# Special Topics in Complexity Theory

Emanuele Viola

December 28, 2017

This document collects the Fall 2017 lectures given by the instructor. Compared to the lectures on the class website, I have made some minor edits to harmonize the various lectures. In addition we had two guest lectures and presentations by students, which can also be found on the class website. Many thanks to Matthew Dippel, Xuangui Huang, Chin Ho Lee, Biswaroop Maiti, Tanay Mehta, Willy Quach, and Giorgos Zirdelis for doing an excellent job scribing these lectures. Many thanks also to all the students, postdocs, and faculty who attended the class and created a great atmosphere.

## Contents

# 1 Bounded independence

In this first lecture we begin with some background on pseudorandomness and then we move on to the study of bounded independence, presenting in particular constructions and lower bounds.

## 1.1 Background

Let us first give some background on randomness. There are 3 different theories:

(1) Classical probability theory. For example, if we toss a coin 12 times then the probability of each outcome is the same, i.e., $\Pr[010101010101] = \Pr[011011100011]$. However, intuitively we feel that the first outcome is less random than the second.

(2) Kolmogorov complexity. Here the randomness is measured by the length of the shortest program outputting a string. In the previous example, the program for the second outcome could be "print `011011100011`", whereas the program for the first outcome can be "print `01` six times", which is shorter than the first program.

(3) Pseudorandomness. This is similar to resource-bounded Kolmogorov complexity. Here random means the distribution "looks random" to "efficient observers."

Let us now make the above intuition precise.

**Definition 1.**[Pseudorandom generator (PRG)] A function $f\colon \{0,1\}^s \to \{0,1\}^n$ is a *pseudorandom generator (PRG)* against a class of tests $T \subseteq \{t\colon \{0,1\}^n \to \{0,1\}\}$ with error $\epsilon$, if it satisfies the following 3 conditions:

(1) the output of the generator must be longer than its input, i.e., $n > s$;

(2) it should *fool* $T$, that is, for every test $t \in T$, we have $\Pr[t(U_n) = 1] = \Pr[t(f(U_s)) = 1] \pm \epsilon$;

(3) the generator must be efficient.

To get a sense of the definition, note that a PRG is easy to obtain if we drop any one of the above 3 conditions. Dropping condition (1), then we can define our PRG as $f(x) := x$. Dropping condition (2), then we can define our PRG as $f(x) := 0$. Dropping condition (3), then the PRG is not as obvious to obtain as the previous two cases. We have the following claim.

**Claim 2.** For every class of tests $T$, there exists an inefficient PRG with error $\epsilon$ and seed length $s = \lg_2 \lg_2(|T|) + 2\lg_2(1/\epsilon) + O(1)$.

Before proving the claim, consider the example where $T$ is the class of circuits of size $n^{100}$ over $n$-bit input, it is known that $|T| = 2^{n^{O(1)}}$. Hence, applying our claim above we see that there is an inefficient PRG that fools $T$ with error $\epsilon$ and seed length $s = O(\lg_2(n/\epsilon))$.

We now prove the claim using the probabilistic method.

*Proof.* Consider picking $f$ at random. Then by the Chernoff bound, we have for every test $t \in T$,

$$\Pr_f[|\Pr_{U_s}[t(f(U_s)) = 1] - \Pr_{U_n}[t(U_n) = 1]| \geq \epsilon] \leq 2^{-\Omega(\epsilon^2 2^s)} < 1/|T|,$$

if $s = \lg_2 \lg_2(|T|) + 2\lg_2(1/\epsilon) + O(1)$. Therefore, by a union bound over $t \in T$, there exists a fixed $f$ such that for every $t \in T$, the probabilities are within $\epsilon$. □

## 1.2 $k$-wise independent distribution

A major goal in research in pseudorandomness is to construct PRGs for (1) richer and richer class $T$, (2) smaller and smaller seed length $s$, and making the PRG explicit. For starters, let us consider a simple class of tests.

**Definition 3.**[$d$-local tests] The *$d$-local tests* are tests that depend only on $d$ bits.

We will show that for this class of tests we can actually achieve error $\epsilon = 0$. To warm up, consider what happens when $d = 1$, then we can have a PRG with seed length $s = 1$ by defining $f(0) := 0^n$ and $f(1) := 1^n$.

For $d = 2$, we have the following construction. Define

$$f(x)_y := \langle x, y \rangle = \sum_i x_i y_i \bmod 2.$$

Here the length of $x$ and $y$ is $|x| = |y| = \lg_2 n$, and we exclude $y = 0^{\lg_2 n}$. Note that the output has $n - 1$ bits, but we can append one uniform bit to the output of $f$. So the seed length would be $\lg_2 n + 1$.

Now we prove the correctness of this PRG.

**Claim 4.** The $f$ defined above is a PRG against 2-local tests with error $\epsilon = 0$.

*Proof.* We need to show that for every $y \neq z$, the random variable $(f(x)_y, f(x)_z)$ over the choice of $x$ is identical to $U_2$, the uniform 2-bit string. Since $y \neq z$, suppose without loss of generality that there exists an $i$ such that $y_i = 1$ and $z_i = 0$. Now $f(x)_z$ is uniform, and conditioned on $z$, $f(x)_y$ is also uniform, thanks to the index $y_i$. □

The case for $d = 3$ becomes much more complicated and involves the use of *finite fields*. One can think of a finite field as a finite domain that behaves like $\mathbb{Q}$ in the sense that it allows you to perform arithmetic operations, including division, on the elements. We will use the following fact about finite fields.

**Lemma 5.** There exist finite fields of size $p^k$, for every prime $p$ and integer $k$. Moreover, they can be constructed and operated with in time $\mathrm{poly}(k, p)$.

**Remark 6.** Ideally one would like the dependence on $p$ to be $\lg_2 p$. However, such construction remains an open question and there have been many attempts to constructing finite fields in time $\mathrm{poly}(k, \lg_2 p)$. Here we only work with finite fields with $p = 2$, and there are a lot of explicit constructions for that.

One simple example of finite fields are integers modulo $p$.

**Theorem 7.** Let $D = \{0,1\}^{\lg_2 n}$. For every $k$, there exists an explicit construction over $D^n$ such that

(1) elements in $D^n$ can be sampled with $s = k \lg_2 n$ bits, and

(2) every $k$ symbols are uniform in $D^k$.

For $d = 3$, we can use the above theorem with $k = 3$, and the PRG can output the first bit of every symbol.

**Remark 8.** There exist other constructions that are similar to the inner product construction for the case $d = 2$, with $y$ carefully chosen, but the way to choose $y$ involves the use of finite fields as well.

Note that we can also apply the theorem for larger $d$ to fool $d$-local tests with seed length $s = d \lg_2 n$.

We now prove the theorem.

*Proof.* Pick a finite field $\mathbb{F}$ of size $2^{\lg_2 n}$. Let $a_0, \ldots, a_{n-1} \in \mathbb{F}$ be uniform random elements in $\mathbb{F}$ which we think of as a polynomial $a(x)$ of degree

$k-1$. We define the generator $f$ to be

$$f(a_0, \ldots, a_{n-1})_x = a(x) = \sum_{i=0}^{n-1} a_i x^i.$$

(One should think of the outputs of $f$ as lines and curves in the real plane.)

The analysis of the PRG follows from the following useful fact: For every $k$ points $(x_0, y_0), (x_1, y_1), \ldots, (x_{k-1}, y_{k-1})$, there exists exactly one degree $k-1$ polynomial going through them. □

Let us now introduce a terminology for PRGs that fool $d$-local tests.

**Definition 9.** We call distributions that look uniform (with error 0) to $k$-local tests *$k$-wise independent* (also known as $k$-wise uniform). The latter terminology is more precise, but the former is more widespread.

We will soon see an example of a distribution where every $k$ elements are independent but not necessarily uniform.

## 1.3 Lower bounds

We have just seen a construction of $k$-wise independent distributions with seed length $s = d \lg_2 n$. It is natural to ask, what is the minimum seed length of generating $k$-wise independent distributions?

**Claim 10.** For every $k \geq 2$, every PRG for $k$-local tests over $\{0,1\}^n$ has seed length $s \geq \Omega(k \lg_2(n/k))$.

*Proof.* We use the linear-algebraic method. See the book by Babai–Frankl [BF92] for more applications of this method.

To begin, we will switch from $\{0,1\}$ to $\{-1,1\}$, and write the PRG as a $2^s \times n$ matrix $M$, where the rows are all the possible outputs of the PRG. Since the PRG fools $k$-local tests and $k \geq 2$, one can verify that every 2 columns of $M$ are orthogonal, i.e., $\langle M_i, M_j \rangle = 0$ for $i \neq j$. As shown below, this implies that the vectors are independent. And by linear algebra this gives a lower bound on $s$.

However so far we have not used $k$. Here's how to use it. Consider all the column vectors $v$ obtained by taking the entry-wise products of any of the $k/2$ vectors in $M$. Because of $k$-wise independence, these $v$'s are again orthogonal, and this also implies that they are linearly independent.

**Claim 11.** If $v_1, v_2, \ldots, v_t$ are orthogonal, then they are linearly independent.

*Proof.* Suppose they are not and we can write $v_i = \sum_{j \in S, i \notin S} v_j$ for some $S$. Taking inner product with $v_i$ on both sides, we have that the L.H.S. is nonzero, whereas the R.H.S. is zero because the vectors are orthogonal, a contradiction. □

Therefore, the rank of $M$ must be at least the number of $v$'s, and so

$$2^s \geq \text{number of v's} \geq \binom{n}{k/2} \geq (2n/k)^{k/2}.$$

Rearranging gives $s \geq (k/2)\lg_2(2n/k)$. □

## 1.4 Who is fooled by $k$-wise independence?

In the coming lectures we will see that $k$-wise independence fools $\mathrm{AC}^0$, the class of constant-depth circuits with unbounded fan-in. Today, let us see what else is fooled by $k$-independence in addition to $k$-local tests.

(1) Suppose we have $n$ independent variables $x_1, \ldots, X_n \in [0,1]$ and we want to understand the behavior of their sum $\sum_i X_i$. Then we can apply tools such as the Chernoff bound, tail bounds, Central Limit Theorem, and the Berry–Esseen theorem. The first two give bounds on large deviation from the mean. The latter two are somewhat more precise facts that show that the sum will approach a normal distribution (i.e., the probability of being larger than $t$ for any $t$ is about the same). One can show that similar results hold when the $X_i$'s are $k$-wise independent. The upshot is that the Chernoff bound gives error $2^{-\text{samples}}$, while under $k$-wise independence we can only get an error $(\text{samples})^{-k/2}$.

(2) We will see next time that $k$-wise independence fools DNF and $\mathrm{AC}^0$.

(3) $k$-wise independence is also used as hashing in load-balancing.

### 1.4.1 $k$-wise independence fools AND

We now show that $k$-wise independent distributions fool the AND function.

**Claim 12.** Every $k$-wise uniform distribution fools the AND functions on bits with error $\epsilon = 2^{-\Omega(k)}$.

*Proof.* If the AND function is on at most $k$ bits, then by definition the error is $\epsilon = 0$. Otherwise the AND is over more than $k$ bits. Without loss of generality we can assume the AND is on the first $t > k$ bits. Observe that for any distribution $D$, we have

$$\Pr_D[\text{AND on t bits is 1}] \leq \Pr_D[\text{AND on k bits is 1}].$$

The right-hand-side is the same under uniform and $k$-wise uniformity, and is $2^{-k}$. Hence,

$$|\Pr_{\text{uniform}}[AND = 1] - \Pr_{\text{k-wise ind.}}[\text{AND} = 1]| \leq 2^{-k}.$$

□

Instead of working over bits, let us now consider what happens over a general domain $D$. Given $n$ functions $f_1, \ldots, f_n \colon D \to \{0,1\}$. Suppose $x_1, \ldots, x_n$ are $k$-wise uniform over $D^n$. What can you say about the AND of the outputs of the $f_i$'s, $f_1(x_1), f_2(x_2), \ldots, f_n(x_n)$?

This is similar to the previous example, except now that the variables are independent but not necessarily uniform. Nevertheless, we can show that a similar bound of $2^{-\Omega(k)}$ still holds.

**Theorem 13.**[[EGL$^+$92]] Let $X_1, X_2, \ldots, X_n$ be random variables over $\{0,1\}$, which are $k$-wise independent, but not necessarily uniform. Then

$$\Pr[\prod_{i=1}^{n} X_i = 1] = \prod_{i=1}^{n} \Pr[X_i = 1] \pm 2^{-\Omega(k)}.$$

This fundamental theorem appeared in the conference version of [EGL$^+$92], but was removed in the journal version. One of a few cases where the journal version contains *less* results than the conference version.

*Proof.* Let $D$ be the distribution of $(X_1, \ldots, X_n)$. Let $B$ be the $n$-wise independent distribution $(Y_1, \ldots, Y_n)$ such that $\Pr[Y_i = 1] = \Pr[X_i = 1]$ for all $i \in [n]$ and the $Y_i$ are independent. The theorem is equivalent to the following statement.

$$|\Pr_{X \leftarrow D}\left[\bigwedge_{i=1}^{n} X_i = 1\right] - \Pr_{X \leftarrow B}\left[\bigwedge_{i=1}^{n} X_i = 1\right]| \leq 2^{-\Omega(k)}$$

We will prove the above statement by the following version of the Inclusion-Exclusion principle.

**Inclusion-Exclusion Principle** Let $V$ be any distribution over $\{0,1\}^n$. Note that by De Morgan's laws, we have

$$\Pr\left[\bigwedge V_i = 1\right] = 1 - \Pr\left[\bigvee V_i = 0\right]$$

Let $E_i$ be the event that $V_i = 0$. We want to bound the quantity $\Pr\left[\bigcup E_i\right]$. By looking at the Venn diagram of the events $E_i$, we can see that

$$\Pr\left[\bigcup E_i\right] \leq \Pr[E_1] + \cdots + \Pr[E_n] = \sum_i \Pr[E_i]$$

$$\Pr\left[\bigcup E_i\right] \geq \sum_i \Pr[E_i] - \sum_{i,j} \Pr[E_i \cap E_j]$$

$$\Pr\left[\bigcup E_i\right] \leq \sum_i \Pr[E_i] \quad - \sum_{S\subseteq[n],|S|=2} \Pr\left[\bigcap_{i\in S} E_i\right] \quad + \sum_{S\subseteq[n],|S|=3} \Pr\left[\bigcap_{i\in S} E_i\right],$$

and so on. In general, we have the following. Define

$$T_j := \sum_{S\subseteq[n],|S|=j} \Pr\left[\bigcap_{i\in S} E_i\right]$$

$$S_h := \sum_{i=1}^{h} (-1)^{i+1} T_i$$

Then, we have the bounds $\Pr\left[\bigcup E_i\right] \leq S_j$ for odd $j$, and $\Pr\left[\bigcup E_i\right] \geq S_j$ for even $j$. This fact holds for *any* distribution.

Let us return to the proof. Note that the $S_h$ are the same for $D$ and $B$ up to $h = k$ because they only involve sums of ANDs of at most $k$ events. Hence, we have that

$$\left|\Pr_D\left[\bigwedge X_i = 1\right] - \Pr_B\left[\bigwedge X_i = 1\right]\right| \leq |S_k - S_{k-1}| = |T_k|$$

where the last equality comes from the definition of $S_k$. Therefore, we are done if $|T_k| \leq 2^{-\Omega(k)}$. We have that

$$T_k = \sum_{S\subseteq[n],|S|=k} \Pr\left[\bigcap_{i\in S} E_i\right] = \binom{n}{k} \ \mathbb{E}_{S\subseteq[n],|S|=k}\left[\prod_{i\in S} P_i\right]$$

where $P_i := \Pr[E_i] = 1 - \Pr[X_i = 1]$. To bound the expectation we recall a useful inequality.

**A Useful Inequality** Let $Q_1, \ldots, Q_n$ be non-negative real numbers. Then, by the AM-GM inequality, we have that

$$\frac{\sum_i Q_i}{n} \geq \left(\prod_i Q_i\right)^{1/n}.$$

Consider the following more general statement,

$$\mathbb{E}_{S\subseteq[n],|S|=1}\left[\prod_{i\in S} Q_i\right] \geq \mathbb{E}_{S\subseteq[n],|S|=2}\left[\prod_{i\in S} Q_i\right]^{1/2} \geq \cdots$$
$$\cdots \geq \mathbb{E}_{S\subseteq[n],|S|=k}\left[\prod_{i\in S} Q_i\right]^{1/k} \geq \cdots \geq \mathbb{E}_{S\subseteq[n],|S|=n}\left[\prod_{i\in S} Q_i\right]^{1/n}$$

and note that the left most term is equal to $\frac{\sum_i Q_i}{n}$, while the right most term is equal to $\left(\prod_i Q_i\right)^{1/n}$

Applying the above inequality to $T_k$ and a common approximation for the binomial coefficient, we have that

$$T_k = \binom{n}{k}\ \mathbb{E}_{S\subseteq[n],|S|=k}\left[\prod_{i\in S} P_i\right] \leq \binom{n}{k}\sum_{i=1}^{n}\left(\frac{P_i}{n}\right)^k \leq \left(\frac{en}{k}\right)^k\left(\frac{\sum P_i}{n}\right)^k = \left(\frac{e\sum P_i}{k}\right)^k.$$

Therefore, we are done if $\sum P_i \leq \frac{k}{2e}$. Recall that $P_i = \Pr[E_i] = 1 - \Pr[X_i = 1]$. So if $P_i$ is small then $\Pr[X_i = 1]$ is close to 1.

