Antichain codes

Talk by Ben Gunby

Notes by Sanjana Das

May 17, 2023

This is based on joint work with Xiaoyu He, Bhargav Narayanan, and Sam Spiro.

§1 Introduction

Suppose that we have a subset $S \subseteq 2^{[n]}$ of the Boolean cube $2^{[n]}$ (we'll think of elements of the Boolean cube as n-bit strings). There are a few interesting properties that S could have. One is that S could be an *antichain* (with respect to the usual (coordinatewise) ordering, where $\varepsilon \leq \varepsilon'$ if and only if $\varepsilon_i \leq \varepsilon'_i$ for all i). In this case, Sperner's theorem states that $|S| \leq \binom{n}{n/2}$, which means the density of S is $O(n^{-1/2})$ (and of course this is tight — we can achieve it by taking S to be the middle layer of the Boolean cube).

Another interesting property is that S could be a *distance-r code*, meaning that every two elements from S differ in at least r bits. We'll focus on odd-distance codes; it suffices to consider this case because given any distance-$(2r+1)$ code, we can easily turn it into a distance-$(2r+2)$ code of roughly the same size by restricting it to either the elements with an even number of 1's, or those with an odd number. (One of these will have size at least half of the original; and we'll only care about the asymptotic sizes of our codes.)

If S is a distance-$(2r+1)$ code (where we think of r as a constant), then the density of S is $O(n^{-r})$. One way to prove this is by considering the Hamming balls of radius r centered at the points in S — these balls can't intersect, because if two balls with radius r did intersect, then their centers would have distance at most $2r$. And each of these Hamming balls has size on the order of n^r, which gives an upper bound for the density of S on the order of n^{-r}.

This bound is also tight — there are lots of examples of codes with this density (e.g., BCH codes).

So we've seen two very natural conditions that we can impose on a subset of the Boolean cube — being an antichain or a distance-$(2r+1)$ code — and we roughly understand how each condition works *individually*. But what we're interested in is how these conditions *interact*.

> **Question 1.1.** If we know that S is *both* an antichain and a distance-$(2r+1)$ code, then what can we say about its density?

You could imagine a variety of answers to this question, depending on how the two conditions interact.

On one hand, we can get a construction of density $n^{-r-1/2}$ (the product of the densities corresponding to each of the two conditions) by taking a large distance-$(2r+1)$ code (one with density n^{-r}) and intersecting it with the middle layer of the Boolean cube (which is an antichain of density $n^{-1/2}$). This doesn't *exactly* work because our code might not necessarily have lots of elements in the middle layer. But we can easily fix this — any translate of a distance-$(2r+1)$ code is still a distance-$(2r+1)$ code, so we can instead intersect some *translate* of the original code with the middle layer of the Boolean cube. Then on *average* (over all possible translates) this will pick out a $n^{-1/2}$-fraction of our original code, which means one of these translates will give us a construction of density $n^{-r-1/2}$.

On the other hand, of course we have an upper bound of n^{-r} (ignoring the fact that S is an antichain, and just using that it's a distance-$(2r+1)$ code). So we could imagine the correct answer being anywhere from n^{-r} (if the conditions interact nicely, so that being a distance-$(2r+1)$ code makes it much easier to also be an antichain) to $n^{-r-1/2}$ (if they don't interact nicely, so it's hard to do much better than just satisfying each property separately and taking an intersection of the two sets, as above). It turns out that the latter is the truth; and that's the main theorem we'll talk about today.

Theorem 1.2 (Gunby–He–Narayanan–Spiro)
If $S \subseteq 2^{[n]}$ is both an antichain and a distance-$(2r+1)$ code, then

$$\frac{|S|}{2^n} = O(n^{-r-1/2}).$$

§2 A connection to anticoncentration

Question 1.1 is fairly natural — both conditions (of being an antichain and a distance-$(2r+1)$ code) are natural, so it makes sense to wonder how they interact. But there's also another source of motivation for it, coming from *anticoncentration*.

