CS 229r Essential Coding Theory April 20, 2020

Lecture 22

Instructor: Madhu Sudan *Scribe: Elbert Du*

1 Coding in Complexity Theory

1.1 Pseudorandomness

Goal: We wish to be able to use a pseudorandom generator G which takes as input a small number of random bits and turn it into a large number of bits indistinguishable from random.

Definition 1. *We define the generator G to be ε-fooling if*

$$P_R[A(x,R)=1] \approx_\varepsilon \Pr_z[A(x,G(z))=1]$$

In general, we would want for all poly time A but here we will only require it for a single poly time A.

The main parameter of G is then the seed length (the length of z in the equation above).

Today, we'll be using codes to get an approximate solution for Max t-SAT.
Input: $C_1, \dots C_m$ where we have variables $x_1, x_2, \dots x_n$ and $C_j = y_1 \vee y_2 \vee \cdots \vee y_t$ where $y_i = \neg x_k$ or $y_i = x_k$ for some k.
Output: find assignment satisfying as many clauses as possible.

Random assignment: satisfies $(1-2^{-t})\,m$ clauses on expectation since each literal has $\frac{1}{2}$ chance to be satisfied independent of the others within a clause.

Definition 2. *A generator $G : \{0,1\}^S \to \{0,1\}^n$ is t-wise independent if for all $T \subseteq [n], |t| \le t$, then*

$$G(z)|_T \simeq Unif(\{0,1\}^T)$$

when $z \sim Unif(\{0,1\}^S)$

If $\exists$ a t-wise independent generator G, then we get a $m(1-2^{-t})$ approximator to Max t-SAT in time $2^S m$ by just testing out all the seeds.

Lemma 3. *Given a linear code $C \subseteq \mathbb{F}_2^n$ with $\Delta(C^\perp) > t$, then the encoding function E of C is t-wise independent.*

Proof. We know from before that $E(m)|_S$ is uniform on the image for any $S \subseteq [n]$. Now, we claim that if $\Delta(C^\perp) > t$,

$$Im(E(m)|_T) = \{0,1\}^T$$

which would complete our proof.

Suppose for the sake of contradiction some $x \in \{0,1\}^T$ is not in the image. Then, there is some codeword with weight t in $C^\perp$ since it is generated by the parity check matrix of C. This violates the assumption that $C^\perp$ has distance greater than t so such an x does not exist. Hence, all $x \in \{0,1\}^T$ is in the image and we are done. □

Thus, we want a small code such that the dual has distance t. What's the best we can do?

Suppose $C^\perp$ is the BCH code of distance $t+1$. Then, we get parameters $C^\perp = [n, n-(\frac{t}{2})\log n, t+1]_2$ so $C = [n, (\frac{t}{2})\log n, t+1]_2$ so $S = (\frac{t}{2})\log n$

Therefore this gives us a deterministic $n^{t/2}m$ time approximation algorithm for max t-SAT

Now, can we do any better? To do so, we will introduce a notion of bias:

Definition 4. *$G : \mathbb{F}_2^S \to \mathbb{F}_2^n$ is ε-biased if for every linear function $\mathcal{L} : \mathbb{F}_2^n \to \mathbb{F}_2$, we have*

$$\left| \Pr_{x \sim Unif}[\mathcal{L}(x) = 1] - \Pr_{z \sim Unif}[\mathcal{L}(G(z)) = 1] \right| \leq \varepsilon$$

We say that G is ε-biased on $\mathcal{L}$ if it satisfies the definition for the particular function $\mathcal{L}$

Definition 5. *$G : \{0,1\}^S \to \{0,1\}^n$ is δ-almost t-wise independence if $\forall T \subseteq [n], |T| \leq t$ we have*

$$\{G(z)|_T\}_{Z \sim Unif} \approx_\delta Unif(\{0,1\}^T)$$

Note: using one of these gives us a $(1 - 2^{-t} - \delta)$ approximation in time $2^S m$ for the same reason as above

Lemma 6. *If $G : \mathbb{F}_2^S \to \mathbb{F}_2^n$ is ε-biased, then $G(z) \approx_{2^n \varepsilon} Unif(\{0,1\}^n)$*

Proof. So far, we have for any nonzero linear function $\mathcal{F}$,

$$\left| \Pr[\mathcal{L}(G(z)) = 1] - \frac{1}{2} \right| = |\Pr[\mathcal{L}(G(z)) = 1] - \Pr[\mathcal{L}(x) = 1]| \leq \varepsilon$$

So $\left| E_z[(-1)^{L(G(z))}] \right| \leq 2\varepsilon$

What we want:

$$\sum_y \left| \Pr[G(z) = y] - \frac{1}{2^n} \right| \leq 2\varepsilon 2^n$$

If we can show that each value in the sum is at most 2ε, we're done.

Now, let

$$\delta_0(x) = \frac{1}{2^n} \sum_{\mathcal{L}} (-1)^{\mathcal{L}(x)}$$

Note that $\delta_0(x) = 1$ if $x = 0$. However, if $x \neq 0$ then all the terms cancel out so $\delta_0(x) = 0$.