It remains to handle the case that $\sum P_i \geq \frac{k}{2e}$. Pick $n'$ such that

$$\sum_{i=1}^{n'} P_i = \frac{k}{2e} \pm 1.$$

By the previous argument, the AND of the first $n'$ is the same up to $2^{-\Omega(k)}$ for $D$ and $B$. Also, for every distribution the probability of that the And of $n$ bits is 1 is at most the probability that the And of $n'$ bits is 1. And also, for the $n$-wise independent distribution $B$ we have

$$\begin{aligned}\Pr_B\left[\bigwedge_{i=1}^{n'} X_i = 1\right] &= \prod_{i=1}^{n'} \Pr[X_i = 1]\\ &= \prod_{i=1}^{n'} (1 - P_i)\\ &\le \left(\frac{\sum_{i=1}^{n'}(1-P_i)}{n'}\right)^{n'} \quad \text{by the AM-GM inequality}\\ &\le \left(\frac{n' - k/2e}{n'}\right)^{n'} \le (1 - k/2en')^{n'} \le e^{-\Omega(k)}.\end{aligned}$$

The combination of these facts concludes this case. To summarize, in this case we showed that

$$\Pr_D[\bigwedge_{i=1}^{n} X_i = 1] \le \Pr_D[\bigwedge_{i=1}^{n'} X_i = 1].$$

as well as

$$\Pr_B[\bigwedge_{i=1}^{n} X_i = 1] \le \Pr_B[\bigwedge_{i=1}^{n'} X_i = 1] \le 2^{-\Omega(k)}.$$

By the choice of $n'$ and the previous argument, we also know that $|\Pr_D[\bigwedge_{i=1}^{n'} X_i = 1] - \Pr_B[\bigwedge_{i=1}^{n'} X_i = 1]| \le 2^{-\Omega(k)}$ and so we are done, as all quantities above are at most $2^{-\Omega(k)}$ (and at least 0). □

**Remark 14.** The bound is tight up to $\Omega(.)$

*Proof.* Let $D$ be the distribution over $\{0,1\}^{k+1}$ as follows: $D_{1,\dots,k} = U_k$ and $D_{k+1} = D_1 + \cdots + D_k \mod 2$. Then, $D$ is $k$-wise independent. However, if $k$ is even, then

$$\Pr[\bigwedge_{i=1}^{k+1} D_i = 1] = 0.$$

Yet, we have that

$$\Pr[\bigwedge_{i=1}^{k+1} U_i = 1] = 2^{-(k+1)}.$$

□

## 1.5 Bounded Independence Fools $\mathrm{AC}^0$

**Acknowledgement**. This section is based on Amnon Ta-Shma's notes for the class 0368.4159 Expanders, Pseudorandomness and Derandomization CS dept, Tel-Aviv University, Fall 2016.

Note that a DNF on $n$ bits can be modeled as a depth two circuit where the top layer is an OR-gate whose inputs are AND-gates, which take inputs $X_1, \ldots, X_n$ and their negations. The circuit class $\mathrm{AC}^0$ can be viewed as a generalization of this to higher (but constant) depth circuits. That is, $\mathrm{AC}^0$ consists of circuits using AND-gates, OR-gates, NOT-gates, and input registers. Each of the gates have unbounded fan-in (i.e. the number of input wires). The size of the circuit is defined to be the number of gates.

$\mathrm{AC}^0$ is one of the most studied classes in complexity theory. $\mathrm{AC}^0$ circuits of polynomial size can do many things, including adding and subtracting $n$-bit integers.

**Conjecture 15.**[Linial-Nisan[LN90]] $\log^{O(d)}$ $s$-wise independence fools $\mathrm{AC}^0$ circuits of depth $d$ and size $s$.

The conjecture was open for a long time, even for in the special case $d = 2$. In 2007 a breakthrough work by Bazzi [Baz09] proved it for $d = 2$. Shortly afterwards, Razborov presented a simpler proof of Bazzi's result [Raz09], and Braverman proved the conjecture for any $d$ with $\log^{d^2}$ $s$-wise independence [Bra10]. Tal improved the result to $\log^{O(d)} s$ [Tal17].

Interestingly, the progress on the conjecture does not use ideas that were not around since the time of its formulation. Bottom line: if a problem is open for a long time, you should immediately attack it with existing tools.

The high-level intuition why such a result should be true is the following:

1. $\mathrm{AC}^0$ is approximated by polylog degree polynomials.

2. $k$-wise independence fools degree-$k$ polynomials.

*Proof of (2).* Let $x = (x_1, \ldots, x_n) \in \{0,1\}^n$. Let $p(x_1, \ldots, x_n)$ be a degree $k$ polynomial over $\mathbb{R}$. Write $p$ as

$$p(x_1, \ldots, x_n) = \sum_{M \subseteq [n], |M| \leq k} c_M \cdot x_M.$$

If $D$ is a $k$-wise independent distribution on $\{0,1\}^n$, then by linearity of expectation

$$\mathbb{E}_D[P] = \sum_{M\subseteq[n],|M|\leq k} c_M \mathbb{E}_D[x_M] = \sum_{M\subseteq[n],|M|\leq k} c_M \mathbb{E}_U[x_M] = \mathbb{E}_U[P].$$

□

There are several notions of approximating AC$^0$ by low-degree polynomials. We now review two of them, explaining why neither of them is sufficient. Braverman showed how to cleverly combine the two methods to prove a version of (1) that's strong enough.

### 1.5.1 Approximation 1

**Theorem 16.** For all AC$^0$ circuits $C(x_1,\ldots,x_n)$ of size $s$ and depth $d$, for all distributions $D$ over $\{0,1\}^n$, for all $\epsilon$, there exists a polynomial $p(x_1,\ldots,x_n)$ of degree $\log^{O(d)} s/\epsilon$ such that

$$\Pr_{x\leftarrow D}[p(x) = C(x)] \geq 1-\epsilon.$$

The important features of this approximation are that it works under any distribution, and when the polynomial is correct it outputs a boolean value.

Similar approximations appear in many papers, going back to Razborov's paper [Raz87] (who considers polynomials modulo 2) which uses ideas from earlier still work.

Note that the polynomial $p$ depends on the circuit $C$ chosen, and on the distribution. This theorem is not a good enough approximation because on the $\epsilon$ fraction of inputs where the polynomial and circuit are unequal, the value of the polynomial can (and does) explode to be much greater than $1/\epsilon$. This prevents us from bounding the average of the polynomial.

Nevertheless, let us prove the above theorem.

*Proof.* Consider one OR-gate of fan-in $s$. We construct a distribution of polynomials that compute any input with high probability. This implies that there is a fixed polynomial that computes the circuit on a large fraction of the inputs by an averaging argument.

For $i = 1, 2, \ldots, \log s$. let $S_i$ be a random subset of $[s]$ where every element is included with probability $1/2^i$, independently.

Suppose $x$ has Hamming weight $2^j$. Then, $\mathbb{E}[\sum_{n \in S_j} x_n] = 1$. And the sum can be shown to equal 1 with constant probability.

Define the approximation polynomial $p$ to be

$$p(x) := 1 - \prod_{i=1}^{\log s} (1 - \sum_{h \in S_i} x_h)$$

Note that if $x$ has weight $w > 0$, then $p(x) = 0$ with constant probability. If $w = 0$, then $p(x) = 1$ with probability 1. We can adjust the error probability to $\epsilon$ by repeating each term in the product $\log(1/\epsilon)$ times.

Thus, we can approximate one gate with the above polynomial of degree $O(\log(s) \cdot \log(1/\epsilon))$. Construct polynomials as $p$ above for each gate, with error parameter $\epsilon/s$. The probability that any of them is wrong is at most $\epsilon$ by a union bound. To obtain the approximating polynomial for the whole circuit compose all the polynomials together. Since the circuit is of depth $d$, the final degree of the approximating polynomial is $(\log(s) \cdot \log(s/\epsilon))^d$, as desired.

As mentioned at the beginning, this is a distribution on polynomials that computes correctly any input with probability at least $1 - \epsilon$. By averaging, there exists a fixed polynomial that computes correctly a $1 - \epsilon$ fraction of inputs. □

It can be verified that the value of the polynomial can be larger than $1/\epsilon$. The polynomial for the gates closest to the input can be as large as $s$. Then at the next level it can be as large as $s^{\log s/\epsilon}$, which is already much larger than $1/\epsilon$.

## 1.6 Approximation 2

**Theorem 17.** For all circuits $C$ of size $s$ and depth $d$, for all error values $\epsilon$, there exists a polynomial $p(x_1, \ldots, x_n)$ of degree $O(\log(s)^{d-1} \log(1/\epsilon))$ such that

$$\mathbb{E}_{x \leftarrow U_n}[(C(x) - p(x))^2] \leq \epsilon.$$

The important feature of this approximation is that it bounds the average, but only under the uniform distribution. Because it does not provide any guarantee on other distributions, including $k$-wise independent distributions, it cannot be used directly for our aims.

**Remark 18.** Approximation 2 is proved via the *switching lemma*, an influential lemma first proved in the early 80's by Ajtai [Ajt83] and by Furst, Saxe, and Sipser [FSS84]. The idea is to randomly set a subset of the variables to simplify the circuit. You can do this repeatedly to simplify the circuit even further, but it only works on the uniform distribution. Hastad [Hås87] gave a much tighter analysis of the switching lemma, and the paper [LMN93] used it to prove a version of Approximation 2 with a slightly worse dependence on the error. Recently, a refinement of the switching lemma was proved in [Hås14, IMP12]. Based on that, Tal [Tal17] obtained the corresponding refinement of Approximation 2 where the parameters are as stated above. (The polynomial is simply obtained from the Fourier expansion of the function computed by the circuit by removing all Fourier coefficients larger than a certain threshold. The bound on the Fourier decay in [Tal17] implies the desired approximation.)

## 1.7 Bounded Independence Fools $\mathrm{AC}^0$

**Theorem 19.** For all circuits $C$ with unbounded fan-in of size $s$ and depth $d$, for all error values $\epsilon$, for all $k$-wise independent distributions $D$ on $\{0,1\}^n$, we have that

$$|\mathbb{E}[C(D)] - \mathbb{E}[C(U_n)]| \leq \epsilon$$

for $k = \log(s/\epsilon)^{O(d)}$.

**Corollary 20.** In particular, if $s = \mathrm{poly}(n)$, $d = O(1)$, $s = 1/\mathrm{poly}(n)$, then $k = \log^{O(1)}(n)$ suffices.

The next claim is the ultimate polynomial approximation used to prove the theorem.

**Claim 21.** For all circuits $C$ with unbounded fan-in of size $s$ and depth $d$, for all error values $\epsilon$, for all $k$-wise independent distributions $D$ on $\{0,1\}^n$, there is a set $E$ of inputs, and a degree-$k$ polynomial $p$ such that:

1. $E$ is 'rare' under both $D$ and $U_n$:

$\Pr_{x \leftarrow U_n}[E(x) = 1] \leq \epsilon$, and $\Pr_{x \leftarrow D}[E(x) = 1] \leq \epsilon$. Here we write $E(x)$ for the indicator function of the event $x \in E$.

2. For all $x$, $p(x) \leq C(x) \vee E(x)$. Here $\vee$ is the logical Or.

3. $\mathbb{E}[p(U_n)] = \mathbb{E}[C(U_n)] \pm \epsilon$.

We only need (1) under $D$, but (1) under $U$ is used to prove (3).

*Proof of Theorem 19 from Claim 21.*

$$\begin{aligned}
\mathbb{E}[C(D)] &= \mathbb{E}[C(D) \vee E(D)] \pm \epsilon, \text{ by Claim.(1)} \\
&\geq \mathbb{E}[p(D)] \pm \epsilon, \text{ by Claim.(2)} \\
&= \mathbb{E}[p(U_n)] \pm \epsilon, \text{ because } p \text{ has degree } k \text{ and } D \text{ is } k\text{-wise independent} \\
&= \mathbb{E}[C(U_n)] \pm \epsilon, \text{ by Claim.(3)}
\end{aligned}$$

For the other direction, repeat the argument for 'not $C$'. □

We can construct the polynomial approximation from Claim 21 by using a combination of Approximation 1 and 2. First we need a little more information about Approximation 1.

**Claim 22.** Two properties of approximation 1:

1. For all $x$, $p(x) \leq 2^{\log(s/\epsilon)^{O(d)}}$.

2. The 'bad' set $E$ is computable by a circuit of size poly$(s)$, and depth $d + O(1)$.

*Proof of Claim 22 part 2.* Consider a single OR gate with input gates $g_1, \ldots, g_s$. This is represented in the approximating polynomial by the term

$$1 - \prod_{i=1}^{\text{polylog}(s/\epsilon)} (1 - \sum_{j \in S_i} g_j).$$

Note that the term is incorrect exactly when the input $g_1, \ldots, g_s$ has weight $> 0$ but all the sets $S_i$ intersect 0 or $\geq 2$ ones. This can be checked in AC$^0$, in parallel for all gates in the circuit. □

*Proof of Claim 21.* Run approximation 1 for the distribution $\frac{D+U}{2}$, yielding the polynomial $p_c$ and the set $E$. This already proves the first part of the claim for both $D$ and $U$, because if $E$ has probability $\epsilon$ under $D$ it has probability $\geq \epsilon/2$ under $(D+U)/2$, and the same for $U$ . Use Claim 22 part 2, and run approximation 2 on $E$. Call the resulting polynomial $p_E$, which has degree $\log(s/\delta)^{O(d)}$ with error bound $\delta$.

The idea in the ultimate approximating polynomial is to "check if there is a mistake, and if so, output 0. Otherwise, output $C$". Formally:

$$p(x) := 1 - (1 - p_c(1 - p_E))^2$$

Claim 21 part 2 can be shown as follows. $p(x) \leq 1$ by definition. So, if $C(x) \vee E(x) = 1$, then we are done. Otherwise, $C(x) \vee E(x) = 0$. So there is no mistake, and $C = 0$. Hence, by the properties of Approximation 1, $p_c(x) = 0$. This implies $p(x) = 0$.

It only remains to show Claim 21 part 3:

$$\mathbb{E}_U[p(x)] = \mathbb{E}_U[C(x)] \pm \epsilon.$$

By part 1 of Claim 21,

$$\mathbb{E}_U[C(x) - p(x)] = \mathbb{E}_U[C(x) \vee E(x) - p(x)] \pm \epsilon.$$

We can show that this equals

$$\mathbb{E}_U\left[(C(x) \vee E(x) - p_c(x)(1 - p_E(x)))^2\right] \pm \epsilon$$

by the following argument: If $C(x) \vee E(x) = 1$ then $1 - p(x) = (1 - p_c(x)(1 - p_E(x)))^2$ by definition. If $C(x) \vee E(x) = 0$, then there is no mistake, and $C(x) = 0$. This implies that $p_c(x)(1 - p_E(x)) = p(x) = 0$.

Let us rewrite the above expression in terms of the expectation $\ell_2$ norm.

$$||C \vee E - p_c(1 - p_E)||_2^2.$$

Recall the triangle inequality, which states: $||u - v||_2 \leq ||u - w||_2 + ||w - v||_2$. Therefore, letting $w = p_c(1 - E)$ we have that the above quantity is

$$\leq (||p_c(1 - E) - p_c(1 - p_E)||_2 \quad + \quad ||p_c(1 - E) - C \vee E||_2)^2$$

$$\leq O(||p_c(1 - E) - p_c(1 - p_E)||_2^2 + ||p_c(1 - E) - C \vee E||_2^2).$$

To conclude, we will show that each of the above terms are $\leq \epsilon$. Note that

$$||p_c(1-E) - p_c(1-p_E)||_2^2 \leq \max_x |p_c(x)|^2 ||(1-E) - (1-p_E)||_2^2.$$

By Claim 22 part 1 and Approximation 2, this is at most

$$2^{\log(s/\epsilon)^{O(d)}} \cdot ||E - p_E||_2^2 \leq 2^{\log(s/\epsilon)^{O(d)}} \cdot \delta.$$

For this quantity to be at most $\epsilon$ we set $\delta = \epsilon \cdot 2^{-\log(s/\epsilon)^{O(d)}}$. Here we critically set the error in Approximation 2 much lower, to cancel the large values arising from Approximation 1. By Theorem 17, the polynomial arising from approximation 2 has degree $O(\log(s)^{d-1}\log(1/\delta)) = \log(s/\epsilon)^{O(d)}$.

Finally, let us bound the other term, $||p_c(1-E) - C \vee E||_2^2$. If $E(x) = 0$, then the distance is 0. If $E(x) = 1$, then the distance is $\leq 1$. Therefore, this term is at most $\Pr_U[E(x) = 1]$, which we know to be at most $\epsilon$. □

# 2 Small-bias distributions

**Definition 1.**[Small bias distributions] A distribution $D$ over $\{0,1\}^n$ has bias $\epsilon$ if no parity function can distinguish it from uniformly random strings with probability greater than $\epsilon$. More formally, we have:

$$\forall S \subseteq [n], S \neq \emptyset, \left| \mathbb{P}_{x \in D}\left[\bigoplus_{i \in S} x_i = 1\right] - 1/2 \right| \leq \epsilon.$$

In this definition, the 1/2 is simply the probability of a parity test being 1 or 0 over the uniform distribution. We also note that whether we change the definition to have the probability of the parity test being 0 or 1 doesn't matter. If a test has probability $1/2 + \epsilon$ of being equal to 1, then it has probability $1 - (1/2 + \epsilon) = 1/2 - \epsilon$ of being 0, so the bias is independent of this choice.

This can be viewed as a distribution which fools tests $T$ that are restricted to computing parity functions on a subset of bits.

Before we answer the important question of how to construct and efficiently sample from such a distribution, we will provide one interesting application of small bias sets to expander graphs.

**Theorem 2.**[Expander construction from a small bias set] Let $D$ be a distribution over $\{0,1\}^n$ with bias $\epsilon$. Define $G=(V,E)$ as the following graph:

$$V=\{0,1\}^n, E=\{(x,y)|x\oplus y\in \text{support}(D)\}.$$

Then, when we take the eigenvalues of the random walk matrix of $G$ in descending order $\lambda_1,\lambda_2,...\lambda_{2^n}$, we have that:

$$\max\{|\lambda_2|,|\lambda_{2^n}|\}\leq \epsilon.$$

Thus, small-bias sets yields expander graphs. Small-bias sets also turn out to be equivalent to constructing good linear codes. Although all these questions have been studied much before the definition of small-bias sets [NN90], the computational perspective has been quite useful, even in answering old questions. For example Ta-Shma used this perspective to construct better codes [Ta-17].