§2.1 Anticoncentration and Halász's theorem

The setup for anticoncentration is as follows — we're given n nonzero real numbers $a_1, \dots, a_n$, and we have n independent and identically distributed random variables $\varepsilon_1, \dots, \varepsilon_n$, which each take the value 1 with probability $1/2$ and 0 otherwise. We're interested in the sum $\sum a_i\varepsilon_i$. In other words, we're looking at the sum of a uniform random subset of $a_1, \dots, a_n$. More specifically, we're interested in the maximum point probability of this sum.

Question 2.1 (Littlewood–Offord)**.** What is $\max_a \mathbb{P}[\sum a_i\varepsilon_i = a]$?

There are a few classic results on this question.

Theorem 2.2 (Erdős)
For all nonzero $a_1, \dots, a_n$ and all a, we have

$$\mathbb{P}\left[\sum a_i\varepsilon_i = a\right] \le \frac{\binom{n}{n/2}}{2^n} = O(n^{-1/2}).$$

This is tight — as an equality case, we can take $a_1 = \cdots = a_n = 1$. Then $\sum a_i\varepsilon_i$ simply picks out the size of our subset (i.e., how many ε_i's are 1); so $\mathbb{P}[\sum a_i\varepsilon_i = a]$ is maximized when $a = n/2$, at which point we get exactly this probability.

But in this construction, the reason it's so easy to make lots of sums $\sum a_i\varepsilon_i$ the same (making this probability large) is that all the a_i's are equal, which means we can just swap them in and out (i.e., if we have some subset sum, we can swap out some term a_i for another term a_j without affecting the sum). So it's natural to ask what happens if we *can't* do this.

Question 2.3. What if all the a_i's are distinct — can we get a better bound?

The answer is yes; this is another classic result.

Theorem 2.4 (Sárkőzy–Szemerédi)

If $a_1, \dots, a_n$ are distinct, then for all a we have

$$\mathbb{P}\left[\sum a_i \varepsilon_i = a\right] = O(n^{-3/2}).$$

This is also tight; as a construction, we can take $a_i = i$ for each i. Then we're taking a random subset sum of $\{1, \dots, n\}$; its variance will be on the order of n^3, which means its standard deviation will be roughly $n^{3/2}$, and therefore we can find some a which gives us a probability of at least $n^{-3/2}$.

Remark 2.5. In fact, it's a classic result (due to Richard Stanley) that this construction (where we take $a_i = i$) is the *exact* optimal case (i.e., the construction where this probability is largest).

In this construction, it's no longer the case that we can swap in any a_i for any other a_j (as in the all-1's case from earlier). But there are still lots of swaps that we *can* make — there's lots of small linear relations between $1, \dots, n$ that allow us to swap some terms in and out. For example, we have $1 + 4 = 2 + 3$, so if we have a subset sum containing 1 and 4, we can swap them out and replace them with 2 and 3.

Question 2.6. What if we forbid 'small linear relations' among the a_i's — can we get a better bound?

The answer is yes; there's another theorem that deals with this more general case.

Theorem 2.7 (Halász)

Let r be an integer, and suppose that there is no index subset $I \subseteq [n]$ of size $|I| \leq 2r$ for which $a_1, \dots, a_n$ satisfy a linear relation of the form $\sum_{i \in I} \pm a_i = 0$. Then

$$\mathbb{P}\left[\sum a_i \varepsilon_i = a\right] = O(n^{-r-1/2}).$$

Intuitively, this condition on $a_1, \dots, a_n$ bans small relations among them that allow us to swap some out and others in. For example, if $r = 1$ then we're banning relations of length at most 2, so we're essentially banning the relations $a_i + a_j = 0$ (which corresponds to swapping out a_i and a_j for nothing) and $a_i - a_j = 0$ (which corresponds to swapping out a_i for a_j). In particular, if $a_1, \dots, a_n$ are positive, then the condition for $r = 1$ corresponds exactly to them being distinct.

Remark 2.8. In general, to swap out some a_i's for others, our linear relation doesn't need to have the same number of $+$'s and $-$'s — we don't need our subset sum to have a fixed length, so the numbers of terms we swap in and out can be different. But it turns out that the 'relevant' linear relations (in the proof) are just the ones with the same number of $+$'s and $-$'s.

