod_{i=1}^{\ell}\prod_{j=1}^{k} f(x_i,y_j)\right].$$

Intuitively, we're choosing a random $\ell\times k$ grid in $X\times Y$ and taking the expectation of the product of the $f$-values of all the grid points.



In particular, if we think of $f$ as representing a bipartite graph, then $U_{(\ell,k)}(f)$ corresponds to the subgraph (or rather, homomorphism) count of $K_{\ell,k}$ in this graph.

**Definition 2.9.** We define the $U_{(\ell,k)}$ norm (or grid norm) of $f$ as $\|f\|_{U_{(\ell,k)}} = |U_{(\ell,k)}(f)|^{1/\ell k}$.

The first step of the proof is the following lemma, which states that if the $U_{(\ell,k)}$ norm of $A$ is large, then $A$ cannot be $(r,\varepsilon)$-spread — equivalently, if $A$ is $(r,\varepsilon)$-spread, then its $U_{(\ell,k)}$ norm is small.

**Lemma 2.10**

If $\|A\|_1 \geq \delta$ and $A$ is $(r,\varepsilon)$-spread with $r = (\ell k+1)\log(1/\delta) + \log(1/\varepsilon)$, then

$$\|A\|_{U_{(\ell,k)}} \leq (1+2\varepsilon)\,\|A\|_1\,.$$

(We use $\|A\|_1$ to denote $\mathbb{E}\,|A|$; this is the same as $\mathbb{E}[A]$, as $A$ is always nonnegative.)

Note that the $U_{(\ell,k)}$ norm of $A$ is always *at least* its 1-norm. So Lemma 2.10 says that the $U_{(\ell,k)}$ norm of $A$ is pretty close to the minimum it could possibly be, which is quite strong.

*Proof.* We'll prove the contrapositive — we'll assume that

$$\|A\|_{U_{(\ell,k)}} > (1+2\varepsilon)\,\|A\|_1\,, \tag{1}$$

and we'll use this to find a large rectangle $R$ on which $A$ is too dense — specifically, such that

$$\mathbb{E}_{(x,y)\in R}[A(x,y)] > (1+\varepsilon)\,\|A\|_1\,,$$

which will violate the spreadness condition.

To start with, we'll raise both sides of our assumption (1) to the $\ell k$th power to get

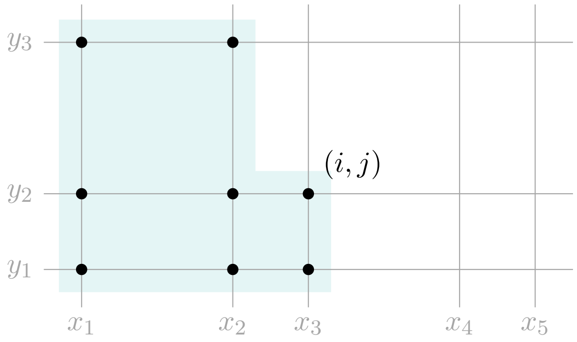
$$\|A\|_{U_{(\ell,k)}}^{\ell k} > (1+2\varepsilon)^{\ell k}\,\|A\|_1^{\ell k}\,. \tag{2}$$

Our goal is to go from here to some equation where the left-hand side looks like a weighted average of $A$, and the right-hand side still gives a lower bound involving $\|A\|_1$. If we can do this, then we can try to decompose this weighted average in terms of rectangles; and then we can just find one rectangle with a large contribution, which will be the one that we take as $R$.

If our goal is to make the left-hand side a weighted average of $A$, then we want to write it as an expectation over just two variables (right now, it's an expectation over $\ell$ and $k$ variables). First, we use $\preceq$ to denote the lexicographic ordering on $[\ell]\times[k]$, and for each $(i,j)\in[\ell]\times[k]$ and for vectors $x = x_1\ldots x_\ell$ and $y = y_1\ldots y_k$ (of lengths $\ell$ and $k$), we consider the prefix product

$$\varphi_{\preceq(i,j)}(x,y) = \prod_{(i',j')\preceq(i,j)} A(x_{i'},y_{j'}).$$

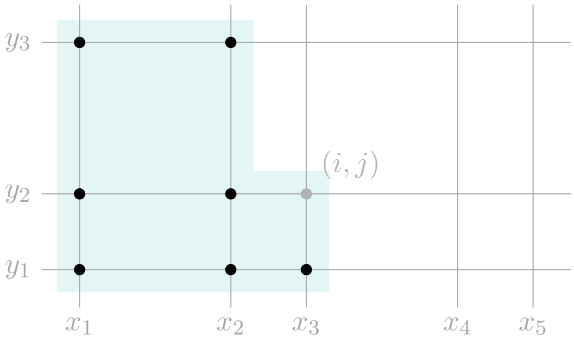
So we're considering the product of $A$ over only the part of the grid that's (weakly) smaller than $(i,j)$ in the lexicographic order.