## 2.1 Constructions of small bias distributions

Just like our construction of bounded-wise independent distributions from the previous lecture, we will construct small-bias distributions using polynomials over finite fields.

**Theorem 3.**[Small bias construction] Let $\mathcal{F}$ be a finite field of size $2^\ell$, with elements represented as bit strings of length $\ell$. We define the generator $G:\mathcal{F}^2\to\{0,1\}^n$ as the following:

$$G(a,b)_i=\left\langle a^i,b\right\rangle=\sum_{j\leq\ell}(a^i)_jb_j \mod 2.$$

In this notation, a subscript of $j$ indicates taking the $j$th bit of the representation. Then the output of $G(a,b)$ over uniform $a$ and $b$ has bias $n/2^\ell$.

*Proof.* Consider some parity test induced by a subset $S\subset[n]$. Then when applied to the output of $G$, it simplifies as:

$$\sum_{i\in S}G(a,b)_i=\sum_{i\in S}\left\langle a^i,b\right\rangle=\left\langle\sum_{i\in S}a^i,b\right\rangle.$$

Note that $\sum_{i\in S} a^i$ is the evaluation of the polynomial $P_S(x) := \sum_{i\in S} x^i$ at the point $a$. We note that if $P_S(a) \neq 0$, then the value of $\langle P_S(a), b\rangle$ is equally likely to be 0 or 1 over the probability of a uniformly random $b$. This follows from the fact that the inner product of any non-zero bit string with a uniformly random bit string is equally likely to be 0 or 1. Hence in this case, our generator has no bias.

In the case where $P_S(a) = 0$, then the inner product will always be 0, independent of the value of $b$. In these situations, the bias is 1/2, but this is conditioned on the event that $P_S(a) = 0$.

We claim that this event has probability $\leq n/2^\ell$. Indeed, for non empty $S$, $P_S(a)$ is a polynomial of degree $\leq n$. Hence it has at most $n$ roots. But we are selecting $a$ from a field of size $2^\ell$. Hence the probability of picking one root is $\leq n/2^\ell$.

Hence overall the bias is at most $n/2^\ell$. □

To make use of the generator, we need to pick a specific $\ell$. Note that the seed length will be $|a| + |b| = 2\ell$. If we want to achieve bias $\epsilon$, then we must have $\ell = \log\left(\frac{n}{\epsilon}\right)$. Al the logarithms in this lecture are in base 2. This gives us a seed length of $2\log\left(\frac{n}{\epsilon}\right)$.

Small-bias are so important that a lot of attention has been devote to optimizing the constant "2" above. A lower bound of $\log n + (2 - o(1))\log(1/\epsilon)$ on the seed length was known. Ta-Shma recently [Ta-17] gave a nearly matching construction with seed length $\log n + (2 + o(1))\log(1/\epsilon)$.

We next give a sense of how to obtain different tradeoffs between $n$ and $\epsilon$ in the seed length. We specifically focus on getting a nearly optimal dependence on $n$, because the construction is a simple, interesting "derandomization" of the above one.

## 2.2 An improved small bias distribution via bootstrapping

We will show another construction of small bias distributions that achieves seed length $(1 + o(1))\log n + O(\log(1/\epsilon))$. It will make use of the previous construction and proof.

The intuition is the following: the only time we used that $b$ was uniform was in asserting that if $P_S(a) \neq 0$, then $\langle P_S(a), b\rangle$ is uniform. But we don't need $b$ to be uniform for that. What do we need from $b$? We need that it has small-bias!

Our new generator is $G(a, G'(a', b'))$ where $G$ and $G'$ are as before but with different parameters. For $G$, we pick $a$ of length $\ell = \log n/\epsilon$, whereas $G'$ just needs to be an $\epsilon$-biased generator on $\ell$ bits, which can be done as we just saw with $O(\log \ell/\epsilon)$ bits. This gives a seed length of $\log n + \log\log n + O(\log 1/\epsilon)$, as promised.

We can of course repeat the argument but the returns diminish.

## 2.3 Connecting small bias to k-wise independence

We will show that using our small bias generators, we can create distributions which are almost $k$-wise independent. That is, they are very close to a $k$-wise independent distribution in statistical distance, while having a substantially shorter seed length than what is required for $k$-wise independence. In particular, we will show two results:

- Small bias distributions are themselves close to $k$-wise independent.
- We can improve the parameters of the above by feeding a small bias distribution to the generator for $k$-wise independence from the previous lectures. This will improve the seed length of simply using a small bias distribution.

Before we can show these, we'll have to take a quick aside into some fundamental theorems of Fourier analysis of boolean functions.

## 2.4 Fourier analysis of boolean functions 101

Let $f : \{-1,1\}^n \rightarrow \{-1,1\}$. Here the switch between $\{0,1\}$ and $\{-1,1\}$ is common, but you can think of them as being isomorphic. One way to think of $f$ is as being a vector in $\{-1,1\}^{2^n}$. The $x$th entry of $f$ indicates the value of $f(x)$. If we let $\mathbf{1_S}$ be the indicator function returning 1 iff $x = S$, but once again written as a vector like $f$ is, then any function $f$ can be written over the basis of the $\mathbf{1_S}$ vectors, as:

$$f = \sum_S f(s)\mathbf{1_S}.$$

This is the "standard" basis.

Fourier analysis simply is a different basis in which to write functions, which is sometimes more useful. The basis functions are $\chi_S(x) : \{-1, 1\}^n \to \{-1, 1\} = \prod_{i \in S} x_i$. Then any boolean function $f$ can be expressed as:

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

where the $\hat{f}(S)$, called the "Fourier coefficients," can be derived as:

$$\hat{f}(S) = \mathbb{E}_{x \;\, U_n}\left[f(x)\chi_S(x)\right],$$

where the expectation is over uniformly random $x$.

**Claim 4.** For any function $f$ with range $\{-1, 1\}$, its Fourier coefficients satisfy:

$$\sum_{S \subseteq [n]} \hat{f}(S)^2 = 1.$$

*Proof.* We know that $\mathbb{E}[f(x)^2] = 1$, as squaring the function makes it 1. We can re-express this expectation as:

$$\mathbb{E}[f(x)f(x)] = \mathbb{E}\left[\sum_S \hat{f}(s)\chi_S(x) \cdot \sum_T \hat{f}(T)\chi_T(x)\right] = \mathbb{E}\left[\sum_{S,T} \hat{f}(s)\chi_S(x)\hat{f}(T)\chi_T(x)\right].$$

We make use of the following fact: if $S \neq T$, then $\mathbb{E}[\chi_S(x)\chi_T(x)] = \mathbb{E}[\chi_{S \oplus T}(x)] = 0$. If they equal each other, then their difference is the empty set and this function is 1.

Overall, this implies that the above expectation can be simply rewritten as:

$$\sum_{S=T} \hat{f}(S)\hat{f}(T) = \sum_S \hat{f}(S)^2.$$

Since we already decided that the expectation is 1, the claim follows. □

## 2.5 Small bias distributions are close to $k$-wise independent

Before we can prove our claim, we formally introduce what we mean for two distributions to be close. We use the most common definition of statistical difference, which we repeat here:

**Definition 5.** Let $D_1, D_2$ be two distributions over the same domain $H$. Then we denote their statistical distance $\mathrm{SD}(D_1, D_2)$, and sometimes written as $\Delta(D_1, D_2)$, as

$$\Delta(D_1, D_2) = \max_{T \subseteq H} \left| \mathcal{P}[D_1 \in T] - \mathcal{P}[D_2 \in T] \right|.$$

Note that the probabilities are with respect to the individual distributions $D_1$ and $D_2$. We may also say that $D_1$ is $\epsilon$-close to $D_2$ if $\Delta(D_1, D_2) \leq \epsilon$.

We can now show our result, which is known as **Vazirani's XOR Lemma**:

**Theorem 6.** If a distribution $D$ over $\{0,1\}^n$ has bias $\epsilon$, then $D$ is $\epsilon 2^{n/2}$ close to the uniform distribution.

*Proof.* Let $T$ be a test. To fit the above notation, we can think of $T$ as being defined as the set of inputs for which $T(x) = 1$. Then we want to bound:

$$|\mathbb{E}[T(D)] - \mathbb{E}[T(U)]|.$$

Expanding $T$ in Fourier basis we rewrite this as

$$|\mathbb{E}[\sum_S \hat{T}_S \chi_S(D)] - \mathbb{E}[\sum_S \hat{T}_S \chi_S(U)]| = |\sum_S \hat{T}_S \left( \mathbb{E}[\chi_S(D)] - \mathbb{E}[\chi_S(U)] \right)|.$$

We know that $\mathbb{E}_U[\chi_S(x)] = 0$ for all non empty $S$, and 1 when $S$ is the empty set. We also know that $\mathbb{E}_D[\chi_S(x)] \leq \epsilon$ for all non empty $S$, and is 1 when $S$ is the empty set. So the above can be bounded as:

$$\leq \sum_{S \neq \emptyset} |\hat{T}_S| |\mathbb{E}_D[\chi_S(x)] - \mathbb{E}_U[\chi_S(x)]| \leq \sum_S |\hat{T}_S| \epsilon = \epsilon \sum_S |\hat{T}_S|.$$

**Lemma 7.** $\sum_S |\hat{T}_S| \leq 2^{n/2}$

*Proof.* By Cauchy Schwartz:

$$\sum |\hat{T}_S| \leq 2^{n/2} \sqrt{\sum \hat{T}_S^{\,2}} \leq 2^{n/2}$$

Where the last simplification follows from Claim 4. □

Using the above lemma completes the upper bound and the proof of the theorem. □

**Corollary 8.** Any $k$ bits of an $\epsilon$ biased distribution are $\epsilon 2^{k/2}$ close to uniform.

Using the corollary above, we see that we can get $\epsilon$ close to a $k$-wise independent distribution (in the sense of the corollary) by taking a small bias distribution with $\epsilon' = \epsilon/2^{k/2}$. This requires seed length $\ell = O(\log(n/\epsilon') = O(\log(2^{k/2}n/\epsilon) = O(\log(n) + k + \log(1/\epsilon))$. Recall that for exact $k$-wise we required seed length $k \log n$.

## 2.6 An improved construction

**Theorem 9.** Let $G : \{0,1\}^{k \log n} \to \{0,1\}^n$ be the generator previously described that samples a $k$-wise independent distribution (or any linear $G$). If we replace the input to $G$ with a small bias distribution of $\epsilon' = \epsilon/2^k$, then the output of $G$ is $\epsilon$-close to being $k$-wise independent.

*Proof.* Consider any parity test $S$ on $k$ bits on the output of $G$. It can be shown that $G$ is a linear map, that is, $G$ simply takes its seed and it multiplies it by a matrix over the field GF(2) with two elements. Hence, $S$ corresponds to a test $S'$ on the input of $G$, on possibly many bits. The test $S'$ is not empty because $G$ is $k$-wise independent. Since we fool $S'$ with error $\epsilon'$, we also fool $S$ with error $\epsilon$, and the theorem follows by Vazirani's XOR lemma. □

Using the seed lengths we saw we get the following.

**Corollary 10.** There is a generator for almost $k$-wise independent distributions with seed length $O(\log \log n + \log(1/\epsilon) + k)$.

# 3 Tribes Functions and the GMRTV Generator

We now move to a more recent result. Consider the Tribes function, which is a read-once CNF on $k \cdot w$ bits, given by the And of $k$ terms, each on $w$ bits. You should think of $n = k \cdot w$ where $w \approx \log n$ and $k \approx n/\log n$.

We'd like a generator for this class with seed length $O(\log n/\epsilon)$. This is still open! (This is just a single function, for which a generator is trivial, but one can make this challenge precise for example by asking to fool the Tribes function for any possible negation of the input variables. These are $2^n$ tests and a generator with seed length $O(\log n/\epsilon)$ is unknown.)

The result we saw earlier about fooling And gives a generator with seed length $O(\log n)$, however the dependence on $\epsilon$ is poor. Achieving a good dependence on $\epsilon$ has proved to be a challenge. We now describe a recent generator [GMR$^+$12] which gives seed length $O(\log n/\epsilon)(\log\log n)^{O(1)}$. This is incomparable with the previous $O(\log n)$, and in particular the dependence on $n$ is always suboptimal. However, when $\epsilon = 1/n$ the generator [GMR$^+$12] gives seed length $O(\log n)\log\log n$ which is better than previously available constructions.

The high-level technique for doing this is based on iteratively restricting variables, and goes back about 30 years [AW89]. This technique seems to have been abandoned for a while, possibly due to the spectacular successes of Nisan [Nis91, Nis92]. It was revived in [GMR$^+$12] (see also [GLS12]) with an emphasis on a good dependence on $\epsilon$.

A main tool is this claim, showing that small-bias distributions fool products of functions with small variance. Critically, we work with non-boolean functions (which later will be certain averages of boolean functions).

**Claim 1.** *Let $f_1, f_2, ..., f_k : \{0,1\}^w \to [0,1]$ be a series of boolean functions. Further, let $D = (v_1, v_2, ..., v_k)$ be an $\epsilon$-biased distribution over $wk$ bits, where each $v_i$ is $w$ bits long. Then*

$$\mathbb{E}_D[\prod_i f_i(v_i)] - \prod_i \mathbb{E}_U[f_i(U)] \le \left(\sum_i \mathrm{var}(f_i)\right)^d + (k2^w)^d\epsilon,$$

*where $\mathrm{var}(f) := \mathbb{E}[f^2] - \mathbb{E}^2[f]$ is variance of $f$ with respect to the uniform distribution.*

This claim has emerged from a series of works, and this statement is from a work in progress with Chin Ho Lee. For intuition, note that constant functions have variance 0, in which case the claim gives good bounds (and indeed any distribution fools constant functions). By contrast, for balanced functions the variance is constant, and the sum of the variances is about $k$, and the claim gives nothing. Indeed, you can write Inner Product as a product of nearly balanced functions, and it is known that small-bias does not fool it. For this claim to kick in, we need each variance to be at most $1/k$.

In the tribes function, the And fucntions have variance $2^{-w}$, and the sum of the variances is about 1 and the claim gives nothing. However, if you perturb the Ands with a little noise, the variance drops polynomially, and the claim is useful.

**Claim 2.** Let $f$ be the AND function on $w$ bits. Rewrite it as $f(x, y)$, where $|x| = |y| = w/2$. That is, we partition the input into two sets. Define $g(x)$ as:

$$g(x) = \mathbb{E}_y[f(x, y)],$$

where $y$ is uniform. Then $\mathrm{var}(g) = \Theta(2^{-3w/2})$.

*Proof.*

$$\mathrm{var}(g) = \mathbb{E}[g(x)^2] - (\mathbb{E}[g(x)])^2 = \mathbb{E}_x[\mathbb{E}_y[f(x, y)]^2] - (\mathbb{E}_x[\mathbb{E}_y[f(x, y)]])^2 .$$

We know that $(\mathbb{E}_x[\mathbb{E}_y[f(x, y)]])$ is simply the expected value of $f$, and since $f$ is the AND function, this is $2^{-w}$, so the right term is $2^{-2w}$.

We reexpress the left term as $\mathbb{E}_{x,y,y'}[f(x, y)f(x, y')]$. But we note that this product is 1 iff $x = y = y' = \mathbf{1}$. The probability of this happening is $(2^{-w/2})^3 = 2^{-3w/2}$.

Thus the final difference is $2^{-3w/2}(1 - 2^{-w/2}) = \Theta(2^{-3w/2})$. □

We'll actually apply this claim to the Or function, which has the same variance as And by De Morgan's laws.

We now present the main inductive step to fool tribes.

**Claim 3.** Let $f$ be the tribes function, where the first $t \leq w$ bits of each of the terms are fixed. Let $w' = w - t$ be the free bits per term, and $k' \leq k$ the number of terms that are non-constant (some term may have become 0 after fixing the bits).

Reexpress $f$ as $f(x, y) = \bigwedge_{k'} (\bigvee(x_i, y_i))$, where each term's input bits are split in half, so $|x_i| = |y_i| = w'/2$.

Let $D$ be a small bias distribution with bias $\epsilon^c$ (for a big enough $c$ to be set later). Then

$$\left|\mathbb{E}_{(x,y)\in U^2}[f(x, y)] - \mathbb{E}_{(x,y)\in (D,U)}[f(x, y)]\right| \leq \epsilon.$$

That is, if we replace half of the free bits with a small bias distribution, then the resulting expectation of the function only changes by a small amount.

To get the generator from this claim, we repeatedly apply Claim 3, replacing half of the bits of the input with another small bias distribution. We repeat this until we have a small enough remaining amount of free bits that replacing all of them with a small bias distribution causes an insignificant change in the expectation of the output.

At each step, $w$ is cut in half, so the required number of repetitions to reduce $w'$ to constant is $R = \log(w) = \log\log(n)$. Actually, as explained below, we'll stop when $w = c' \log\log 1/\epsilon$ for a suitable constant $c'$ (this arises from the error bound in the claim above, and we).

After each replacement, we incur an error of $\epsilon$, and then we incur the final error from replacing all bits with a small bias distribution. This final error is negligible by a result which we haven't seen, but which is close in spirit to the proof we saw that bounded independence fools AND.

The total accumulated error is then $\epsilon' = \epsilon \log\log(n)$. If we wish to achieve a specific error $\epsilon$, we can run each small bias generator with $\epsilon / \log\log(n)$.

At each iteration, our small bias distribution requires $O(\log(n/\epsilon))$ bits, so our final seed length is $O(\log(n/\epsilon))\text{poly}\log\log(n)$.

*Proof of Claim 3.* Define $g_i(x) = \mathbb{E}_y[\bigvee_i(x_i, y_i)]$, and rewrite our target expression as:

$$\mathbb{E}_{x\in U}\left[\prod g_i(x_i)\right] - \mathbb{E}_{x\in D}\left[\prod g_i(x_i)\right].$$

This is in the form of Claim 1. We also note that from Claim 2 that $\text{var}(g_i) = 2^{-3w'/2}$.