Remark 2.9. This isn't actually the full version of Halász's theorem, but it's the version we'll be talking about today. The full version says something stronger — it supposes that we have some number of linear relations of the form $\sum_{i \in I} \pm a_i = 0$, and bounds the probability in terms of the number of linear relations we have.

§2.2 From antichain codes to anticoncentration

The connection to our problem is basically as follows — suppose we have real numbers $a_1, \dots, a_n, a \in \mathbb{R}$. We can then define a set $S \subseteq 2^{[n]}$ corresponding to all possible ways to get a subset sum of exactly a — i.e.,

$$S = \left\{(\varepsilon_1, \dots, \varepsilon_n) \mid \sum \varepsilon_i a_i = a\right\} \subseteq 2^{[n]}.$$

Then $\mathbb{P}[\sum a_i\varepsilon_i = a]$ is precisely the density of S.

There's a bit more we can say about this set S. First, we can assume without loss of generality that $a_1, \dots, a_n$ are all positive. (This is because we can switch from $\{0,1\}$-valued random variables to $\{-1,1\}$-valued ones without changing what the distribution of $\sum a_i\varepsilon_i$ looks like, and with $\{-1,1\}$-valued random variables it's clear that the sign of a_i doesn't matter.)

Claim 2.10 — The set S is an antichain.

Proof. Suppose we have two strings $\varepsilon, \varepsilon' \in 2^{[n]}$ such that $\varepsilon < \varepsilon'$ in the partial order on the Boolean cube. Then we have $\sum a_i\varepsilon_i < \sum a_i\varepsilon_i'$ (the subset sum on the right-hand side simply has more terms than the one on the left-hand side, and all terms are positive). This means ε and ε' can't both be in S, as these two sums can't both be equal to the same value a (since the one on the right is larger). □

Meanwhile, the condition from Halász's theorem implies that S is a $(2r+1)$-code — this is because if we had two strings ε and ε' differing in at most $2r$ bits with $\sum a_i\varepsilon_i = \sum a_i\varepsilon_i'$, then we could cancel out the common terms and end up with a linear relation (with coefficients ± 1) of length at most $2r$ (and the condition states that such relations don't exist).

And the upshot of this is that Theorem 1.2 (on antichain codes) implies Theorem 2.7 (of Halász). This is interesting because the previous proofs of Halász's theorem are Fourier analytic in nature, while the proof of Theorem 1.2 is purely combinatorial; so this allows us to get a combinatorial proof of Halász's theorem. (More precisely, this gives a combinatorial proof of the version of Halász's theorem where there are *no* small relations; it's still open whether we can get the full theorem, where we allow a small number of linear relations and get a slightly weaker bound.)

§3 The proof of Theorem 1.2

The proof of Theorem 1.2 has two parts. We'll first motivate what the main lemma is, and then if there's time, we'll discuss the proof of this main lemma.

§3.1 Shadows

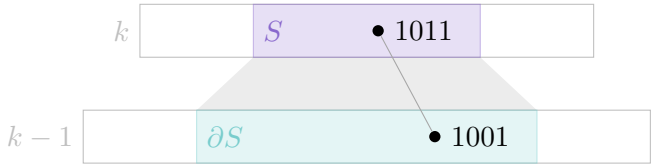
First, we'll need a few definitions.

Definition 3.1. The kth layer of the Boolean cube $2^{[n]}$ is defined as $\binom{[n]}{k} = \{\varepsilon \in 2^{[n]} \mid \sum \varepsilon_i = k\}$.

Definition 3.2. For a subset $S \subseteq \binom{[n]}{k}$, we define its shadow as

$$\partial S = \left\{ x \in \binom{[n]}{k-1} \;\middle|\; x \text{ lies below an element of } S \right\}.$$

In other words, we're taking a set S that lives in the kth layer of the Boolean cube, and we're looking at all the elements immediately below it.



One useful fact about shadows is that they grow (at least, in the top half of the cube).

Lemma 3.3
If $k > n/2$ and $S \subseteq \binom{[n]}{k}$, then $|\partial S| \geq |S|$.