Similarly, we can also define

$$\varphi_{\prec(i,j)}(x,y) = \prod_{(i',j')\prec(i,j)} A(x_{i'}, y_{j'})$$

(where we have all the same terms except the one from $(i,j)$).



In particular, for vectors $x$ and $y$ of lengths $\ell$ and $k$, we have

$$\|A\|_{U_{(\ell,k)}}^{\ell k} = \mathbb{E}_{x,y}[\varphi_{\preceq(\ell,k)}(x,y)]$$

by definition (since $\varphi_{\leq(\ell,k)}$ corresponds to taking a product over the entire grid). We'll then write this as a telescoping product, as

$$\|A\|_{U_{(\ell,k)}}^{\ell k} = \frac{\mathbb{E}[\varphi_{\preceq(1,1)}]}{1} \cdot \frac{\mathbb{E}[\varphi_{\preceq(1,2)}]}{\mathbb{E}[\varphi_{\prec(1,2)}]} \cdots \frac{\mathbb{E}[\varphi_{\preceq(\ell,k)}]}{\mathbb{E}[\varphi_{\prec(\ell,k)}]} = \prod_{(i,j)\in[\ell]\times[k]} \frac{\mathbb{E}[\varphi_{\preceq(i,j)}]}{\mathbb{E}[\varphi_{\prec(i,j)}]}. \quad (3)$$

There are $\ell k$ terms in this product, and we know from (2) that it's greater than $(1+2\varepsilon)^{\ell k} \|A\|_1^{\ell k}$. So at least one of these terms must be fairly large — there must exist $(i^*, j^*) \in [\ell] \times [k]$ for which we have

$$\frac{\mathbb{E}[\varphi_{\preceq(i^*,j^*)}]}{\mathbb{E}[\varphi_{\prec(i^*,j^*)}]} > (1+2\varepsilon)\|A\|_1 . \quad (4)$$

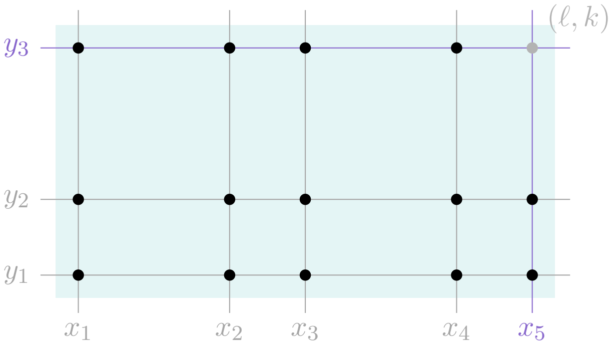
> **Remark 2.11.** One way to think about this step intuitively is that if we think of $A$ as a $\{0,1\}$-valued function representing a subset of $X \times Y$, then on the left-hand side of (3), we're choosing a random grid and looking at the probability that all the grid points are in our set $A$. And we can imagine computing this probability by looking at one point at a time (in lexicographic order), and writing our probability as a product of conditional probabilities (where for each point $(i,j)$, we look at the probability this point is in our set conditional on all the previous points being in the set); these conditional probabilities exactly correspond to the terms in the telescoping product. And we're focusing on one term where this conditional probability is much bigger than what we might expect.

We now fix this special point $(i^*, j^*)$; for simplicity, we'll just assume it's $(\ell, k)$.

Recall that our goal was in some sense to write the left-hand side of (2) as an expectation over two variables (rather than $\ell + k$). We're now getting closer to being able to do so — we define an auxiliary function in two variables as

$$F(x_\ell, y_k) = \mathbb{E}_{x_1,\ldots,x_{\ell-1}\in X, y_1,\ldots,y_{k-1}\in Y} \prod_{(i,j)\prec(\ell,k)} A(x_i, y_j). \quad (5)$$

This is very similar to $\mathbb{E}[\varphi_{\prec(\ell,k)}]$, but here we're *fixing* the values of $x_\ell$ and $y_k$, and only taking an expectation over the remaining variables (of the same product).