We further assume that $k' \le 2^{w'}\log(1/\epsilon)$. For if this is not true, then the expectation over the first $2^{w'}\log(1/\epsilon)$ terms is $\le \epsilon$, because of the calculation

$$(1 - 2^{-w'})^{2^{w'}\log(1/\epsilon)} \le \epsilon.$$

Then we can reason as in the proof that bounded independence fools AND (i.e., we can run the argument just on the first $2^{w'}\log(1/\epsilon)$ terms to show that the products are close, and then use the fact that it is small under uniform, and the fact that adding terms only decreases the probability under any distribution).

Under the assumption, we can bound the sum of the variances of $g$ as:

$$\sum \text{var}(g_i) \le k' 2^{-3w'/2} \le 2^{-\Omega(w')}\log(1/\epsilon).$$

If we assume that $w' \ge c \log\log(1/\epsilon)$ then this sum is $\le 2^{-\Omega(w')}$.

We can then plug this into the bound from Claim 1 to get

$$(2^{-\Omega(w')})^d + (k2^{w'})^d \epsilon^c = 2^{-\Omega(dw')} + 2^{O(dw')}\epsilon^c.$$

Now we set $d$ so that $\Omega(dw') = \log(1/\epsilon) + 1$, and the bound becomes:

$$\epsilon/2 + (1/\epsilon)^{O(1)}\epsilon^c \leq \epsilon.$$

By making $c$ large enough the claim is proved. □

In the original paper, they apply these ideas to read-once CNF formulas. Interestingly, this extension is more complicated and uses additional ideas. Roughly, the progress measure is going to be number of terms in the CNF (as opposed to the width). A CNF is broken up into a small number of Tribes functions, the above argument is applied to each Tribe, and then they are put together using a general fact that they prove, that if $f$ and $g$ are fooled by small-bias then also $f \wedge g$ on disjoint inputs is fooled by small-bias.

In these lectures, we introduce $k$-wise indistinguishability and link this notion to the approximate degree of a function. Then, we study the approximate degree of some functions, namely, the AND function and the AND-OR function. For the latter function we begin to see a proof that is different (either in substance or language) from the proofs in the literature. We begin with some LaTeXtips.

# 4 Bounded indistinguishability

We studied previously the following questions:

- What is the minimum $k$ such that any $k$-wise independent distribution $P$ over $\{0,1\}^n$ fools AC$^0$ (*i.e.* $\mathbb{E}C(P) \approx \mathbb{E}C(U)$ for all *poly*$(n)$-size circuits $C$ with constant depth)?

  We saw that $k = \log^{\mathcal{O}(d)}(s/\epsilon)$ is enough.

- What is the minimum $k$ such that $P$ fools the AND function?

  Taking $k = \mathcal{O}(1)$ for $\epsilon = \mathcal{O}(1)$ suffices (more precisely we saw that $k$-wise independence fools the AND function with $\epsilon = 2^{-\Omega(k)}$).

Consider now $P$ and $Q$ two distributions over $\{0,1\}^n$ that are *$k$-wise indistinguishable*, that is, any projection over $k$ bits of $P$ and $Q$ have the same distribution. We can ask similar questions:

- What is the minimum $k$ such that $\mathrm{AC}^0$ cannot distinguish $P$ and $Q$ (*i.e.* $\mathbb{E}C(P) \approx \mathbb{E}C(Q)$ for all $poly(n)$-size circuits $C$ with constant depth)?

  It turns out this requires $k \geq n^{1-o(1)}$: there are some distributions that are almost always distinguishable in this regime. (Whether $k = \Omega(n)$ is necessary or not is an open question.)

  Also, $k = n\left(1 - \frac{1}{polylog(n)}\right)$ suffices to fool $\mathrm{AC}^0$ (in which case $\epsilon$ is essentially exponentially small).

- What is the minimum $k$ such that the AND function (on $n$ bits) cannot distinguish $P$ and $Q$?

  It turns out that $k = \Theta(\sqrt{n})$ is necessary and sufficient. More precisely:

  - There exists some $P, Q$ over $\{0,1\}^n$ that are $c\sqrt{n}$-wise indistinguishable for some constant $c$, but such that:

    $$\left|\Pr_P[AND(P) = 1] - \Pr_Q[AND(Q) = 1]\right| \geq 0.99\,;$$

  - For all $P, Q$ that are $c'\sqrt{n}$-wise indistinguishable for some bigger constant $c'$, we have:

    $$\left|\Pr_P[AND(P) = 1] - \Pr_Q[AND(Q) = 1]\right| \leq 0.01\,.$$

## 4.1 Duality.

Those question are actually equivalent to ones related about approximation by real-valued polynomials:

**Theorem 1.** Let $f : \{0,1\}^n \to \{0,1\}$ be a function, and $k$ an integer. Then:

$$\max_{P,Q\ k\text{-wise indist.}} |\mathbb{E}f(P) - \mathbb{E}f(Q)| = \min\{\, \epsilon \mid \exists g \in \mathbb{R}_k[X] : \forall x, |f(x) - g(x)| \leq \epsilon\}.$$

Here $\mathbb{R}_k[X]$ denotes degree-$k$ real polynomials. We will denote the right-hand side $\epsilon_k(f)$.

Some examples:

- $f = 1$: then $\mathbb{E}f(P) = 1$ for all distribution $P$, so that both sides of the equality are 0.

- $f(x) = \sum_i x_i \bmod 2$ the parity function on $n$ bits.

  Then for $k = n - 1$, the left-hand side is at least $1/2$: take $P$ to be uniform; and $Q$ to be uniform on $n-1$ bits, defining the $n$th bit to be $Q_n = \sum_{i<n} Q_i \bmod 2$ to be the parity of the first $n-1$ bits. Then $\mathbb{E}f(P) = 1/2$ but $\mathbb{E}f(Q) = 0$.

  Furthermore, we have:

  **Claim 2.** $\epsilon_{n-1}(\text{Parity}) \geq 1/2$.

  *Proof.* Suppose by contradiction that some polynomial $g$ has degree $k$ and approximates Parity by $\epsilon < 1/2$.

  The key ingredient is to *symmetrize* a polynomial $p$, by letting

$$p^{sym}(x) := \frac{1}{n!} \sum_{\pi \in \mathfrak{S}_n} f(\pi x),$$

  where $\pi$ ranges over permutations. Note that $p^{sym}(x)$ only depends on $\|x\| = \sum_i x_i$.

  Now we claim that there is a *univariate* polynomial $p'$ also of degree $k$ such that

$$p'(\sum x_i) = p^{sym}(x_1, x_2, \ldots, x_n)$$

  for every $x$.

  To illustrate, let $M$ be a monomial of $p$. For instance if $M = X_1$, then $p'(i) = i/n$, where $i$ is the Hamming weight of the input. (For this we think of the input as being $\in \{0,1\}$. Similar calculations can be done for $\in \{-1,-1\}$.)

  If $M = X_1 X_2$, then $p'(i) = \frac{i}{n} \cdot \frac{i-1}{n}$ which is quadratic in $i$.

  And so on.

  More generally $p^{sym}(X_1, \ldots, X_n)$ is a symmetric polynomial. As $\{(\sum_j X_j)^\ell\}_{\ell \leq k}$ form a basis of symmetric polynomials of degree $k$, $p^{sym}$ can be written as a linear combination in this basis. Now note that $\{(\sum_j X_j)^\ell(x)\}_{\ell \leq k}$ only depends on $\|x\|$; substituting $i = \sum_j X_j$ gives that $p'$ is of degree $\leq k$ in $i$.

  (Note that the degree of $p'$ can be strictly less than the degree of $p$ (*e.g.* for $p(X_1, X_2) = X_1 - X_2$: we have $p^{sym} = p' = 0$).)

Then, applying symmetrization on $g$, if $g$ is a real polynomial $\epsilon$-close to Parity (in $\ell_\infty$ norm), then $g'$ is also $\epsilon$-close to Parity' (as a convex combination of close values).

Finally, remark that for every integer $k \in \{0, \dots, \lfloor n/2 \rfloor\}$, we have: $\text{Parity}'(2k) = 0$ and $\text{Parity}'(2k+1) = 1$. In particular, as $\epsilon < 1/2$, $g' - 1/2$ must have at least $n$ zeroes, and must therefore be zero, which is a contradiction.

□

We will now focus on proving the theorem.

Note that one direction is easy: if a function $f$ is closely approximated by a polynomial $g$ of degree $k$, it cannot distinguish two $k$-wise indistinguishable distributions $P$ and $Q$:

$$\begin{aligned} \mathbb{E}[f(P)] &= \mathbb{E}[g(P)] \pm \epsilon \\ &\overset{(*)}{=} \mathbb{E}[g(Q)] \pm \epsilon \\ &= \mathbb{E}[f(Q)] \pm 2\epsilon\,, \end{aligned}$$

where $(*)$ comes from the fact that $P$ and $Q$ are $k$-wise indistinguishable.

The general proof goes by a Linear Programming Duality (aka finite-dimensional Hahn-Banach theorem, min-max, etc.). This states that:

If $A \in \mathbb{R}^{n \times m}$, $x \in \mathbb{R}^m$, $b \in \mathbb{R}^n$ and $c \in \mathbb{R}^m$, then:

$$\begin{array}{lrcl|c|lr} & \min\langle c, x\rangle & = & \sum_{i \leq m} c_i x_i & & & \max\langle b, y\rangle \\ \text{subject to:} & Ax & = & b & = & \text{subject to:} & A^T y \leq c \\ & x & \geq & 0 & & & \end{array}$$

We can now prove the theorem:

*Proof.* The proof will consist in rewriting the sides of the equality in the theorem as outputs of a Linear Program. Let us focus on the left side of the equality: $\max_{P,Q\, k\text{-wise indist.}} |\mathbb{E}f(P) - \mathbb{E}f(Q)|$.

We will introduce $2^{n+1}$ variables, namely $P_x$ and $Q_x$ for every $x \in \{0,1\}^n$, which will represent $\Pr[D = x]$ for $D = P, Q$.

We will also use the following, which can be proved similarly to the Vazirani XOR Lemma:

**Claim 3.** Two distributions $P$ and $Q$ are $k$-wise indistinguishable if and only if: $\forall S \subseteq \{1,\ldots,n\}$ with $|S| \leq k$, $\sum_x P_x\chi_S(x) - \sum_x Q_x\chi_S(x) = 0$, where $\chi_S(X) = \prod_S X_i$ is the Fourier basis of boolean functions.

The quantity $\max_{P,Q\,k\text{-wise indist.}} |\mathbb{E}f(P) - \mathbb{E}f(Q)|$ can then be rewritten:

$$-\min \textstyle\sum_x P_x f(x) - \sum_x Q_x f(x)$$

$$\begin{array}{lr}
\text{subject to:} & \sum_x P_x \quad = 1 \\
 & \sum_x Q_x \quad = 1 \\
 & \forall S \subseteq \{1,\ldots,n\} \text{ s.t. } |S| \leq k, \sum_x (P_x - Q_x)\chi_S(x) \quad = 0
\end{array}$$

Following the syntax of LP Duality stated above, we have:

$c^T = \overbrace{\cdots f(x)\cdots}^{2^n}\overbrace{\cdots - f(x)\cdots}^{2^n} \in \mathbb{R}^{2n}$, (where $x$ goes over $\{0,1\}^n$),

$x^T = \overbrace{\cdots P_x\cdots}^{2^n}\overbrace{\cdots Q_x\cdots}^{2^n} \in \mathbb{R}^{2n}$,

$b^T = 1\,1\,\overbrace{0\cdots 0}^{\#S}$,

$$A = \left(\begin{array}{cc}
\overbrace{1\cdots\cdots 1}^{2^n} & \overbrace{0\cdots\cdots 0}^{2^n} \\
0\cdots\cdots 0 & 1\cdots\cdots 1 \\
\cdots\cdots & \cdots\cdots \\
\vdots\cdots\cdots\vdots & \vdots\cdots\cdots\vdots \\
\cdots\chi_S(x)\cdots & \cdots - \chi_S(x)\cdots \\
\vdots\cdots\cdots\vdots & \vdots\cdots\cdots\vdots \\
\cdots\cdots & \cdots\cdots
\end{array}\right),$$

where the rows of $A$ except the first two correspond to some $S \subseteq \{1,\ldots,n\}$ such that $|S| \leq k$.

We apply LP duality. We shall denote the new set of variables by

$y^T = d\,d'\,\overbrace{\cdots d_S\cdots}^{\#S}$.

We have the following program:

$$-\max d + d'$$

$$\begin{array}{llr}
\text{subject to:} & \forall x, d + \sum_x d_S\chi_S(x) & \leq f(x) \\
 & \forall x, d' - \sum_x d_S\chi_S(x) & \leq -f(x)
\end{array}$$

Writing $d' = -d - \epsilon$, the objective becomes to minimize $\epsilon$, while the second set of constraints can be rewritten:

$$\forall x, d + \epsilon + \sum_S d_S \chi_S(x) \geq f(x)\,.$$

The expression $d + \sum_S d_S \chi_S(X)$ is an arbitrary degree-$k$ polynomial which we denote by $g(X)$. So our constrains become

$$\begin{aligned} g(x) &\leq f(x) \\ g(x) + \epsilon &\geq f(x). \end{aligned}$$

Where $g$ ranges over all degree-$k$ polynomials, and we are trying to minimize $\epsilon$. Because $g$ is always below $f$, but when you add $\epsilon$ it becomes bigger, $g$ is always within $\epsilon$ of $f$. □

## 4.2 Approximate Degree of AND.

Let us now study the AND function on $n$ bits. Let us denote $d_\epsilon(f)$ the minimal degree of a polynomial approximating $f$ with error $\epsilon$.

We will show that $d_{1/3}(\text{AND}) = \Theta(\sqrt{n})$.

Let us first show the upper bound:

**Claim 4.** We have:

$$d_{1/3}(\text{AND}) = \mathcal{O}(\sqrt{n}).$$

To prove this claim, we will consider a special family of polynomials:

**Definition 5. (Chebychev polynomials of the first kind.)**

The Chebychev polynomials (of the first kind) are a family $\{T_k\}_{k\in\mathbb{N}}$ of polynomials defined inductively as:

- $T_0(X) := 1$,
- $T_1(X) := X$,
- $\forall k \geq 1, T_{k+1}(X) := 2XT_k - T_{k-1}$.

Those polynomials satisfy some useful properties:

1. $\forall x \in [-1, 1], T_k(x) = \cos(k \arccos(x))$,

2. $\forall x \in [-1,1], \forall k, |T_k(x)| \leq 1\,,$

3. $\forall x$ such that $|x| \geq 1, |T'_k(x)| \geq k^2\,,$

4. $\forall k, T_k(1) = 1\,.$

Property 2 follows from 1, and property 4 follows from a direct induction. For a nice picture of these polynomials you should have come to class (or I guess you can check wikipedia). We can now prove our upper bound:

*Proof.* Proof of Claim:

We construct a univariate polynomial $p : \{0, 1, \dots, n\} \to \mathbb{R}$ such that:

- $\deg p = \mathcal{O}(\sqrt{n})$;
- $\forall i < n, |P(i)| \leq 1/3$;
- $|P(n) - 1| \leq 1/3$.

In other words, $p$ will be close to 0 on $[0, n-1]$, and close to 1 on $n$. Then, we can naturally define the polynomial for the AND function on $n$ bits to be $q(X_1, \dots, X_n) := p(\sum_i X_i)$, which also has degree $\mathcal{O}(\sqrt{n})$. Indeed, we want $q$ to be close to 0 if $X$ has Hamming weight less than $n$, while being close to 1 on $X$ of Hamming weight $n$ (by definition of AND). This will conclude the proof.

Let us define $p$ as follows:

$$\forall i \leq n, \quad p(i) := \frac{T_k\left(\frac{i}{n-1}\right)}{T_k\left(\frac{n}{n-1}\right)}.$$

Intuitively, this uses the fact that Chebychev polynomials are bounded in $[-1, 1]$ (Property 2.) and then increase very fast (Property 3.).

More precisely, we have:

- $p(n) = 1$ by construction;
- for $i < n$, we have:

  $T_k\left(\frac{i}{n-1}\right) \leq 1$ by Property 2.;

  $T_k\left(\frac{n}{n-1}\right) = T_k\left(1 + \frac{1}{n-1}\right) \geq 1 + \frac{k^2}{n-1}$ by Property 3. and 4., and therefore for some $k = \mathcal{O}(\sqrt{n})$, we have: $T_k\left(\frac{n}{n-1}\right) \geq 3$.

□

Let us now prove the corresponding lower bound:

**Claim 6.** We have:

$$d_{1/3}(\text{AND}) = \Omega(\sqrt{n}).$$

*Proof.* Let $p$ be a polynomial that approximates the AND function with error $1/3$. Consider the univariate symmetrization $p'$ of $p$.

We have the following result from approximation theory:

**Theorem 7.** Let $q$ be a real univariate polynomial such that:

1. $\forall i \in \{0, \ldots, n\}, |q(i)| \leq \mathcal{O}(1)$;
2. $q'(x) \geq \Omega(1)$ for some $x \in [0, n]$.
   Then $\deg q = \Omega(\sqrt{n})$.

To prove our claim, it is therefore sufficient to check that $p'$ satisfies conditions 1. and 2., as we saw that $\deg p \geq \deg p'$:

1. We have: $\forall i \in \{0, \ldots, n\}, |p'(i)| \leq 1 + 1/3$ by assumption on $p$;
2. We have $p'(n-1) \leq 1/3$ and $p'(n) \geq 2/3$ (by assumption), so that the mean value theorem gives some $x$ such that $p'(x) \geq \Omega(1)$.

This concludes the proof. □

## 4.3 Approximate Degree of AND-OR.

Consider the AND function on $R$ bits and the OR function on $N$ bits. Let AND-OR: $\{0,1\}^{R\times N} \to \{0,1\}$ be their composition (which outputs the AND of the $R$ outputs of the $OR$ function on $N$-bits (disjoint) blocks).