The proof is a double-counting argument, where we consider the number of ways to go down from S to its shadow and the number of ways to go up from the shadow to S.

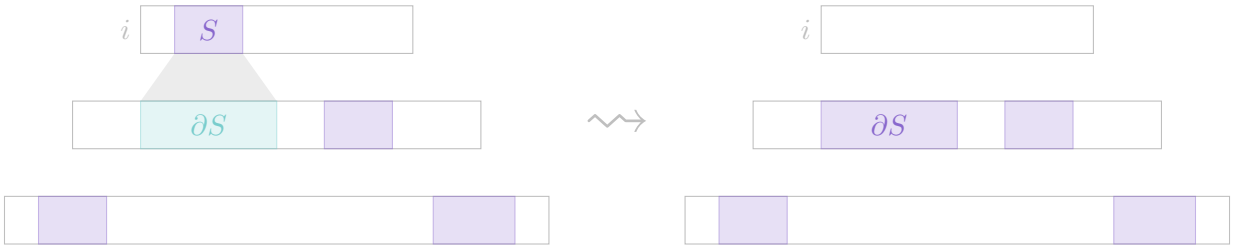
§3.2 A proof of Sperner's theorem

We'll first give a proof of Sperner's theorem, which is essentially the $r = 0$ case of Theorem 1.2; we're doing this because our proof of Theorem 1.2 will be a modified version of this argument, and it'll be useful to describe exactly how it'll change.

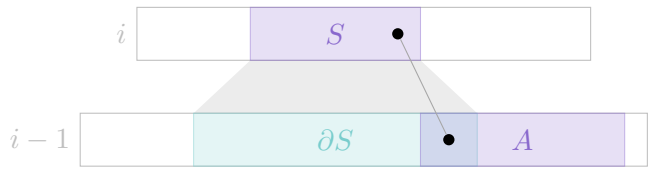
Theorem 3.4 (Sperner)
Any antichain in $2^{[n]}$ has size at most $\binom{n}{n/2}$.

Proof. Let $T \subseteq 2^{[n]}$ be an antichain. Let i be the highest layer of the Boolean cube containing elements of T, and let $S = T \cap \binom{[n]}{i}$ be the intersection of T with the ith layer.

For now, suppose that $i > n/2$. We then replace S with its shadow ∂S — so we take T, swap out S, and swap in its shadow.

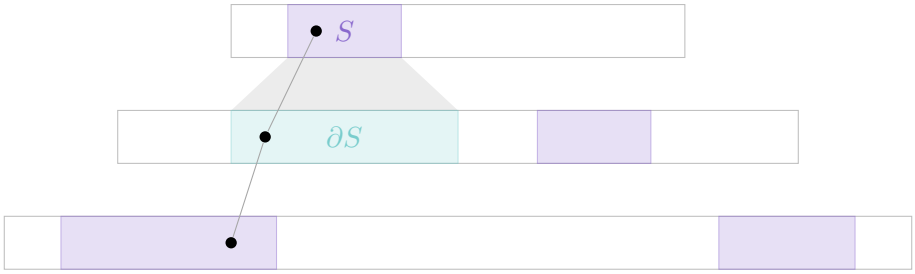


Note that if we let $A = T \cap \binom{[n]}{i-1}$ be the subset of T that lives in the $(i-1)$th layer, then A and ∂S must be disjoint — otherwise we'd have an element of A lying below an element of S, contradicting the fact that T is an antichain (which means none of its elements can lie below another).



Then when we swap out S for its shadow, the fact that $|\partial S| \geq |S|$ (from Lemma 3.3) means that we're adding more elements than we're taking away.

Meanwhile, our set T remains an antichain — if there were some element in ∂S that lay above an element of T, then the element of S that it came from would lie above the same element of T (contradicting the fact that our original set was an antichain).



So this (i.e., replacing S with ∂S) preserves the condition that T is an antichain and doesn't decrease its size, but it wipes out everything in layer i and replaces it with stuff in layer $i-1$. Then we can repeat this until we get down to layer $n/2$ — so we've flattened down all elements that were initially above layer $n/2$ down to layer $n/2$. (We have to stop at layer $n/2$ because Lemma 3.3 stops being true beyond that.)