Similarly, we can define $\delta_y(x) = \frac{1}{2^n} \sum_{\mathcal{L}} (-1)^{\mathcal{L}(x-y)}$ and get $\delta_y(y) = 1$ and $\delta_y(x) = 0$ for $x \neq y$

Now, we want to show

$$\left| \Pr[G(z) = y] - \frac{1}{2^n} \right| \leq 2\varepsilon$$

which is true iff

$$\left| \mathbb{E}[\delta_y(G(z)] - \frac{1}{2^n} \right| \leq 2\varepsilon$$

and by the definition of δ_y, this means what we want is

$$\left|\mathbb{E}[(-1)^{\mathcal{L}(G(z)-y)}] - \frac{1}{2^n}\right| \leq 2\varepsilon$$

and as we work through the equations, we eventually arrive at a statement which follows from G being ε-biased

□

Lemma 7. *If G is ε-biased, then $\forall$ t G is $(\varepsilon 2^t)$-almost t-wise independent.*

Proof. Since G is ε-biased, then $\forall T$ $G|_T$ is ε-biased so G is almost uniform. □

Now, if we just set $\delta < \frac{1}{2m}$ and $\varepsilon \leq \frac{1}{2^t 2m}$, then the δ-almost t-wise independence won't give us any margin for error and we get the desired generator.

Definition 8. *A code is ε-balanced if the distance is between $(\frac{1}{2} \pm \varepsilon)N$*

Lemma 9. *If G is a generator of ε-balanced code, then the encoding given my $i \to i^{th}$ column of G is ε-biased.*

Proof. Code is generated by $k \times N$ matrix M, take the i^{th} column to get a generator $G : [N] \to \mathbb{F}_2^K$.

What does linear test look like? Consider multiplying a vector $[\alpha_1, \ldots \alpha_k]$ by M, we get αM, the encoding of the vector. For it to be balanced, the proportion of 1s and 0s must be within $\frac{1}{2} \pm \varepsilon$ so in particular the i^{th} coordinate should be uniform over the choice of α □

Conclusion: we previously constructed explicit codes w/ $N = \frac{k}{\varepsilon^3}$ and $\frac{k^2}{\varepsilon^2}$ so for our setting, set $K = n$ and $\varepsilon = \frac{1}{2^t m}$

$2^S = N = 2^{2t} m^2 n^2 = O(2^{2t} m^4)$ so runtime is $O(2^{2t} m^5)$

Now, what if we combined the two things we constructed today?

Lemma 10. *Let G_1 be ε-biased and G_2 be t-wise independent and linear. Then $G = G_2 \circ G_1$ is $\varepsilon 2^t$-almost t-wise independent.*

Proof. Since G_2 is t-wise independent, $G_2(w)|_T$ has no non-trivvial dependencies among the output regardless of distribution of w.

Now, if G_2 fools some linear $\mathcal{L}$, then $\mathcal{L}(G_2(w))$ is some parity of bits of w so

$$\mathcal{L} \neq 0 \implies (\mathcal{L} \circ G_2) \neq 0$$

Furthermore, we know that $\mathcal{L} \circ G_2$ is a linear test since both are linear and G_1 fools all linear tests so $(G_2 \circ G_1)|_T$ is ε-biased. □

Applying this we get a runtime of $\frac{2^{2t}}{\varepsilon^2} \log^2 n \cdot n$ for a $(1 - 2^{-t} - \varepsilon)$-approximation

2 Exercises

1. How can you use random assignment to get a similar bound on the number of clauses satisfied in Max t-SAT with high probability?

   Solution: Consider the probability that we get a value greater than $(1 - 2^{-t} - \varepsilon)m$.

If we have p probability of getting greater than $(1-2^{-t}-\varepsilon)m$ and $1-p$ of getting at most $(1-2^{-t}-\varepsilon)m$, then we need the expected value to be $(1-2^{-t})m$ so we have

$$pm + (1-p)(1-2^{-t}-\varepsilon)m \geq (1-2^{-t})m$$

$$\frac{p}{1-p} \geq 2^t\varepsilon$$

if we have $\varepsilon = 2^{-t}$, then $p \geq \frac{1}{2}$. We can now repeat this process k times and take the maximum for a probability of $1-2^{-k}$ of getting a result of at least $(1-2^{1-t})m$

2. Complete the proof of lemma 6.

   Solution: in the notes Madhu posted.

   ε-bias of G means that the difference between distribution of $\mathcal{L}(G(z)-y)$ and $\mathcal{L}(x-y)$ for x chosen uniformly is ε. Each difference in the distributions can change the value of the sum by at most 2 as it either changes a value from -1 to 1 or vice versa or it does nothing and the number of differences is at most ε.

   This tells us that the difference between $\mathbb{E}[(-1)^{\mathcal{L}(G(z)-y)}]$ and $\mathbb{E}[(-1)^{\mathcal{L}(x-y)}]$ is at most 2ε. Over random x $\Pr[\mathcal{L}(x-y)=0] - \Pr[\mathcal{L}(x-y)=1] = 0$ if $\mathcal{L}$ is non-zero and 1 if $\mathcal{L}$ is zero so

   $$\mathbb{E}[(-1)^{\mathcal{L}(x-y)}] = \Pr[\mathcal{L}=0] = \frac{1}{2^n}$$

   which gives us the desired result.