It is known that $d_{1/3}(\text{AND-OR}) = \Theta(\sqrt{RN})$. To prove the upper bound, we will need a technique to compose approximating polynomials which we will discuss later.

Now we focus on the lower bound. This lower bound was recently proved independently by Sherstov and by Bun and Thaler. We present a proof that is different (either in substance or in language) and which we find more intuitive. Our proof replaces the “dual block method” with the following lemma.

**Lemma 8.** Suppose that

distributions $A^0, A^1$ over $\{0,1\}^{n_A}$ are $k_A$-wise indistinguishable distributions; and

distributions $B^0, B^1$ over $\{0,1\}^{n_B}$ are $k_B$-wise indistinguishable distributions.

Define $C^0, C^1$ over $\{0,1\}^{n_A \cdot n_B}$ as follows: $C^b$: draw a sample $x \in \{0,1\}^{n_A}$ from $A^b$, and replace each bit $x_i$ by a sample of $B^{x_i}$ (independently).

Then $C^0$ and $C^1$ are $k_A \cdot k_B$-wise indistinguishable.

*Proof.* Consider any set $S \subseteq \{1, \ldots, n_A \cdot n_B\}$ of $k_A \cdot k_B$ bit positions; let us show that they have the same distribution in $C^0$ and $C^1$.

View the $n_A \cdot n_B$ as $n_A$ *blocks* of $n_B$ bits. Call a block $K$ of $n_B$ bits *heavy* if $|S \cap K| \geq k_B$; call the other blocks *light*.

There are at most $k_A$ heavy blocks by assumption, so that the distribution of the (entire) heavy blocks are the same in $C^0$ and $C^1$ by $k_A$-wise indistinguishability of $A^0$ and $A^1$.

Furthermore, conditioned on any outcome for the $A^b$ samples in $C^b$, the light blocks have the same distribution in both $C^0$ and $C^1$ by $k_B$-wise indistinguishability of $B^0$ and $B^1$.

Therefore $C^0$ and $C^1$ are $k_A \cdot k_B$-wise indistinguishable. □

## 4.4 Lower Bound of $d_{1/3}$(AND-OR)

To prove the lower bound on the approximate degree of the AND-OR function, it remains to see that AND-OR can distinguish well the distributions $C^0$ and $C^1$. For this, we begin with observing that we can assume without loss of generality that the distributions have disjoint supports.

**Claim 9.** For any function $f$, and for any $k$-wise indistinguishable distributions $A^0$ and $A^1$, if $f$ can distinguish $A^0$ and $A^1$ with probability $\epsilon$ then there are distributions $B^0$ and $B^1$ with the same properties ($k$-wise indistinguishability yet distinguishable by $f$) and also with disjoint supports. (By disjoint support we mean for any $x$ either $\Pr[B^0 = x] = 0$ or $\Pr[B^1 = x] = 0$.)

*Proof.* Let distribution $C$ be the “common part” of $A^0$ and $A^1$. That is to say, we define $C$ such that $\Pr[C = x] := \min\{\Pr[A^0 = x], \Pr[A^1 = x]\}$

multiplied by some constant that normalize $C$ into a distribution.

Then we can write $A^0$ and $A^1$ as

$$\begin{aligned} A^0 &= pC + (1-p)B^0\,, \\ A^1 &= pC + (1-p)B^1\,, \end{aligned}$$

where $p \in [0,1]$, $B^0$ and $B^1$ are two distributions. Clearly $B^0$ and $B^1$ have disjoint supports.

Then we have

$$\begin{aligned} \mathbb{E}[f(A^0)] - \mathbb{E}[f(A^1)] &= p\mathbb{E}[f(C)] + (1-p)\mathbb{E}[f(B^0)] \\ &\quad - p\mathbb{E}[f(C)] - (1-p)\mathbb{E}[f(B^1)] \\ &= (1-p)\big(\mathbb{E}[f(B^0)] - \mathbb{E}[f(B^1)]\big) \\ &\le \mathbb{E}[f(B^0)] - \mathbb{E}[f(B^1)]\,. \end{aligned}$$

Therefore if $f$ can distinguish $A^0$ and $A^1$ with probability $\epsilon$ then it can also distinguish $B^0$ and $B^1$ with such probability.

Similarly, for all $S \neq \varnothing$ such that $|S| \le k$, we have

$$0 = \mathbb{E}[\chi_S(A^0)] - \mathbb{E}[\chi_S(A^1)] = (1-p)\big(\mathbb{E}[\chi_S(B^0)] - \mathbb{E}[\chi_S(B^1)]\big) = 0\,.$$

Hence, $B^0$ and $B^1$ are $k$-wise indistinguishable. □

Equipped with the above lemma and claim, we can finally prove the following lower bound on the approximate degree of AND-OR.

**Theorem 10.** $d_{1/3}(\text{AND-OR}) = \Omega(\sqrt{RN})$.

*Proof.* Let $A^0, A^1$ be $\Omega(\sqrt{R})$-wise indistinguishable distributions for AND with advantage 0.99, i.e. $\Pr[\text{AND}(A^1) = 1] > \Pr[\text{AND}(A^0) = 1] + 0.99$. Let $B^0, B^1$ be $\Omega(\sqrt{N})$-wise indistinguishable distributions for OR with advantage 0.99. By the above claim, we can assume that $A^0, A^1$ have disjoint supports, and the same for $B^0, B^1$. Compose them by the lemma, getting $\Omega(\sqrt{RN})$-wise indistinguishable distributions $C^0, C^1$. We now show that AND-OR can distinguish $C^0, C^1$:

- $C_0$: First sample $A^0$. As there exists a unique $x = 1^R$ such that $\text{AND}(x) = 1$, $\Pr[A^1 = 1^R] > 0$. Thus by disjointness of support $\Pr[A^0 = 1^R] = 0$. Therefore when sampling $A^0$ we always get a string with at least one "0". But then "0" is replaced with sample from $B^0$. We have $\Pr[B^0 = 0^N] \ge 0.99$, and when $B^0 = 0^N$, AND-OR$= 0$.

- $C_1$: First sample $A^1$, and we know that $A^1 = 1^R$ with probability at least 0.99. Each bit "1" is replaced by a sample from $B^1$, and we know that $\Pr[B^1 = 0^N] = 0$ by disjointness of support. Then AND-OR= 1.

Therefore we have $d_{1/3}(\text{AND-OR}) = \Omega(\sqrt{RN})$. □

## 4.5 Lower Bound of $d_{1/3}$(SURJ)

In this subsection we discuss the approximate degree of the surjectivity function. This function is defined as follows.

**Definition 11.** The surjectivity function SURJ: $\left(\{0,1\}^{\log R}\right)^N \to \{0,1\}$, which takes input $(x_1, \ldots, x_N)$ where $x_i \in [R]$ for all $i$, has value 1 if and only if $\forall j \in [R], \exists i \colon x_i = j$.

First, some history. Aaronson first proved that the approximate degree of SURJ and other functions on $n$ bits including "the collision problem" is $n^{\Omega(1)}$. This was motivated by an application in quantum computing. Before this result, even a lower bound of $\omega(1)$ had not been known. Later Shi improved the lower bound to $n^{2/3}$, see [AS04]. The instructor believes that the quantum framework may have blocked some people from studying this problem, though it may have very well attracted others. Recently Bun and Thaler [BT17] reproved the $n^{2/3}$ lower bound, but in a quantum-free paper, and introducing some different intuition. Soon after, together with Kothari, they proved [BKT17] that the approximate degree of SURJ is $\Theta(n^{3/4})$.

We shall now prove the $\Omega(n^{3/4})$ lower bound, though one piece is only sketched. Again we present some things in a different way from the papers.

For the proof, we consider the AND-OR function under the promise that the Hamming weight of the $RN$ input bits is at most $N$. Call the approximate degree of AND-OR under this promise $d_{1/3}^{\leq N}(\text{AND-OR})$. Then we can prove the following theorems.

**Theorem 12.** $d_{1/3}(\text{SURJ}) \geq d_{1/3}^{\leq N}(\text{AND-OR})$.

**Theorem 13.** $d_{1/3}^{\leq N}(\text{AND-OR}) \geq \Omega(N^{3/4})$ for some suitable $R = \Theta(N)$.

In our settings, we consider $R = \Theta(N)$. Theorem 12 shows surprisingly that we can somehow "shrink" $\Theta(N^2)$ bits of input into $N \log N$ bits while maintaining the approximate degree of the function, under some promise. Without this promise, we just showed in the last subsection that the approximate degree of AND-OR is $\Omega(N)$ instead of $\Omega(N^{3/4})$ as in Theorem 13.

*Proof of Theorem 12.* Define an $N \times R$ matrix $Y$ s.t. the 0/1 variable $y_{ij}$ is the entry in the $i$-th row $j$-th column, and $y_{ij} = 1$ iff $x_i = j$. We can prove this theorem in following steps:

1. $d_{1/3}(\mathrm{SURJ}(\overline{x})) \geq d_{1/3}(\text{AND-OR}(\overline{y}))$ under the promise that each row has weight 1;

2. let $z_j$ be the sum of the $j$-th column, then $d_{1/3}(\text{AND-OR}(\overline{y}))$ under the promise that each row has weight 1, is at least $d_{1/3}(\text{AND-OR}(\overline{z}))$ under the promise that $\sum_j z_j = N$;

3. $d_{1/3}(\text{AND-OR}(\overline{z}))$ under the promise that $\sum_j z_j = N$, is at least $d_{1/3}^{=N}$(AND-OR($\overline{y}$));

4. we can change "$= N$" into "$\leq N$".

Now we prove this theorem step by step.

1. Let $P(x_1, \ldots, x_N)$ be a polynomial for SURJ, where $x_i = (x_i)_1, \ldots, (x_i)_{\log R}$. Then we have
$$(x_i)_k = \sum_{j:k\text{-th bit of } j \text{ is } 1} y_{ij}.$$
Then the polynomial $P'(\overline{y})$ for AND-OR($\overline{y}$) is the polynomial $P(\overline{x})$ with $(x_i)_k$ replaced as above, thus the degree won't increase. Correctness follows by the promise.

2. This is the most extraordinary step, due to Ambainis [Amb05]. In this notation, AND-OR becomes the indicator function of $\forall j, z_j \neq 0$. Define
$$Q(z_1, \ldots, z_R) := \mathop{\mathbb{E}}_{\substack{\overline{y}\text{: his rows have weight 1} \\ \text{and is consistent with } \overline{z}}} P(\overline{y}).$$
Clearly it is a good approximation of AND-OR($\overline{z}$). It remains to show that it's a polynomial of degree $k$ in $z$'s if $P$ is a polynomial of degree $k$ in $y$'s.

Let's look at one monomial of degree $k$ in $P$: $y_{i_1j_1}y_{i_2j_2}\cdots y_{i_kj_k}$. Observe that all $i_\ell$'s are distinct by the promise, and by $u^2 = u$ over $\{0,1\}$. By chain rule we have

$$\mathbb{E}[y_{i_1j_1}\cdots y_{i_kj_k}] = \mathbb{E}[y_{i_1j_1}]\mathbb{E}[y_{i_2j_2}|y_{i_1j_1} = 1]\cdots\mathbb{E}[y_{i_kj_k}|y_{i_1j_1} = \cdots = y_{i_{k-1}j_{k-1}} = 1].$$

By symmetry we have $\mathbb{E}[y_{i_1j_1}] = \frac{z_{j_1}}{N}$, which is linear in $z$'s. To get $\mathbb{E}[y_{i_2j_2}|y_{i_1j_1} = 1]$, we know that every other entry in row $i_1$ is 0, so we give away row $i_1$, average over $y$'s such that $\begin{cases} y_{i_1j_1} = 1 & \\ y_{ij} = 0 & j \neq j_1 \end{cases}$ under the promise and consistent with $z$'s. Therefore

$$\mathbb{E}[y_{i_2j_2}|y_{i_1j_1} = 1] = \begin{cases} \frac{z_{j_2}}{N-1} & j_1 \neq j_2, \\ \frac{z_{j_2}-1}{N-1} & j_1 = j_2. \end{cases}$$

In general we have

$$\mathbb{E}[y_{i_kj_k}|y_{i_1j_1} = \cdots = y_{i_{k-1}j_{k-1}} = 1] = \frac{z_{j_k} - \#\ell < k\colon j_\ell = j_k}{N-k+1},$$

which has degree 1 in $z$'s. Therefore the degree of $Q$ is not larger than that of $P$.

3. Note that $\forall j$, $z_j = \sum_i y_{ij}$. Hence by replacing $z$'s by $y$'s, the degree won't increase.

4. We can add a "slack" variable $z_0$, or equivalently $y_{01}, \ldots, y_{0N}$; then the condition $\sum_{j=0}^{R} z_j = N$ actually means $\sum_{j=1}^{R} z_j \leq N$.

□

*Proof idea for Theorem 13.* First, by the duality argument we can verify that $d_{1/3}^{\leq N}(f) \geq d$ if and only if there exists $d$-wise indistinguishable distributions $A, B$ such that:

- $f$ can distinguish $A, B$;
- $A$ and $B$ are supported on strings of weight $\leq N$.

**Claim 14.** $d_{1/3}^{\leq\sqrt{N}}(\mathrm{OR}_N) = \Omega(N^{1/4})$.

The proof needs a little more information about the weight distribution of the indistinguishable distributions corresponding to this claim. Basically, their expected weight is very small.

Now we combine these distributions with the usual ones for And using the lemma mentioned at the beginning.

What remains to show is that the final distribution is supported on Hamming weight $\leq N$. Because by construction the $R$ copies of the distributions for Or are sampled independently, we can use concentration of measure to prove a tail bound. This gives that all but an exponentially small measure of the distribution is supported on strings of weight $\leq N$. The final step of the proof consists of slightly tweaking the distributions to make that measure 0. □

# 5 Pseudorandom groups and communication complexity

Groups have many applications in theoretical computer science. Barrington [Bar89] used the permutation group $S_5$ to prove a very surprising result, which states that the majority function can be computed efficiently using only constant bits of memory (something which was conjectured to be false). More recently, catalytic computation [BCK$^+$14] shows that if we have a lot of memory, but it's full with junk that cannot be erased, we can still compute more than if we had little memory. We will see some interesting properties of groups in the following.

Some famous groups used in computer science are:

- $\{0,1\}^n$ with bit-wise addition;
- $\mathbb{Z}_m$ with addition mod $m$ ;
- $S_n$, which are permutations of $n$ elements;
- Wreath product $G := (\mathbb{Z}_m \times \mathbb{Z}_m) \wr \mathbb{Z}_2$ , whose elements are of the form $(a,b)z$ where $z$ is a "flip bit", with the following multiplication rules:
    - $(a,b)1 = 1(b,a)$ ;
    - $z \cdot z' := z + z'$ in $\mathbb{Z}_2$ ;

- $(a,b)\cdot(a',b') := (a+a', b+b')$ is the $\mathbb{Z}_m \times \mathbb{Z}_m$ operation;

An example is $(5,7)1 \cdot (2,1)1 = (5,7)1 \cdot 1(1,2) = (6,9)0$ . Generally we have

$$(a,b)z \cdot (a',b')z' = \begin{cases} (a+a', b+b')z+z' & z=1\,, \\ (a+b', b+a')z+z' & z=0\,; \end{cases}$$

- $SL_2(q) := \{2\times 2 \text{ matrices over } \mathbb{F}_q \text{ with determinant } 1\}$, in other words, group of matrices $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ such that $ad-bc=1$.

The group $SL_2(q)$ was invented by Galois. (If you haven't, read his biography on wikipedia.)

**Quiz**. Among these groups, which is the "least abelian"? The latter can be defined in several ways. We focus on this: If we have two high-entropy distributions $X, Y$ over $G$, does $X\cdot Y$ has more entropy? For example, if $X$ and $Y$ are uniform over some $\Omega(|G|)$ elements, is $X\cdot Y$ close to uniform over $G$? By "close to" we mean that the statistical distance is less that a small constant from the uniform distribution. For $G = (\{0,1\}^n, +)$, if $Y = X$ uniform over $\{0\}\times\{0,1\}^{n-1}$, then $X\cdot Y$ is the same, so there is not entropy increase even though $X$ and $Y$ are uniform on half the elements.

**Definition 1.**[Measure of Entropy] For $\|A\|_2 = (\sum_x A(x)^2)^{\frac{1}{2}}$, we think of $\|A\|_2^2 = 100\frac{1}{|G|}$ for "high entropy".

Note that $\|A\|_2^2$ is exactly the "collision probability", i.e. $\Pr[A = A']$. We will consider the entropy of the uniform distribution $U$ as very small, i.e. $\|U\|_2^2 = \frac{1}{|G|} \approx \|\bar{0}\|_2^2$. Then we have

$$\begin{aligned} \|A-U\|_2^2 &= \sum_x \left(A(x) - \frac{1}{|G|}\right)^2 \\ &= \sum_x A(x)^2 - 2A(x)\frac{1}{|G|} + \frac{1}{|G|^2} \\ &= \|A\|_2^2 - \frac{1}{|G|} \\ &= \|A\|_2^2 - \|U\|_2^2 \\ &\approx \|A\|_2^2\,. \end{aligned}$$

**Theorem 2.**[[Gow08], [BNP08]] If $X, Y$ are independent over $G$, then

$$\|X \cdot Y - U\|_2 \leq \|X\|_2 \|Y\|_2 \sqrt{\frac{|G|}{d}},$$

where $d$ is the minimum dimension of irreducible representation of $G$.