And the Boolean cube is symmetric, so we can do the same for everything below layer $n/2$ (i.e., we can push all those elements up to layer $n/2$). Then we've squished our entire set into layer $n/2$, which means it has size at most $\binom{n}{n/2}$. □

Now let's zoom in on what happens at one step of this algorithm, to figure out what facts we really needed to make this argument work. In one step, we'll begin with a set S (living in the layer i) — in the first step S will be a subset of T, but after several steps, S will basically consist of the accumulated shadows of stuff in T (by the time we reach the ith layer, we've taken all the elements of T that were originally above it and flattened them down to this layer). Meanwhile, A is the subset of T living in the layer $i-1$.

Then our algorithm replaces S with its shadow. Now the top layer of our new set is in layer $i-1$, so it accumulates A as well — this means the set S we'll be using on the next step of the algorithm is $\partial S \cup A$.

And the key property we need is that

$$|\partial S \cup A| \geq |S| + |A|\,. \tag{1}$$

This property means that that when we go from the ith layer to the $(i-1)$st layer, we've successfully accumulated all our points — we're not decreasing the size of our set. (In this proof, (1) was true just because ∂S and A are disjoint and $|\partial S| \geq |S|$.)

§3.3 Modifications for the $r = 1$ case

We'll now discuss how to modify this argument to prove Theorem 1.2. For the rest of this talk, we'll focus on the case $r = 1$ — so we're looking at distance-3 codes, and the bound we want to prove is $n^{-3/2}$.

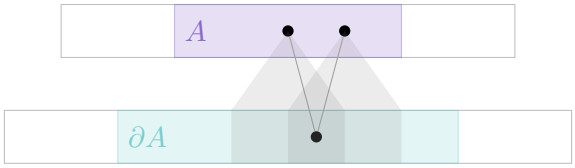
First, we can assume without loss of generality that our entire set lives in layer $n/2 + o(n)$ (this is because almost all of the Boolean cube is close to the middle as $n \to \infty$, so elements far from the middle contribute very little to the density of our set).

When we try to run the same argument as before, what's the problem? The argument does work as written, but the problem is that it still gives a bound of $n^{-1/2}$, while the bound we actually want is $n^{-3/2}$. So we need to somehow get another factor of n.

One potential way we could imagine gaining this factor of n is if we could gain a factor of n when we add A in (1) — i.e., if we had $|\partial S \cup A| \geq |S| + n\,|A|$. Of course, this is not true. But what *is* true is that A is a distance-3 code (note that S isn't necessarily a distance-3 code, since it's an accumulated shadow rather than a subset of our original antichain code; but A is). And it turns out that if you take the *shadow* of a distance-3 code, it actually expands by quite a lot. So we'll actually look at the shadow of A.

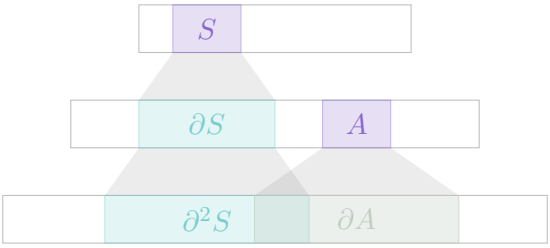
Claim 3.5 — We have $|\partial A| = \Omega(n)\,|A|$.

Proof. First, no point in ∂A can be in the shadow of two distinct elements of A, or else those two elements would have distance 2.



So the shadows of every two elements of A are disjoint; and since each of these shadows has size roughly $n/2$, this means their total size accumulates a factor of roughly $n/2$ (compared to the size of A). □

So in one step, if we just pushed S down *one* layer then we'd only accumulate A; but if we push go down one more layer (so that we're pushing A down), then A *does* accumulate a factor of n. This is the good news; but the bad news is that if we do so, we'll also have to push down S a second layer, and this will no longer necessarily be disjoint from the shadow of A. (The condition that our set is an antichain means that A can't overlap the shadow of S, but it doesn't say anything about how the *shadow* of A overlaps with the shadow of the shadow of S.) So even though we do gain a factor of n when we push A down, we don't know how much of this gain is encompassed by what we got from pushing S down.