Now we can use $F$ to rewrite the relevant expectations from before — we have

$$\mathbb{E}[\varphi_{\prec(\ell,k)}] = \mathbb{E}_{x,y}[F(x,y)] \text{ and } \mathbb{E}[\varphi_{\preceq(\ell,k)}] = \mathbb{E}_{x,y}[F(x,y)A(x,y)].$$

(Here $x \in X$ and $y \in Y$ are single variables, not vectors — they correspond to $x_\ell$ and $y_k$ from before.) We'll use $\langle F, A\rangle$ to denote $\mathbb{E}_{x,y}[F(x,y)A(x,y)]$ (since we can think of this as a normalized inner product). Then plugging this into (4) gives that

$$\frac{\mathbb{E}[\varphi_{\preceq(\ell,k)}]}{\mathbb{E}[\varphi_{\prec(\ell,k)}]} = \frac{\langle F, A\rangle}{\|F\|_1} > (1+2\varepsilon)\,\|A\|_1\,,$$

which we can rearrange to

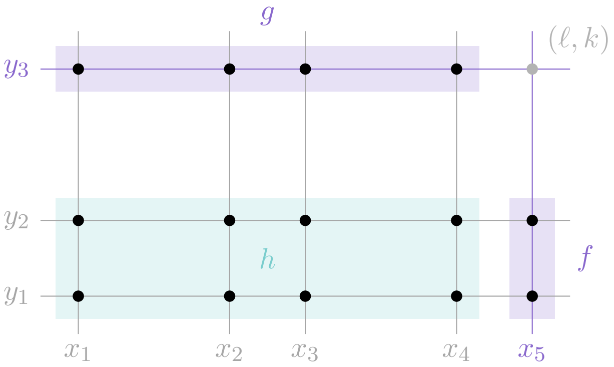
$$\left\langle \frac{F}{\|F\|_1}, \frac{A}{\|A\|_1} \right\rangle > 1 + 2\varepsilon.$$

And we can think of this as a weighted average of $A/\|A\|_1$, where the weight function is given by $F/\|F\|_1$ — so we've accomplished our first goal (which was essentially to lower-bound a weighted average of $A/\|A\|_1$).

Then to finish the proof, we can decompose $F$ as a sum of rectangles — to perform this decomposition, we need to use the fact that we can write $F$ in the form

$$F(x,y) = \mathbb{E}_w[f(x,w)g(y,w)h(w)]$$

(where $x$ and $y$ correspond to $x_k$ and $y_\ell$ in (5), $w$ corresponds to the choices for all the remaining variables $x_1, \ldots, x_{\ell-1} \in X$ and $y_1, \ldots, y_{k-1} \in Y$, and $f$, $g$, and $h$ correspond to the parts of the product involving $x_\ell$, $y_k$, and neither, respectively).



We can use this to decompose $F$ as a sum of rectangles (where we have a rectangle corresponding to each $w$) — intuitively, if $A$ is $\{0,1\}$-valued (so $f$, $g$, and $h$ are as well), then for each $w$, the set of inputs for which $F(x,y) = 1$ forms a rectangle (namely, the rectangle $\{x \mid f(x,w) = 1\} \times \{y \mid g(y,w) = 1\}$).

So now we have a weighted average of $A/\|A\|_1$ over *rectangles* that's larger than it should be, and we can use this to find a sufficiently large rectangle $R$ with a large contribution to this weighted average — meaning that $A$ is overly dense on that rectangle. □

**Remark 2.12.** This proof is similar in spirit to the proof of the counting lemma from graph regularity (and the statement is also kind of similar).

We'll now get to the second lemma; this is called the *sifting of rectangles* step.

**Lemma 2.13**

Fix $\varepsilon \in (0, 1/80)$ and let $p = \lceil k/\varepsilon \rceil$. Let $A\colon X \times Z \to \mathbb{R}_{\geq 0}$ and $B\colon Y \times Z \to \mathbb{R}_{\geq 0}$ be such that:

- $\|A\|_{U(2,p)} \leq (1+\varepsilon)\|A\|_1$.
- $\|B\|_{U(2,p)} \leq (1+\varepsilon)\|B\|_1$.
- $A$ and $B$ are $\varepsilon$-left-lower bounded.

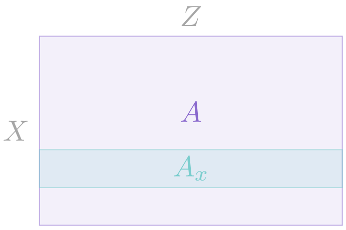
Define the function $D\colon X \times Y \to \mathbb{R}_{\geq 0}$ as

$$D(x,y) = \frac{(A \circ B)(x,y)}{\mathbb{E}[A \circ B]}.$$

Then we have $\|D - 1\|_k \leq 80\varepsilon$.

This lemma together with Lemma 2.10 implies Theorem 2.7.

*Proof.* We can assume that $\mathbb{E}[A] = \mathbb{E}[B] = 1$ (by scaling $A$ and $B$ otherwise). We then define the auxiliary functions $A_x = A(x, \bullet)$ and $B_y = B(y, \bullet)$ — i.e., the row functions of $A$ and $B$. (The statement that $A$ and $B$ are $\varepsilon$-left-lower bounded translates to a condition on these auxiliary functions, which we'll use later.)