By this theorem, for high entropy distributions $X$ and $Y$, we get $\|X \cdot Y - U\|_2 \leq \frac{O(1)}{\sqrt{|G|d}}$, thus we have

$$\|X \cdot Y - U\|_1 \leq \sqrt{|G|} \|X \cdot Y - U\|_2 \leq \frac{O(1)}{\sqrt{d}}. \tag{1}$$

If $d$ is large, then $X \cdot Y$ is very close to uniform. The following table shows the $d$'s for the groups we've introduced.

| $G$ | $\{0,1\}^n$ | $\mathbb{Z}_m$ | $(\mathbb{Z}_m \times \mathbb{Z}_m) \wr \mathbb{Z}_2$ | $A_n$ | $SL_2(q)$ |
|---|---|---|---|---|---|
| $d$ | 1 | 1 | should be very small | $\frac{\log |G|}{\log \log |G|}$ | $|G|^{1/3}$ |

Here $A_n$ is the alternating group of even permutations. We can see that for the first groups, Equation (1) doesn't give non-trivial bounds.

But for $A_n$ we get a non-trivial bound, and for $SL_2(q)$ we get a strong bound: we have $\|X \cdot Y - U\|_2 \leq \frac{1}{|G|^{\Omega(1)}}$.

We now study the communication complexity of some problems on groups. We give the definition of a protocol when two parties are involved and generalize later to more parties.

**Definition 3.** A 2-party c-bit *deterministic* communication protocol is a depth-c binary tree such that:

- the leaves are the output of the protocol
- each internal node is labeled with a party and a function from that party's input space to $\{0, 1\}$

Computation is done by following a path on edges, corresponding to outputs of functions at the nodes.

A public-coin *randomized* protocol is a distribution on deterministic protocols.

## 5.1 2-party communication protocols

We start with a simple protocol for the following problem.

Let $G$ be a group. Alice gets $x \in G$ and Bob gets $y \in G$ and their goal is to check if $x \cdot y = 1_G$, or equivalently if $x = y^{-1}$.

There is a simple deterministic protocol in which Alice simply sends her input to Bob who checks if $x \cdot y = 1_G$. This requires $O(\log |G|)$ communication complexity.

We give a randomized protocol that does better in terms on communication complexity. Alice picks a random hash function $h : G \to \{0,1\}^\ell$. We can think that both Alice and Bob share some common randomness and thus they can agree on a common hash function to use in the protocol. Next, Alice sends $h(x)$ to Bob, who then checks if $h(x) = h(y^{-1})$.

For $\ell = O(1)$ we get constant error and constant communication.

## 5.2 3-party communication protocols

There are two ways to extend 2-party communication protocols to more parties. We first focus on the Number-in-hand (NIH), where Alice gets $x$, Bob gets $y$, Charlie gets $z$, and they want to check if $x \cdot y \cdot z = 1_G$. In the NIH setting the communication depends on the group $G$.

## 5.3 A randomized protocol for the hypercube

Let $G = (\{0,1\}^n, +)$ with addition modulo 2. We want to test if $x+y+z = 0^n$. First, we pick a linear hash function $h$, i.e. satisfying $h(x+y) = h(x) + h(y)$. For a uniformly random $a \in \{0,1\}^n$ set $h_a(x) = \sum a_i x_i \pmod 2$. Then,

- Alice sends $h_a(x)$
- Bob send $h_a(y)$
- Charlie accepts if and only if $\underbrace{h_a(x) + h_a(y)}_{h_a(x+y)} = h_a(z)$

The hash function outputs 1 bit. The error probability is 1/2 and the communication is $O(1)$. For a better error, we can repeat.

## 5.4 A randomized protocol for $\mathbb{Z}_m$

Let $G = (\mathbb{Z}_m, +)$ where $m = 2^n$. Again, we want to test if $x + y + z = 0 \pmod m$. For this group, there is no 100% linear hash function but there are almost linear hash function families $h : \mathbb{Z}_m \to \mathbb{Z}_\ell$ that satisfy the following properties:

1. $\forall a, x, y$ we have $h_a(x) + h_a(y) = h_a(x + y) \pm 1$
2. $\forall x \neq 0$ we have $\Pr_a[h_a(x) \in \{\pm 2, \pm 1, 0\}] \leq 2^{-\Omega(\ell)}$
3. $h_a(0) = 0$

Assuming some random hash function $h$ (from a family) that satisfies the above properties the protocol works similar to the previous one.

- Alice sends $h_a(x)$
- Bob sends $h_a(y)$
- Charlie accepts if and only if $h_a(x) + h_a(y) + h_a(z) \in \{\pm 2, \pm 1, 0\}$

We can set $\ell = O(1)$ to achieve constant communication and constant error.

Analysis

To prove correctness of the protocol, first note that $h_a(x)+h_a(y)+h_a(z) = h_a(x + y + z) \pm 2$, then consider the following two cases:

- if $x + y + z = 0$ then $h_a(x + y + z) \pm 2 = h_a(0) \pm 2 = 0 \pm 2$
- if $x + y + z \neq 0$ then $\Pr_a[h_a(x + y + z) \in \{\pm 2, \pm 1, 0\}] \leq 2^{-\Omega(\ell)}$

It now remains to show that such hash function families exist.

Let $a$ be a random odd number modulo $2^n$. Define

$$h_a(x) := (a \cdot x \gg n - \ell) \pmod{2^\ell}$$

where the product $a \cdot x$ is integer multiplication. In other words we output the bits $n - \ell + 1, n - \ell + 2, \ldots, n$ of the integer product $a \cdot x$.

We now verify that the above hash function family satisfies the three properties we required above.

Property (3) is trivially satisfied.

For property (1) we have the following. Let $s = a \cdot x$ and $t = a \cdot y$ and $u = n - \ell$. The bottom line is how $(s \gg u) + (t \gg u)$ compares with $(s + t) \gg u$. In more detail we have that,

- $h_a(x+y) = ((s+t) \gg u) \pmod{2^\ell}$
- $h_a(x) = (s \gg u) \pmod{2^\ell}$
- $h_a(x) = (t \gg u) \pmod{2^\ell}$

Notice, that if in the addition $s+t$ the carry into the $u+1$ bit is 0, then

$$(s \gg u) + (t \gg u) = (s+t) \gg u$$

otherwise

$$(s \gg u) + (t \gg u) + 1 = (s+t) \gg u$$

which concludes the proof for property (1).

Finally, we prove property (2). We start by writing $x = s \cdot 2^c$ where $s$ is odd. Bitwise, this looks like $(\cdots\cdots 1\underbrace{0\cdots 0}_{c\text{ bits}})$.

The product $a{\cdot}x$ for a uniformly random $a$, bitwise looks like $(\mathit{uniform}\, 1\underbrace{0\cdots 0}_{c\text{ bits}})$. We consider the two following cases for the product $a \cdot x$:

1. If $a{\cdot}x = (\underbrace{\mathit{uniform}\ 1\ \overbrace{00}^{2\ \mathit{bits}}}_{\ell\ \mathit{bits}}\cdots 0)$, or equivalently $c \geq n-\ell+2$, the output never lands in the bad set $\{\pm 2, \pm 1, 0\}$ (some thought should be given to the representation of negative numbers – we ignore that for simplicity).

2. Otherwise, the hash function output has $\ell - O(1)$ uniform bits. Again for simplicity, let $B = \{0,1,2\}$. Thus,

   $$\Pr[\text{output} \in B] \leq |B| \cdot 2^{-\ell+O(1)}$$

   In other words, the probability of landing in any small set is small.

## 5.5 Other groups

What happens in other groups? Do we have an almost linear hash function for $2 \times 2$ matrices? The answer is negative. For $SL_2(q)$ and $A_n$ the problem of testing equality with $1_G$ is hard.

We would like to rule out randomized protocols, but it is hard to reason about them directly. Instead, we are going to rule out deterministic protocols on random inputs. For concreteness our main focus will be $SL_2(q)$.

First, for any group element $g \in G$ we define the distribution on triples, $D_g := (x, y, (x \cdot y)^{-1} g)$, where $x, y \in G$ are uniformly random elements. Note the product of the elements in $D_g$ is always $g$.

Towards a contradiction, suppose we have a randomized protocol $P$ for the $xyz =^? 1_G$ problem. In particular, we have

$$\Pr[P(D_1) = 1] \geq \Pr[P(D_h) = 1] + \frac{1}{10}.$$

This implies a deterministic protocol with the same gap, by fixing the randomness.

We reach a contradiction by showing that for every *deterministic* protocols $P$ using little communication (will quantify later), we have

$$|\Pr[P(D_1) = 1] - \Pr[P(D_h) = 1]| \leq \frac{1}{100}.$$

We start with the following lemma, which describes a protocol using product sets.

**Lemma 4.** (The set of accepted inputs of) A deterministic $c$-bit protocol can be written as a disjoint union of $2^c$ "rectangles," that is sets of the form $A \times B \times C$.

*Proof.* (sketch) For every communication transcript $t$, let $S_t \subseteq G^3$ be the set of inputs giving transcript $t$. The sets $S_t$ are disjoint since an input gives only one transcript, and their number is $2^c$, i.e. one for each communication transcript of the protocol. The rectangle property can be proven by induction on the protocol tree. □

Next, we show that these product sets cannot distinguish these two distributions $D_1, D_h$, and for that we will use the pseudorandom properties of the group $G$.

**Lemma 5.** For all $A, B, C \subseteq G$ and we have

$$|\Pr[A \times B \times C(D_1) = 1] - \Pr[A \times B \times C(D_h) = 1]| \leq \frac{1}{d^{\Omega(1)}}.$$

Recall the parameter $d$ from the previous lectures and that when the group $G$ is $SL_2(q)$ then $d = |G|^{\Omega(1)}$.

*Proof.* Pick any $h \in G$ and let $x, y, z$ be the inputs of Alice, Bob, and Charlie respectively. Then

$$\Pr[A \times B \times C(D_h) = 1] = \Pr[(x,y) \in A \times B] \cdot \Pr[(x \cdot y)^{-1} \cdot h \in C | (x,y) \in A \times B]$$

If either $A$ or $B$ is small, that is $\Pr[x \in A] \leq \epsilon$ or $\Pr[y \in B] \leq \epsilon$, then also $\Pr[P(D_h) = 1] \leq \epsilon$ because the term $\Pr[(x,y) \in A \times B]$ will be small. We will choose $\epsilon$ later.

Otherwise, $A$ and $B$ are large, which implies that $x$ and $y$ are uniform over at least $\epsilon|G|$ elements. Recall from Lecture 9 that this implies $\|x \cdot y - U\|_2 \leq \|x\|_2 \cdot \|y\|_2 \cdot \sqrt{\frac{|G|}{d}}$, where $U$ is the uniform distribution.

By Cauchy–Schwarz we obtain,

$$\|x \cdot y - U\|_1 \leq |G| \cdot \|x\|_2 \cdot \|y\|_2 \cdot \sqrt{\frac{1}{d}} \leq \frac{1}{\epsilon} \cdot \frac{1}{\sqrt{d}}.$$

The last inequality follows from the fact that $\|x\|_2, \|y\|_2 \leq \sqrt{\frac{1}{\epsilon|G|}}$.

This implies that $\|(x \cdot y)^{-1} - U\|_1 \leq \frac{1}{\epsilon} \cdot \frac{1}{\sqrt{d}}$ and $\|(x \cdot y)^{-1} \cdot h - U\|_1 \leq \frac{1}{\epsilon} \cdot \frac{1}{\sqrt{d}}$, because taking inverses and multiplying by $h$ does not change anything. These two last inequalities imply that,

$$\Pr[(x \cdot y)^{-1} \in C | (x,y) \in A \times B] = \Pr[(x \cdot y)^{-1} \cdot h \in C | (x,y) \in A \times B] \pm \frac{2}{\epsilon} \frac{1}{\sqrt{d}}$$

and thus we get that,

$$\Pr[P(D_1) = 1] = \Pr[P(D_h) = 1] \pm \frac{2}{\epsilon} \frac{1}{\sqrt{d}}.$$

To conclude, based on all the above we have that for all $\epsilon$ and independent of the choice of $h$, it is either the case that

$$|\Pr[P(D_1) = 1] - \Pr[P(D_h) = 1]| \leq 2\epsilon$$

or

$$|\Pr[P(D_1) = 1] - \Pr[P(D_h) = 1]| \leq \frac{2}{\epsilon} \frac{1}{\sqrt{d}}$$

and we will now choose the $\epsilon$ to balance these two cases and finish the proof:

$$\frac{2}{\epsilon} \frac{1}{\sqrt{d}} = 2\epsilon \Leftrightarrow \frac{1}{\sqrt{d}} = \epsilon^2 \Leftrightarrow \epsilon = \frac{1}{d^{1/4}}.$$

□

The above proves that the distribution $D_h$ behaves like the uniform distribution for product sets, for all $h \in G$.

Returning to arbitrary deterministic protocols $P$, write $P$ as a union of $2^c$ disjoint rectangles by the first lemma. Applying the second lemma and summing over all rectangles we get that the distinguishing advantage of $P$ is at most $2^c/d^{1/4}$. For $c \leq (1/100)\log d$ the advantage is at most $1/100$ and thus we get a contradiction on the existence of such a correct protocol. We have concluded the proof of this theorem.

**Theorem 6.** Let $G$ be a group, and $d$ be the minimum dimension of an irreducible representation of $G$. Consider the 3-party, number-in-hand communication protocol $f : G^3 \to \{0,1\}$ where $f(x,y,z) = 1 \Leftrightarrow x \cdot y \cdot z = 1_G$. Its randomized communication complexity is $\Omega(\log d)$.

For $SL_2(q)$ the communication is $\Omega(\log |G|)$. This is tight up to constants, because Alice can send her entire group element.

For the group $A_n$ the known bounds on $d$ yield communication $\Omega(\log\log |G|)$. This bound is tight for the problem of distinguishing $D_1$ from $D_h$ for $h \neq 1$, as we show next. The identity element $1_G$ for the group $A_n$ is the identity permutation. If $h \neq 1_G$ then $h$ is a permutation that maps some element $a \in G$ to $h(a) = b \neq a$. The idea is that the parties just need to "follow" $a$, which is logarithmically smaller than $G$. Specifically, let $x, y, z$ be the permutations that Alice, Bob and Charlie get. Alice sends $x(a) \in [n]$. Bob gets $x(a)$ and sends $y(x(a)) \in [n]$ to Charlie who checks if $z(y(x(a))) = 1$. The communication is $O(\log n)$. Because the size of the group is $|G| = \Theta(n!) = \Theta\left(\left(\frac{n}{e}\right)^n\right)$, the communication is $O(\log\log |G|)$.

This is also a proof that $d$ cannot be too large for $A_n$, i.e. is at most $(\log |G|)^{O(1)}$.

## 5.6 More on 2-party protocols

We move to another setting where a clean answer can be given. Here we only have two parties. Alice gets $x_1, x_2, \ldots, x_n$, Bob gets $y_1, y_2, \ldots, y_n$, and they want to know if $x_1 \cdot y_1 \cdot x_2 \cdot y_2 \cdots x_n \cdot y_n = 1_G$.

When $G$ is abelian, the elements can be reordered as to check whether $(x_1 \cdot x_2 \cdots x_n) \cdot (y_1 \cdot y_2 \cdots y_n) = 1_G$. This requires constant communication (using randomness) as we saw in Lecture 12, since it is equivalent to the check $x \cdot y = 1_G$ where $x = x_1 \cdot x_2 \cdots x_n$ and $y = y_1 \cdot y_2 \cdots y_n$.

We will prove the next theorem for non-abelian groups.

**Theorem 7.** For every non-abelian group $G$ the communication of deciding if $x_1 \cdot y_1 \cdot x_2 \cdot y_2 \cdots x_n \cdot y_n = 1_G$ is $\Omega(n)$.

*Proof.* We reduce from unique disjointness, defined below. For the reduction we will need to encode the And of two bits $x, y \in \{0,1\}$ as a group product. (This question is similar to a puzzle that asks how to hang a picture on the wall with two nails, such that if either one of the nails is removed, the picture will fall. This is like computing the And function on two bits, where both bits (nails) have to be 1 in order for the function to be 1.) Since $G$ is non-abelian, there exist $a, b \in G$ such that $a \cdot b \neq b \cdot a$, and in particular $a \cdot b \cdot a^{-1} \cdot b^{-1} = h$ with $h \neq 1$. We can use this fact to encode And as

$$a^x \cdot b^y \cdot a^{-x} \cdot b^{-y} = \begin{cases} 1, & \text{if And(x,y)=0} \\ h, & \text{otherwise} \end{cases} .$$

In the disjointness problem Alice and Bob get inputs $x, y \in \{0,1\}^n$ respectively, and they wish to check if there exists an $i \in [n]$ such that $x_i \wedge y_i = 1$. If you think of them as characteristic vectors of sets, this problem is asking if the sets have a common element or not. The communication of this problem is $\Omega(n)$. Moreover, in the variant of this problem where the number of such $i$'s is 0 or 1 (i.e. unique), the same lower bound $\Omega(n)$ still applies. This is like giving Alice and Bob two sets that either are disjoint or intersect in exactly one element, and they need to distinguish these two cases.

Next, we will reduce the above variant of the set disjointness to group products. For $x, y \in \{0,1\}^n$ we product inputs for the group problem as follows:

$$\begin{aligned} x &\longrightarrow (a^{x_1}, a^{-x_1}, \ldots, a^{x_n}, a^{-x_n}) \\ y &\longrightarrow (b^{y_1}, b^{-y_1}, \ldots, b^{y_n}, b^{-y_n}). \end{aligned}$$

Now, the product $x_1 \cdot y_1 \cdot x_2 \cdot y_2 \cdots x_n \cdot y_n$ we originally wanted to compute becomes

$$\underbrace{a^{x_1} \cdot b^{y_1} \cdot a^{-x_1} \cdot b^{-y_1}}_{\text{1 bit}} \cdot \cdots \cdot a^{x_n} \cdot b^{y_n} \cdot a^{-x_n} \cdot b^{-y_n}.$$

If there isn't an $i \in [n]$ such that $x_i \wedge y_i = 1$, then each product term $a^{x_i} \cdot b^{y_i} \cdot a^{-x_i} \cdot b^{-y_i}$ is 1 for all $i$, and thus the whole product is 1.