But for the argument to work, we don't actually need something as strong as disjointness to hold — we really only need a statement of the form

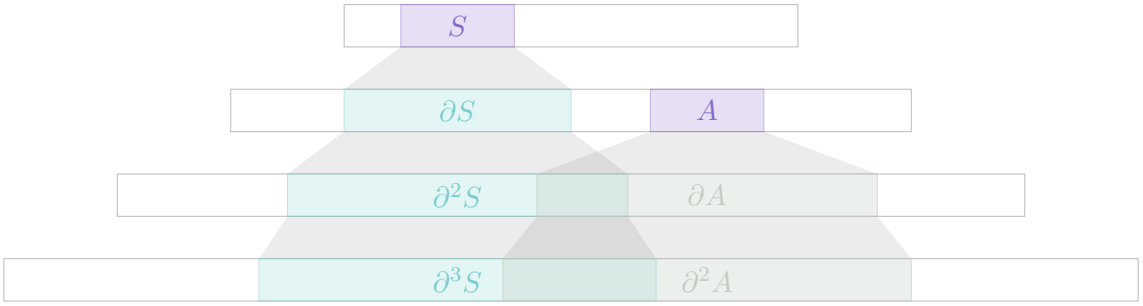
$$|\partial^2 S \cup \partial A| \geq |S| + \Omega(n)\,|A|\,.$$

Unfortunately, we don't know how to prove this (the authors originally tried this and had a lot of trouble; it's either false or quite hard to prove). But it turns out that it's actually easier to prove if we go down one *more* layer. So the main lemma is as follows.

Lemma 3.6

Let $k = n/2 + o(n)$, and let $S \subseteq \binom{[n]}{k}$ and $A \subseteq \binom{[n]}{k-1}$ be sets such that A is a distance-3 code and A is disjoint from S. Then we have

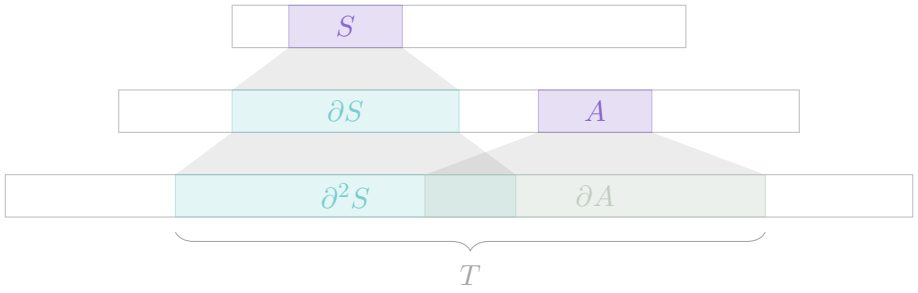
$$|\partial^3 S \cup \partial^2 A| \geq |S| + \Omega(n)\,|A|\,.$$



Once we have this main lemma, it's fairly easy to finish the proof — we repeatedly push things down, so that at each step, we're looking at the accumulated shadow of everything down to some layer i. And we know that when we push a set A down by 3 layers, its contribution grows by a factor of n; this gives us the additional factor of n that we want.

§3.4 Proof sketch of Lemma 3.6

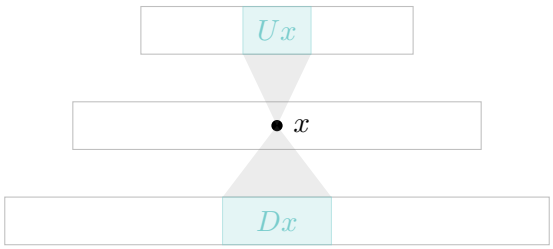
We'll now very quickly sketch how Lemma 3.6 is proven. In our picture, we're starting with a set S and a distance-3 code A that's disjoint from the shadow of S. Let T be everything on the third layer down — i.e., we define $T = \partial^2 S \cup \partial A$.