Now we can write $D$ as

$$D(x,y) = \frac{\langle A_x, B_y\rangle}{\mathbb{E}_{x,y}[\langle A_x, B_y\rangle]}$$

(where we use $\langle f, g\rangle$ to denote $\mathbb{E}_z f(z)g(z)$). We're going to show that

$$\|\langle A_x, B_y\rangle - 1\|_k = O(\varepsilon)$$

(where when we talk about the $k$-norm of this quantity, we're viewing it as a function of $x$ and $y$). This will also mean that the denominator of $D$, namely $\mathbb{E}_{x,y}[\langle A_x, B_y\rangle]$, is close to 1; and then this gives the desired statement (that $D$ is close to 1 in $k$-norm).

We're interested in bounding the $k$-norm of $\langle A_x, B_y\rangle - 1$, and we'll do so by decomposing this into three terms and using the triangle inequality — we have

$$\langle A_x, B_y\rangle - 1 = \langle A_x - 1, B_y - 1\rangle + \langle A_x - 1, 1\rangle + \langle B_y - 1, 1\rangle, \tag{6}$$

so it suffices to bound the $k$-norm of each term on the right-hand side individually.

We'll first bound $\|\langle A_x - 1, 1\rangle\|_k$. To do so, define the function $a\colon X \to \mathbb{R}_{\geq 0}$ as

$$a(x) = \mathbb{E}_z[A_x(z)],$$

i.e., $a(x)$ is the average of the $x$th row of $A$. Then by definition, we have

$$\left\|\langle A_x - 1, 1\rangle\right\|_k = \left\|a - 1\right\|_k.$$

We'll bound this by focusing on its positive part and negative part separately — we define

$$(a-1)_+ = \max(0, a-1) \text{ and } (a-1)_- = \max(0, 1-a)$$

(the first term captures all the positive parts of $a - 1$, and the second captures all the negative parts, but flipped to be positive). Then $a - 1 = (a-1)_+ - (a-1)_-$, so by the triangle inequality we have

$$\left\|a-1\right\|_k \le \left\|(a-1)_+\right\|_k + \left\|(a-1)_-\right\|_k,$$

which means it suffices to bound the positive and negative parts separately.

For the negative part, we use the fact that $A$ is $\varepsilon$-left-lower bounded, which means $a(x) \ge 1 - \varepsilon$ for *all* $x$ (since we assumed $\mathbb{E}[A] = 1$). So $(a-1)_-$ is at most $\varepsilon$ pointwise, which means its $k$-norm is also at most $\varepsilon$.

Bounding $\left\|(a-1)_+\right\|_k$ involves more steps (and another lemma). The idea is that we first use the fact that $\left\|A\right\|_{U_{(2,p)}}$ is small to show that $\left\|a\right\|_k \le 1 + \varepsilon$. (So we can essentially pass down from the $U_{(2,p)}$ norm to the $k$-norm, where $p = \lceil k/\varepsilon \rceil$.) Then since even the small values of $a$ are always at least $1 - \varepsilon$, this means the large values of $a$ can't be *too* large (i.e., too far away from 1) either. So with some work we can get the bound $\left\|(a-1)_+\right\|_k \le 4\varepsilon$, which means $\left\|a-1\right\|_k \le 5\varepsilon$.

We can bound the term $\left\|\langle B_y - 1, 1\rangle\right\|_k$ in (6) in the same way. We're not going to go through the proof of how we bound the first term $\left\|\langle A_x - 1, B_y - 1\rangle\right\|$, but here's a high-level overview — we first have

$$\left\|\langle A_x - 1, B_y - 1\rangle\right\|_k \le \left\|A - 1\right\|_{U(2,k)} \left\|B - 1\right\|_{U_{(2,k)}}.$$

And then we can use some more facts to pass from bounds on $\left\|A\right\|_{U_{(2,p)}}$ (where $p = \lceil k/\varepsilon \rceil$ is much larger than $k$) to bounds on $\left\|A-1\right\|_{U_{(2,k)}}$. This will end up giving us a bound of $O(\varepsilon)$, as desired. □

## §2.5 The conclusion

Finally, we'll talk about how Theorem 2.7 relates to the big-picture idea of the proof of Theorem 1.10. Recall that in Theorem 1.10, we had an explicitly defined set $D \subseteq X \times Y \times Z$, and as described in Subsection 2.2, in order to lower-bound the deterministic communication complexity of $\mathbf{1}[D]$, we wanted to show that any cylinder intersection can only occupy a small fraction of $D$.

It's not obvious how to get there, but the idea is that we first use Theorem 2.7 to say that $D$ is 'quasirandom' in some sense (in particular, in a sense closely related to rectangles). Then we're taking a cylinder intersection, which is in some sense a kind of rectangle-like object, and looking at its intersection with $D$. And if the cylinder intersection has total density $\alpha$ in the global set, we can use the quasirandomness of $D$ to show that it only fills up about an $\alpha$-fraction of $D$ as well.