Otherwise, there exists a unique $i$ such that $x_i \wedge y_i = 1$ and thus the product will be $1 \cdots 1 \cdot h \cdot 1 \cdots 1 = h$, with $h$ being in the $i$-th position. If

Alice and Bob can test if the above product is equal to 1, they can also solve the unique set disjointness problem, and thus the lower bound applies for the former. □

We required the uniqueness property, because otherwise we might get a product $h^c$ that could be equal to 1 in some groups.

# 6 Number-on-forehead communication complexity

In number-on-forehead (NOH) communication complexity each party $i$ sees all of the input $(x_1, \ldots, x_k)$ except its own input $x_i$. For background, it is not known how to prove negative results for $k \geq \log n$ parties. We shall focus on the problem of separating deterministic and randomizes communication. For $k = 2$, we know the optimal separation: The equality function requires $\Omega(n)$ communication for deterministic protocols, but can be solved using $O(1)$ communication if we allow the protocols to use public coins. For $k = 3$, the best known separation between deterministic and randomized protocol is $\Omega(\log n)$ vs $O(1)$ [BDPW10]. In the following we give a new proof of this result, for a simpler function: $f(x, y, z) = 1$ if and only if $x \cdot y \cdot z = 1$ for $x, y, z \in SL_2(q)$.

For context, let us state and prove the upper bound for randomized communication.

**Claim 1.** $f$ has randomized communication complexity $O(1)$.

*Proof.* In the NOH model, computing $f$ reduces to 2-party equality with no additional communication: Alice computes $y \cdot z =: w$ privately, then Alice and Bob check if $x = w^{-1}$. □

To prove a $\Omega(\log n)$ lower bound for deterministic protocols, where $n = \log |G|$, we reduce the communication problem to a combinatorial problem.

**Definition 2.** A *corner* in a group $G$ is $\{(x, y), (xz, y), (x, zy)\} \subseteq G^2$, where $x, y$ are arbitrary group elements and $z \neq 1_G$.

For intuition, consider the case when $G$ is Abelian, where one can replace multiplication by addition and a corner becomes $\{(x, y), (x + z, y), (x, y + z)\}$ for $z \neq 0$.

We now state the theorem that gives the lower bound.

**Theorem 3.** Suppose that every subset $A \subseteq G^2$ with $\mu(A) := |A|/|G^2| \geq \delta$ contains a corner. Then the deterministic communication complexity of $f(x,y,z) = 1 \iff x \cdot y \cdot z = 1_G$ is $\Omega(\log(1/\delta))$.

It is known that when $G$ is Abelian, then $\delta \geq 1/\text{polyloglog}|G|$ implies a corner. We shall prove that when $G = SL_2(q)$, then $\delta \geq 1/\text{polylog}|G|$ implies a corner. This in turn implies communication $\Omega(\log\log|G|) = \Omega(\log n)$.

*Proof.* We saw that a number-in-hand (NIH) $c$-bit protocol can be written as a disjoint union of $2^c$ rectangles. Likewise, a number-on-forehead $c$-bit protocol $P$ can be written as a disjoint union of $2^c$ *cylinder intersections* $C_i := \{(x,y,z) : f_i(y,z)g_i(x,z)h_i(x,y) = 1\}$ for some $f_i, g_i, h_i \colon G^2 \to \{0,1\}$:

$$P(x,y,z) = \sum_{i=1}^{2^c} f_i(y,z)g_i(x,z)h_i(x,y).$$

The proof idea of the above fact is to consider the $2^c$ transcripts of $P$, then one can see that the inputs giving a fixed transcript are a cylinder intersection.

Let $P$ be a $c$-bit protocol. Consider the inputs $\{(x,y,(xy)^{-1})\}$ on which $P$ accepts. Note that at least $2^{-c}$ fraction of them are accepted by some cylinder intersection $C$. Let $A := \{(x,y) : (x,y,(xy)^{-1}) \in C\} \subseteq G^2$. Since the first two elements in the tuple determine the last, we have $\mu(A) \geq 2^{-c}$.

Now suppose $A$ contains a corner $\{(x,y),(xz,y),(x,zy)\}$. Then

$$\begin{aligned}
(x,y) \in A &\implies (x,y,(xy)^{-1}) \in C && \implies h(x,y) = 1,\\
(xz,y) \in A &\implies (xz,y,(xzy)^{-1}) \in C && \implies f(y,(xyz)^{-1}) = 1,\\
(x,zy) \in A &\implies (x,zy,(xzy)^{-1}) \in C && \implies g(x,(xyz)^{-1}) = 1.
\end{aligned}$$

This implies $(x,y,(xzy)^{-1}) \in C$, which is a contradiction because $z \neq 1$ and so $x \cdot y \cdot (xzy)^{-1} \neq 1_G$. □

# 7 Corners in pseudorandom groups

In this section we prove the corners theorem for pseudorandom groups, following Austin [Aus16]. Our exposition has several non-major differences with that in [Aus16], which may make it more computer-science friendly. The instructor suspects a proof can also be obtained via certain local modifications and simplifications of Green's exposition [Gre05b, Gre05a] of an earlier proof for the abelian case. We focus on the case $G = SL_2(q)$ for simplicity, but the proof immediately extends to other pseudorandom groups.

**Theorem 1.** *Let $G = SL_2(q)$. Every subset $A \subseteq G^2$ of density $\mu(A) \geq 1/\log^a |G|$ contains a corner, i.e., a set of the form $\{(x, y), (xz, y), (x, zy) \mid z \neq 1\}$.*

## 7.1 Proof Overview

For intuition, suppose $A$ is a product set, i.e., $A = B \times C$ for $B, C \subseteq G$. Let's look at the quantity

$$\mathbb{E}_{x,y,z \leftarrow G}[A(x, y)A(xz, y)A(x, zy)]$$

where $A(x, y) = 1$ iff $(x, y) \in A$. Note that the random variable in the expectation is equal to 1 exactly when $x, y, z$ form a corner in $A$. We'll show that this quantity is greater than $1/|G|$, which implies that $A$ contains a corner (where $z \neq 1$). Since we are taking $A = B \times C$, we can rewrite the above quantity as

$$\begin{aligned}
&\mathbb{E}_{x,y,z \leftarrow G}[B(x)C(y)B(xz)C(y)B(x)C(zy)] \\
&= \mathbb{E}_{x,y,z \leftarrow G}[B(x)C(y)B(xz)C(zy)] \\
&= \mathbb{E}_{x,y,z \leftarrow G}[B(x)C(y)B(z)C(x^{-1}zy)]
\end{aligned}$$

where the last line follows by replacing $z$ with $x^{-1}z$ in the uniform distribution. If $\mu(A) \geq \delta$, then $\mu(B) \geq \delta$ and $\mu(C) \geq \delta$. Condition on $x \in B$, $y \in C$, $z \in B$. Then the distribution $x^{-1}zy$ is a product of three independent distributions, each uniform on a set of measure greater than $\delta$. By pseudorandomness $x^{-1}zy$ is $1/|G|^{\Omega(1)}$ close to uniform in statistical distance. This implies that the above quantity equals

$$\begin{aligned}
&\mu(A) \cdot \mu(C) \cdot \mu(B) \cdot \left(\mu(C) \pm \frac{1}{|G|^{\Omega(1)}}\right) \\
&\geq \delta^3 \left(\delta - \frac{1}{|G|^{\Omega(1)}}\right) \\
&\geq \delta^4/2 \\
&> 1/|G|.
\end{aligned}$$

Given this, it is natural to try to write an arbitrary $A$ as a combination of product sets (with some error). We will make use of a more general result.

## 7.2 Weak Regularity Lemma

Let $U$ be some universe (we will take $U = G^2$). Let $f : U \to [-1, 1]$ be a function (for us, $f = 1_A$). Let $D \subseteq \{d : U \to [-1, 1]\}$ be some set of functions, which can be thought of as "easy functions" or "distinguishers."

**Theorem 2.**[Weak Regularity Lemma] For all $\epsilon > 0$, there exists a function $g := \sum_{i \leq s} c_i \cdot d_i$ where $d_i \in D$, $c_i \in \mathbb{R}$ and $s = 1/\epsilon^2$ such that for all $d \in D$

$$\mathbb{E}_{x \leftarrow U}[f(x) \cdot d(x)] = \mathbb{E}_{x \leftarrow U}[g(x) \cdot d(x)] \pm \epsilon.$$

The lemma is called 'weak' because it came after Szemerédi's regularity lemma, which has a stronger distinguishing conclusion. However, the lemma is also 'strong' in the sense that Szemerédi's regularity lemma has $s$ as a tower of $1/\epsilon$ whereas here we have $s$ polynomial in $1/\epsilon$. The weak regularity lemma is also simpler. There also exists a proof of Szemerédi's theorem (on arithmetic progressions), which uses weak regularity as opposed to the full regularity lemma used initially.

*Proof.* We will construct the approximation $g$ through an iterative process producing functions $g_0, g_1, \ldots, g$. We will show that $||f - g_i||_2^2$ decreases by $\geq \epsilon^2$ each iteration.

1. **Start**: Define $g_0 = 0$ (which can be realized setting $c_0 = 0$).
2. **Iterate**: If not done, there exists $d \in D$ such that $|\mathbb{E}[(f - g) \cdot d]| > \epsilon$. Assume without loss of generality $\mathbb{E}[(f - g) \cdot d] > \epsilon$.
3. **Update**: $g' := g + \lambda d$ where $\lambda \in \mathbb{R}$ shall be picked later.

Let us analyze the progress made by the algorithm.

$$\begin{aligned}
||f - g'||_2^2 &= \mathbb{E}_x[(f - g')^2(x)] \\
&= \mathbb{E}_x[(f - g - \lambda d)^2(x)] \\
&= \mathbb{E}_x[(f - g)^2] + \mathbb{E}_x[\lambda^2 d^2(x)] - 2\mathbb{E}_x[(f - g) \cdot \lambda d(x)] \\
&\leq ||f - g||_2^2 + \lambda^2 - 2\lambda \mathbb{E}_x[(f - g) d(x)] \\
&\leq ||f - g||_2^2 + \lambda^2 - 2\lambda\epsilon \\
&\leq ||f - g||_2^2 - \epsilon^2
\end{aligned}$$

where the last line follows by taking $\lambda = \epsilon$. Therefore, there can only be $1/\epsilon^2$ iterations because $||f - g_0||_2^2 = ||f||_2^2 \leq 1$. □

## 7.3 Getting more for rectangles

Returning to the lower bound proof, we will use the weak regularity lemma to approximate the indicator function for arbitrary $A$ by rectangles. That is, we take $D$ to be the collection of indicator functions for all sets of the form $S \times T$ for $S, T \subseteq G$. The weak regularity lemma gives us $A$ as a linear combination of rectangles. These rectangles may overlap. However, we ideally want $A$ to be a linear combination of *non-overlapping* rectangles.

**Claim 3.** Given a decomposition of $A$ into rectangles from the weak regularity lemma with $s$ functions, there exists a decomposition with $2^{O(s)}$ rectangles which don't overlap.

*Proof.* Exercise. □

In the above decomposition, note that it is natural to take the coefficients of rectangles to be the density of points in $A$ that are in the rectangle. This gives rise to the following claim.

**Claim 4.** The weights of the rectangles in the above claim can be the average of $f$ in the rectangle, at the cost of doubling the distinguisher error.

Consequently, we have that $f = g + h$, where $g$ is the sum of $2^{O(s)}$ non-overlapping rectangles $S \times T$ with coefficients $\Pr_{(x,y)\in S\times T}[f(x, y) = 1]$.

*Proof.* Let $g$ be a partition decomposition with arbitrary weights. Let $g'$ be a partition decomposition with weights being the average of $f$. It is enough to show that for all rectangle distinguishers $d \in D$

$$|\mathbb{E}[(f - g')d]| \leq |\mathbb{E}[(f - g)d]|.$$

By the triangle inequality, we have that

$$|\mathbb{E}[(f - g')d]| \leq |\mathbb{E}[(f - g)d]| + |\mathbb{E}[(g - g')d]|.$$

To bound $\mathbb{E}[(g - g')d]|$, note that the error is maximized for a $d$ that respects the decomposition in non-overlapping rectangles, i.e., $d$ is the union of some non-overlapping rectangles from the decomposition. This can be argues using that, unlike $f$, the value of $g$ and $g'$ on a rectangle $S \times T$ from the decomposition is fixed. But, for such $d$, $g' = f$! More formally, $\mathbb{E}[(g - g')d] = \mathbb{E}[(g - f)d]$. □

We need to get a little more from this decomposition. The conclusion of the regularity lemma holds with respect to distinguishers that can be written as $U(x) \cdot V(y)$ where $U$ and $V$ map $G \to \{0,1\}$. We need the same guarantee for $U$ and $V$ with range $[-1,1]$. This can be accomplished paying only a constant factor in the error, as follows. Let $U$ and $V$ have range $[-1,1]$. Write $U = U_+ - U_-$ where $U_+$ and $U_-$ have range $[0,1]$, and the same for $V$. The error for distinguisher $U \cdot V$ is at most the sum of the errors for distinguishers $U_+ \cdot V_+$, $U_+ \cdot V_-$, $U_- \cdot V_+$, and $U_- \cdot V_-$. So we can restrict our attention to distinguishers $U(x) \cdot V(y)$ where $U$ and $V$ have range $[0,1]$. In turn, a function $U(x)$ with range $[0,1]$ can be written as an expectation $\mathbb{E}_a U_a(x)$ for functions $U_a$ with range $\{0,1\}$, and the same for $V$. We conclude by observing that

$$\mathbb{E}_{x,y}[(f-g)(x,y)\mathbb{E}_a U_a(x) \cdot \mathbb{E}_b V_b(y)] \leq \max_{a,b} \mathbb{E}_{x,y}[(f-g)(x,y) U_a(x) \cdot V_b(y)].$$

## 7.4 Proof

Let us now finish the proof by showing a corner exists for sufficiently dense sets $A \subseteq G^2$. We'll use three types of decompositions for $f : G^2 \to \{0,1\}$, with respect to the following three types of distinguishers, where $U_i$ and $V_i$ have range $\{0,1\}$:

1. $U_1(x) \cdot V_1(y)$,
2. $U_2(xy) \cdot V_2(y)$,
3. $U_3(x) \cdot V_3(xy)$.

The last two distinguishers can be visualized as parallelograms with a 45-degree angle between two segments. The same extra properties we discussed for rectangles hold for them too.

Recall that we want to show

$$\mathbb{E}_{x,y,g}[f(x,y)f(xg,y)f(x,gy)] > \frac{1}{|G|}.$$

We'll decompose the $i$-th occurrence of $f$ via the $i$-th decomposition listed above. We'll write this decomposition as $f = g_i + h_i$. We do this in the following order:

$$\begin{aligned} & f(x,y)\cdot f(xg,y)\cdot f(x,gy) \\ = \; & f(x,y)f(xg,y)g_3(x,gy) + f(x,y)f(xg,y)h_3(x,gy) \\ & \vdots \\ = \; & g_1g_2g_3 + h_1g_2g_3 + fh_2g_3 + ffh_3 \end{aligned}$$

We first show that $\mathbb{E}[g_1g_2g_3]$ is big (i.e., inverse polylogarithmic in expectation) in the next two claims. Then we show that the expectations of the other terms are small.

**Claim 5.** For all $g \in G$, the values $\mathbb{E}_{x,y}[g_1(x,y)g_2(xg,y)g_3(x,gy)]$ are the same (over $g$) up to an error of $2^{O(s)} \cdot 1/|G|^{\Omega(1)}$.

*Proof.* We just need to get error $1/|G|^{\Omega(1)}$ for any product of three functions for the three decomposition types. By the standard pseudorandomness argument we saw in previous lectures,

$$\begin{aligned} & \mathbb{E}_{x,y}[c_1U_1(x)V_1(y)\cdot c_2U_2(xgy)V_2(y)\cdot c_3U_3(x)V_3(xgy)] \\ & \quad = c_1c_2c_3\mathbb{E}_{x,y}[(U_1\cdot U_3)(x)(V_1\cdot V_2)(y)(U_2\cdot V_3)(xgy)] \\ & \quad = c_1c_2c_3\cdot \mu(U_1\cdot U_3)\mu(V_1\cdot V_2)\mu(U_2\cdot V_3) \pm \frac{1}{|G|^{\Omega(1)}}. \end{aligned}$$

□

Recall that we start with a set of density $\geq 1/\log^a |G|$.

**Claim 6.** $\mathbb{E}_{g,x,y}[g_1g_2g_3] > \Omega(1/\log^{4a}|G|)$.

*Proof.* By the previous claim, we can fix $g = 1_G$. We will relate the expectation over $x, y$ to $f$ by a trick using the Hölder inequality: For random variables $X_1, X_2, \ldots, X_k$,

$$\mathbb{E}[X_1 \ldots X_k] \leq \prod_{i=1}^{k} \mathbb{E}[X_i^{c_i}]^{1/c_i} \text{ such that } \sum 1/c_i = 1.$$

To apply this inequality in our setting, write

$$\mathbb{E}[f] = \mathbb{E}\left[(f\cdot g_1g_2g_3)^{1/4}\cdot\left(\frac{f}{g_1}\right)^{1/4}\cdot\left(\frac{f}{g_2}\right)^{1/4}\cdot\left(\frac{f}{g_3}\right)^{1/4}\right].$$

By the Hölder inequality, we get that

$$\mathbb{E}[f] \leq \mathbb{E}[f \cdot g_1 g_2 g_3]^{1/4} \mathbb{E}\left[\frac{f}{g_1}\right]^{1/4} \mathbb{E}\left[\frac{f}{g_2}\right]^{1/4} \mathbb{E}\left[\frac{f}{g_3}\right]^{1/4}.$$

Note that

$$\begin{aligned}
\mathbb{E}_{x,y}\frac{f(x,y)}{g_1(x,y)} &= \mathbb{E}_{x,y}\frac{f(x,y)}{\mathbb{E}_{x',y' \in \mathit{Cell}(x,y)}[f(x',y')]} \\
&= \mathbb{E}_{x,y}\frac{\mathbb{E}_{x',y' \in \mathit{Cell}(x,y)}[f(x',y')]}{\mathbb{E}_{x',y' \in \mathit{Cell}(x,y)}[f(x',y')]} \\
&= 1
\end{aligned}$$

where $\mathit{Cell}(x,y)$ is the set in the partition that contains $(x,y)$. Finally, by non-negativity of $f$, we have that $\mathbb{E}[f \cdot g_1 g_2 g_3]^{1/4} \leq \mathbb{E}[g_1 g_2 g_3]$. This concludes the proof. □

We've shown that the $g_1 g_2 g_3$ term is big. It remains to show the other terms are small. Let $\epsilon$ be the error in the weak regularity lemma with respect to distinguishers with range $[-1,1]$.