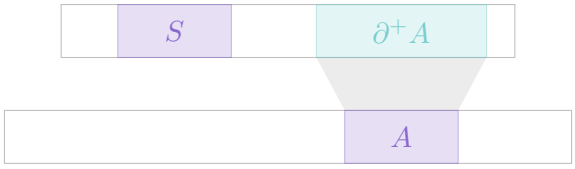
We now define two operators D and U as follows.

Definition 3.7. Suppose that x is a random variable taking values in $\binom{[n]}{k}$.

- We then define Dx as the random variable taking values in $\binom{[n]}{k-1}$ obtained by first sampling x, and then randomly choosing one of the elements directly below x.
- Similarly, we define Ux as the random variable taking values in $\binom{[n]}{k+1}$ obtained by first sampling x, and then randomly choosing one of the elements directly *above* x.



The source of all our issues is that the shadow of A is not necessarily disjoint from the second shadow of S. But something nice happens if we look at the *up*-shadow of A, which we denote by $\partial^+ A$ (defined as the set of all elements directly above an element of A). The up-shadow of A *does* have to be disjoint from S — this is because if an element of the up-shadow were also in S, then we'd have an element of S larger than an element of A (contradicting the antichain property).



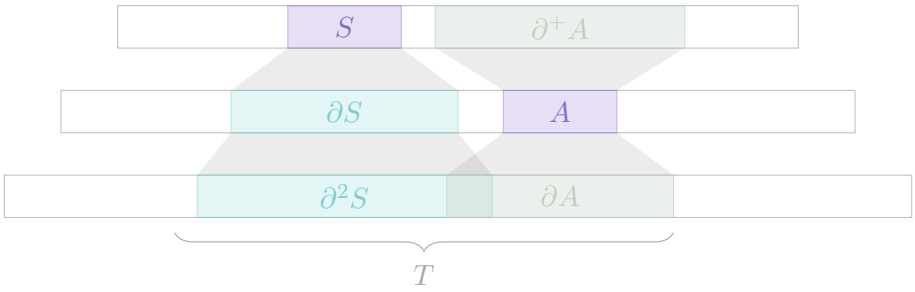
And the same proof of Claim 3.5 (that the shadow of A is roughly n times the size of A) can be used to get the same statement for the *up*-shadow of A, so we have

$$|\partial^+ A| \geq \Omega(n)\,|A|$$

as well, and since S and $\partial^+ A$ are disjoint, this means

$$|S \cup \partial^+ A| \geq |S| + \Omega(n)\,|A|\,.$$

Now if $|T| \geq |S| + \Omega(n)\,|A|$, then we're done (the set whose size we're trying to bound is the shadow of T, and in this case we can just use the bound $|\partial T| \geq |T|$ from Lemma 3.3). So we assume $|T| = |S| + o(n)\,|A|$.

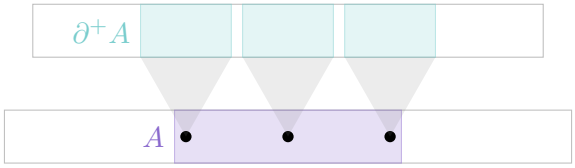


Then the set $S \cup \partial^+ A$ at the top of the picture has size $|S| + \Omega(n)\,|A|$, while the set $T = \partial^2 S \cup \partial A$ at the bottom has size $|S| + o(n)\,|A|$, which means it's much smaller. Now we can imagine taking the second shadow of the set on the top (which pushes it down to the same level as T); the shadow of everything in S has to land inside T, so a bunch of elements in the shadow of $\partial^+ A$ must land *outside* T (since the set on the top is much larger than T). More precisely, if we start with a random element of $\partial^+ A$ and go down two steps, we end up outside T with high probability.

Notation 3.8. For any set S, we use x_S to denote the random variable which is uniform on S.

Then in this notation, the above statement means that $D^2 x_{\partial^+ A} \notin T$ with high probability.

And since A is a code, we can generate a uniform random element of $\partial^+ A$ by starting with a uniform random element of A and then going up — we're not going to get any repeats because A is a code (i.e., each element of $\partial^+ A$ can only come from one element of A).