**Claim 7.** $|\mathbb{E}[ffh_3]| \leq \epsilon^{1/4}$.

*Proof.* Replace $g$ with $gy^{-1}$ in the uniform distribution to get

$$\begin{aligned}
&\mathbb{E}^4_{x,y,g}[f(x,y)f(xg,y)h_3(x,gy)] \\
&= \mathbb{E}^4_{x,y,g}[f(x,y)f(xgy^{-1},y)h_3(x,g)] \\
&= \mathbb{E}^4_{x,y}[f(x,y)\mathbb{E}_g[f(xgy^{-1},y)h_3(x,g)]] \\
&\leq \mathbb{E}^2_{x,y}[f^2(x,y)]\mathbb{E}^2_{x,y}\mathbb{E}^2_g[f(xgy^{-1},y)h_3(x,g)] \\
&\leq \mathbb{E}^2_{x,y}\mathbb{E}^2_g[f(xgy^{-1},y)h_3(x,g)] \\
&= \mathbb{E}^2_{x,y,g,g'}[f(xgy^{-1},y)h_3(x,g)f(xg'y^{-1},y)h_3(x,g')],
\end{aligned}$$

where the first inequality is by Cauchy-Schwarz.

Now replace $g \to x^{-1}g, g' \to x^{-1}g$ and reason in the same way:

$$\begin{aligned}
&= \mathbb{E}^2_{x,y,g,g'}[f(gy^{-1},y)h_3(x,x^{-1}g)f(g'y^{-1},y)h_3(x,x^{-1}g')] \\
&= \mathbb{E}^2_{g,g',y}[f(gy^{-1},y) \cdot f(g'y^{-1},y)\mathbb{E}_x[h_3(x,x^{-1}g) \cdot h_3(x,x^{-1}g')]] \\
&\leq \mathbb{E}_{x,x',g,g'}[h_3(x,x^{-1}g)h_3(x,x^{-1}g')h_3(x',x'^{-1}g)h_3(x',x'^{-1}g')].
\end{aligned}$$

Replace $g \to xg$ to rewrite the expectation as

$$\mathbb{E}[h_3(x,g)h_3(x,x^{-1}g')h_3(x',x'^{-1}xg)h_3(x',x'^{-1}g')].$$

We want to view the last three terms as a distinguisher $U(x) \cdot V(xg)$. First, note that $h_3$ has range $[-1,1]$. This is because $h_3(x,y) = f(x,y) - \mathbb{E}_{x',y' \in Cell(x,y)} f(x',y')$ and $f$ has range $\{0,1\}$.

Fix $x', g'$. The last term in the expectation becomes a constant $c \in [-1,1]$. The second term only depends on $x$, and the third only on $xg$. Hence for appropriate functions $U$ and $V$ with range $[-1,1]$ this expectation can be rewritten as

$$\mathbb{E}[h_3(x,g)U(x)V(xg)],$$

which concludes the proof. □

There are similar proofs to show the remaining terms are small. For $fh_2g_3$, we can perform simple manipulations and then reduce to the above case. For $h_1g_2g_3$, we have a slightly easier proof than above.

### 7.4.1 Parameters

Suppose our set has density $\delta \geq 1/\log^a |G|$. We apply the weak regularity lemma for error $\epsilon = 1/\log^c |G|$. This yields the number of functions $s = 2^{O(1/\epsilon^2)} = 2^{O(\log^{2c}|G|)}$. For say $c = 1/3$, we can bound $\mathbb{E}_{x,y,g}[g_1g_2g_3]$ from below by the same expectation with $g$ fixed to 1, up to an error $1/|G|^{\Omega(1)}$. Then, $\mathbb{E}_{x,y,g=1}[g_1g_2g_3] \geq \mathbb{E}[f]^4 = 1/\log^{4a}|G|$. The expectation of terms with $h$ is less than $1/\log^{c/4}|G|$. So the proof can be completed for all sufficiently small $a$.

# 8 Static data structures

In this section we study lower bounds on data structures. First, we define the setting. We have $n$ bits of data, stored in $s$ bits of memory (the data structure) and want to answer $m$ queries about the data. Each query is answered with $d$ probes. There are two types of probes:

- *bit-probe* which return one bit from the memory, and

- *cell-probe* in which the memory is divided into cells of $\log n$ bits, and each probe returns one cell.

The queries can be adaptive or non-adaptive. In the adaptive case, the data structure probes locations which may depend on the answer to previous probes. For bit-probes it means that we answer a query with depth-$d$ decision trees.

Finally, there are two types of data structure problems:

- The *static* case, in which we map the data to the memory arbitrarily and afterwards the memory remains unchanged.

- The *dynamic* case, in which we have update queries that change the memory and also run in bounded time.

In this lecture we focus on the non-adaptive, bit-probe, and static setting. Some trivial extremes for this setting are the following. Any problem (i.e., collection of queries) admits data structures with the following parameters:

- $s = m$ and $d = 1$, i.e. you write down all the answers, and

- $s = n$ and $d = n$, i.e. you can always answer a query about the data if you read the entire data.

Next, we review the best current lower bound, a bound proved in the 80's by Siegel [Sie04] and rediscovered later. We state and prove the lower bound in a different way. The lower bound is for the problem of $k$-wise independence.

**Problem 1.** The data is a seed of size $n = k \log m$ for a $k$-wise independent distribution over $\{0,1\}^m$. A query $i$ is defined to be the $i$-th bit of the sample.

The question is: if we allow a little more space than seed length, can we compute such distributions fast?

**Theorem 2.** For the above problem with $k = m^{1/3}$ it holds that

$$d \geq \Omega\left(\frac{\lg m}{\lg(s/n)}\right).$$

It follows, that if $s = O(n)$ then $d$ is $\Omega(\lg m)$. But if $s = n^{1+\Omega(1)}$ then nothing is known.

*Proof.* Let $p = 1/m^{1/4d}$. We have the memory of $s$ bits and we are going to subsample it. Specifically, we will select a bit of $s$ with probability $p$, independently.

The intuition is that we will shrink the memory but still answer a lot of queries, and derive a contradiction because of the seed length required to sample $k$-wise independence.

For the "shrinking" part we have the following. We expect to keep $p \cdot s$ memory bits. By a Chernoff bound, it follows that we keep $O(p \cdot s)$ bits except with probability $2^{-\Omega(p \cdot s)}$.

For the "answer a lot of queries" part, recall that each query probes $d$ bits from the memory. We keep one of the $m$ queries if it so happens that we keep all the $d$ bits that it probed in the memory. For a fixed query, the probability that we keep all its $d$ probes is $p^d = 1/m^{1/4}$.

We claim that with probability at least $1/m^{O(1)}$, we keep $\sqrt{m}$ queries. This follows by Markov's inequality. We expect to not keep $m - m^{3/4}$ queries on average. We now apply Markov's inequality to get that the probability that we don't keep at least $m - \sqrt{m}$ queries is at most $(m - m^{3/4})/(m - \sqrt{m})$.

Thus, if $2^{-\Omega(p \cdot s)} \leq 1/m^{O(1)}$, then there exists a fixed choice of memory bits that we keep, to achieve both the "shrinking" part and the "answer a lot of queries" part as above. This inequality is true because $s \geq n > m^{1/3}$ and so $p \cdot s \geq m^{-1/4+1/3} = m^{\Omega(1)}$. But now we have $O(p \cdot s)$ bits of memory while still answering as many as $\sqrt{m}$ queries.

The minimum seed length to answer that many queries while maintaining $k$-wise independence is $k \log \sqrt{m} = \Omega(k \lg m) = \Omega(n)$. Therefore the memory has to be at least as big as the seed. This yields

$$O(ps) \geq \Omega(n)$$

from which the result follows. □

This lower bound holds even if the $s$ memory bits are filled arbitrarily (rather than having entropy at most $n$). It can also be extended to adaptive cell probes.

We will now show a conceptually simple data structure which nearly matches the lower bound. Pick a random bipartite graph with $s$ nodes on the left and $m$ nodes on the right. Every node on the right side has degree $d$. We answer each probe with an XOR of its neighbor bits. By the Vazirani XOR lemma, it suffices to show that any subset $S \subseteq [m]$ of at most $k$ memory bits has an XOR which is unbiased. Hence it suffices that every subset $S \subseteq [m]$

with $|S| \leq k$ has a unique neighbor. For that, in turn, it suffices that $S$ has a neighborhood of size greater than $\frac{d|S|}{2}$ (because if every element in the neighborhood of $S$ has two neighbors in $S$ then $S$ has a neighborhood of size $< d|S|/2$). We pick the graph at random and show by standard calculations that it has this property with non-zero probability.

$$\begin{aligned}
&\Pr\left[\exists S \subseteq [m], |S| \leq k, \text{ s.t. } |\mathsf{neighborhood}(S)| \leq \frac{d|S|}{2}\right] \\
&= \Pr\left[\exists S \subseteq [m], |S| \leq k, \text{ and } \exists T \subseteq [s], |T| \leq \frac{d|S|}{2} \text{ s.t. all neighbors of S land in T}\right] \\
&\leq \sum_{i=1}^{k} \binom{m}{i} \cdot \binom{s}{d \cdot i/2} \cdot \left(\frac{d \cdot i}{s}\right)^{d \cdot i} \\
&\leq \sum_{i=1}^{k} \left(\frac{e \cdot m}{i}\right)^{i} \cdot \left(\frac{e \cdot s}{d \cdot i/2}\right)^{d \cdot i/2} \cdot \left(\frac{d \cdot i}{s}\right)^{d \cdot i} \\
&= \sum_{i=1}^{k} \left(\frac{e \cdot m}{i}\right)^{i} \cdot \left(\frac{e \cdot d \cdot i/2}{s}\right)^{d \cdot i/2} \\
&= \sum_{i=1}^{k} \left[\underbrace{\frac{e \cdot m}{i} \cdot \left(\frac{e \cdot d \cdot i/2}{s}\right)^{d/2}}_{C}\right]^{i}.
\end{aligned}$$

It suffices to have $C \leq 1/2$, so that the probability is strictly less than 1, because $\sum_{i=1}^{k} 1/2^{i} = 1 - 2^{-k}$. We can match the lower bound in two settings:

- if $s = m^{\epsilon}$ for some constant $\epsilon$, then $d = O(1)$ suffices,
- $s = O(k \cdot \log m)$ and $d = O(\lg m)$ suffices.

**Remark 3.** It is enough if the memory is $(d \cdot k)$-wise independent as opposed to completely uniform, so one can have $n = d \cdot k \cdot \log s$. An open question is if you can improve the seed length to optimal.

As remarked earlier the lower bound does not give anything when $s$ is much larger than $n$. In particular it is not clear if it rules out $d = 2$. Next we show a lower bound which applies to this case.

**Problem 4.** Take $n$ bits to be a seed for 1/100-biased distribution over $\{0,1\}^m$. The queries, like before, are the bits of that distribution. Recall that $n = O(\lg m)$.

**Theorem 5.** You need $s = \Omega(m)$.

*Proof.* Every query is answered by looking at $d = 2$ bits. But $t = \Omega(m)$ queries are answered by the same 2-bit function $f$ of probes (because there is a constant number of functions on 2-bits). There are two cases for $f$:

1. $f$ is linear (or affine). Suppose for the sake of contradiction that $t > s$. Then you have a linear dependence, because the space of linear functions on $s$ bits is $s$. This implies that if you XOR those bits, you always get 0. This in turn contradicts the assumption that the distributions has small bias.

2. $f$ is AND (up to negating the input variables or the output). In this case, we keep collecting queries as long as they probe at least one new memory bit. If $t > s$ when we stop we have a query left such that both their probes query bits that have already been queried. This means that there exist two queries $q_1$ and $q_2$ whose probes cover the probes of a third query $q_3$. This in turn implies that the queries are not close to uniform. That is because there exist answers to $q_1$ and $q_2$ that fix bits probed by them, and so also fix the bits probed by $q_3$. But this contradicts the small bias of the distribution.

□

# References

[Ajt83] Miklós Ajtai. $\Sigma_1^1$-formulae on finite structures. *Annals of Pure and Applied Logic*, 24(1):1–48, 1983.

[Amb05] Andris Ambainis. Polynomial degree and lower bounds in quantum complexity: Collision and element distinctness with small range. *Theory of Computing*, 1(1):37–46, 2005.

[AS04] Scott Aaronson and Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. *J. of the ACM*, 51(4):595–605, 2004.

[Aus16] Tim Austin. Ajtai-Szemerédi theorems over quasirandom groups. In *Recent trends in combinatorics*, volume 159 of *IMA Vol. Math. Appl.*, pages 453–484. Springer, [Cham], 2016.

[AW89] Miklos Ajtai and Avi Wigderson. Deterministic simulation of probabilistic constant-depth circuits. *Advances in Computing Research - Randomness and Computation*, 5:199–223, 1989.

[Bar89] David A. Mix Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in $NC^1$. *J. of Computer and System Sciences*, 38(1):150–164, 1989.

[Baz09] Louay M. J. Bazzi. Polylogarithmic independence can fool DNF formulas. *SIAM J. Comput.*, 38(6):2220–2272, 2009.

[BCK+14] Harry Buhrman, Richard Cleve, Michal Koucký, Bruno Loff, and Florian Speelman. Computing with a full memory: catalytic space. In *ACM Symp. on the Theory of Computing (STOC)*, pages 857–866, 2014.

[BDPW10] Paul Beame, Matei David, Toniann Pitassi, and Philipp Woelfel. Separating deterministic from randomized multiparty communication complexity. *Theory of Computing*, 6(1):201–225, 2010.

[BF92] László Babai and Peter Frankl. *Linear algebra methods in combinatorics.* 1992.

[BKT17] Mark Bun, Robin Kothari, and Justin Thaler. The polynomial method strikes back: Tight quantum query bounds via dual polynomials. *CoRR*, arXiv:1710.09079, 2017.

[BNP08] László Babai, Nikolay Nikolov, and László Pyber. Product growth and mixing in finite groups. In *ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 248–257, 2008.

[Bra10] Mark Braverman. Polylogarithmic independence fools $AC^0$ circuits. *J. of the ACM*, 57(5), 2010.

[BT17] Mark Bun and Justin Thaler. A nearly optimal lower bound on the approximate degree of AC0. *CoRR*, abs/1703.05784, 2017.

[EGL$^+$92] Guy Even, Oded Goldreich, Michael Luby, Noam Nisan, and Boban Velickovic. Approximations of general independent distributions. In *ACM Symp. on the Theory of Computing (STOC)*, pages 10–16, 1992.

[FSS84] Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984.

[GLS12] Dmitry Gavinsky, Shachar Lovett, and Srikanth Srinivasan. Pseudorandom generators for read-once accˆ0. In *IEEE Conf. on Computational Complexity (CCC)*, pages 287–297, 2012.

[GMR$^+$12] Parikshit Gopalan, Raghu Meka, Omer Reingold, Luca Trevisan, and Salil Vadhan. Better pseudorandom generators from milder pseudorandom restrictions. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, 2012.

[Gow08] W. T. Gowers. Quasirandom groups. *Combinatorics, Probability & Computing*, 17(3):363–387, 2008.

[Gre05a] Ben Green. An argument of Shkredov in the finite field setting, 2005. Available at people.maths.ox.ac.uk/greenbj/papers/corners.pdf.

[Gre05b] Ben Green. Finite field models in additive combinatorics. *Surveys in Combinatorics, London Math. Soc. Lecture Notes 327, 1-27*, 2005.

[Hås87] Johan Håstad. *Computational limitations of small-depth circuits.* MIT Press, 1987.

[Hås14] Johan Håstad. On the correlation of parity and small-depth circuits. *SIAM J. on Computing*, 43(5):1699–1708, 2014.

[IMP12] Russell Impagliazzo, William Matthews, and Ramamohan Paturi. A satisfiability algorithm for $\mathrm{AC}^0$. In *ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 961–972, 2012.

[LMN93] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, Fourier transform, and learnability. *J. of the ACM*, 40(3):607–620, 1993.

[LN90] Nathan Linial and Noam Nisan. Approximate inclusion-exclusion. *Combinatorica*, 10(4):349–365, 1990.

[Nis91] Noam Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica. An Journal on Combinatorics and the Theory of Computing*, 11(1):63–70, 1991.

[Nis92] Noam Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.

[NN90] J. Naor and M. Naor. Small-bias probability spaces: efficient constructions and applications. In *22nd ACM Symp. on the Theory of Computing (STOC)*, pages 213–223. ACM, 1990.

[Raz87] Alexander Razborov. Lower bounds on the dimension of schemes of bounded depth in a complete basis containing the logical addition function. *Akademiya Nauk SSSR. Matematicheskie Zametki*, 41(4):598–607, 1987. English translation in Mathematical Notes of the Academy of Sci. of the USSR, 41(4):333-338, 1987.

[Raz09] Alexander A. Razborov. A simple proof of Bazzi's theorem. *ACM Transactions on Computation Theory (TOCT)*, 1(1), 2009.

[Sie04] Alan Siegel. On universal classes of extremely random constant-time hash functions. *SIAM J. on Computing*, 33(3):505–543, 2004.

[Ta-17] Amnon Ta-Shma. Explicit, almost optimal, epsilon-balanced codes. In *ACM Symp. on the Theory of Computing (STOC)*, pages 238–251, 2017.

[Tal17] Avishay Tal. Tight bounds on the fourier spectrum of AC0. In *Conf. on Computational Complexity (CCC)*, pages 15:1–15:31, 2017.