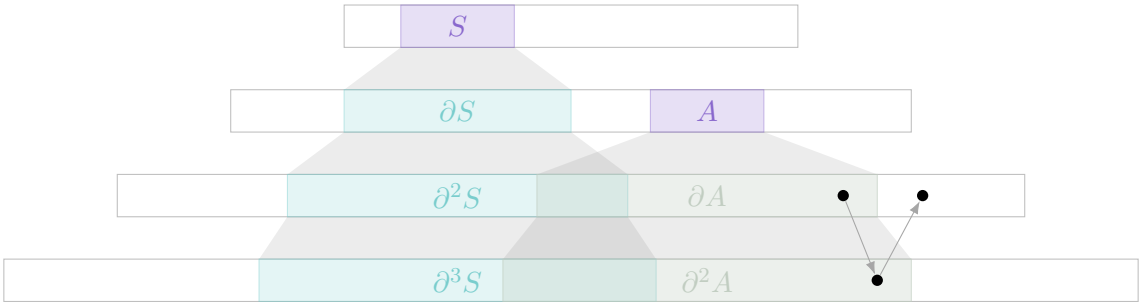


So we can rewrite $x_{\partial^+ A}$ as $U x_A$; then this means we have $D^2 U x_A \notin T$ with high probability.

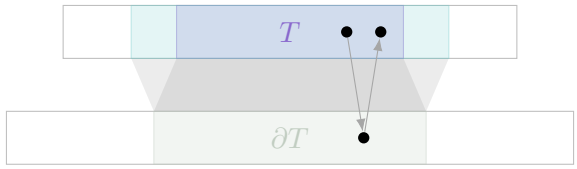
But the operators U and D almost commute — if if we think of elements of the Boolean cube as bit-strings, then U flips a random bit from 0 to 1, while D flips a random bit from 1 to 0; so applying U and then D

is the same as applying D and then U unless the two bits we choose to flip are the same, which is very unlikely. This means D^2Ux_A is almost the same thing as UD^2x_A, so $UD^2x_A \notin T$ with high probability as well. And Dx_A is a uniform random element of the shadow of A, so we can rewrite it as $x_{\partial A}$; then we have $UDx_{\partial A} \notin T$ with high probability. So if we start with a random element of ∂A and then go down and then up, we end up outside of T with high probability.

What if instead of starting with a random element in ∂A, we started with a random element of T? (In other words, we're considering the probability that $UDx_T \notin T$.) On one hand, the probability this happens is at least roughly $|\partial A|/|T|$ (since if the random element we start with is actually in ∂A — which happens with probability $|\partial A|/|T|$ — then the probability we land outside T is nearly 1.) So if we start with a random element in T and go down and then up, then there's a substantial probability we end up outside of T.



Recall that we're trying to show that ∂T is large. Intuitively, if ∂T were roughly the same size as T, then since every way of going down from T lands in ∂T, then by double counting, almost every way of going *up* from ∂T should land back in T.



In other words, this means that if going down and then up from T means we end up outside T with reasonably high probability (here, that probability is roughly $|\partial A|/|T|$), then T must grow when we take its shadow. The quantitative statement we get from this is that

$$\frac{|\partial T|}{|T|} \geq 1 + \Omega\left(\frac{|\partial A|}{|T|}\right).$$

This rearranges to

$$|\partial T| \geq |T| + \Omega(|\partial A|) \geq |S| + \Omega(n)\,|A|$$

(using the facts that $|T| \geq |S|$ and $|\partial A| = \Omega(n)\,|A|$), which is precisely what we wanted.

Remark 3.9. To prove Theorem 1.2 for larger values of r, we go down more layers — instead of using the fact that pushing A down one layer expands it by a factor of n, we use that pushing it down r layers expands it by a factor of n^r.

Remark 3.10. In this proof, when we use the property that any two elements of our original set had distance at least 3, we're only using it *within* a layer; this means that for Halász's theorem, it actually suffices to just consider linear relations with the same number of positive and negative terms (i.e., ones that keep us within the same layer), as mentioned in Remark 2.8.