U.C. Berkeley — CS278: Computational Complexity Handout N20
Professor Luca Trevisan 10/31/02
Scribe: Stephen Sorkin

Notes for Lecture 20

1 Summary of Previous Lecture

1.1 Boolean Fourier Analysis

Recall the definition of the linear function $L_S : \{-1,1\}^n \to \{-1,1\}$ where $S \subseteq \{1,\ldots,n\}$:

$$L_S(x_1,\ldots,x_n) = \prod_{i\in S} x_i$$

The set of all such functions is a basis for all functions $F : \{-1,1\}^n \to \mathbb{R}$. Now consider any $F : \{-1,1\} \to \{-1,1\}$ we then have:

$$F(x_1,\ldots,x_n) = \sum_S \hat{F}_S L_S(x_1,\ldots,x_n)$$

Where $\sum_S \hat{F}_S^2 = 1$ and $\Pr_x[F(x) = L_S(X)] = \frac{1}{2} + \frac{1}{2}\hat{F}_S$.

1.2 Long Code

The long code is a function LC that takes $\log k$ bits and produces output of length 2^k. We think of a codeword of the long code as a function mapping k-bits strings into a bit, that is, $LC : \{1,\ldots,k\} \longrightarrow (\{0,1\}^k \to \{0,1\})$. The definition of the long code is as follows.

$$\begin{aligned} LC(a) &= A : \{0,1\}^k \to \{0,1\} \\ A(x) &= x[a] \\ A &= l_{\{a\}} \end{aligned}$$

2 Test for Long Code

We now consider a cost to distinguish codewords of the long code. Given $A : \{0,1\}^k \to \{0,1\}$ pick $x, y \in \{0,1\}^k$ uniformly and $z \in \{0,1\}^k$ where the bits of z are IID with $\Pr[z[i] = 1] = \epsilon$. Accept iff $A(x) \oplus A(y) = A(x \oplus y \oplus z)$.

If A is a codeword of the long code, then $A = l_{\{a\}}$ and the test check if $x[a] \oplus y[a] = x[a] \oplus y[a] \oplus z[a]$ and succeeds iff $z[a] = 0$. Hence it succeds with probability $1 - \epsilon$.

We now show that if the test accepts with probability $1/2 + \delta$, then list decoding of the codeword produces a list with size dependent on δ rather than k. Intuitively this is similar to the linearity test of the last lecture.

Consider the case where A is a linear function with large support: $A = l_S$, S is large. Then the test checks if $l_S(x) \oplus l_S(y) = l_S(x) \oplus l_S(y) \oplus l_S(z)$ and will succeed iff $l_S(z) = 0$.

The probability of this occurring is $\frac{1}{2} + \frac{1}{2}(1-2\epsilon)^{|S|}$. So the test cannot accept with high probability if A is not linear or if A is linear but has large support.

We now consider the equivalent setting where we replace 0 with 1, 1 with -1, and $\oplus$ with $\times$ so that $A : \{-1,1\} \to \{-1,1\}, x, y, z \in \{-1,1\}^k$. Here the test accepts iff $A(x)A(y)A(xyz) = 1$. Specifically, $\Pr[\text{test accepts } A] = \frac{1}{2} + \frac{1}{2}E_{x,y,z}[A(x)A(y)A(xyz)]$ where:

$$
\begin{aligned}
&E_{x,y,z}[A(x)A(y)A(xyz)] \\
&\quad = E_{x,y,z}\left[\left(\sum_S \hat{A}_S L_S(x)\right)\left(\sum_T \hat{A}_T L_T(y)\right)\left(\sum_U \hat{A}_U L_U(x) L_U(y) L_U(z)\right)\right] \\
&\quad = \sum_{S,T,U} \hat{A}_s \hat{A}_T \hat{A}_U \overbrace{(E_x[L_S(x)L_U(x)])}^{L_S \cdot L_U} \overbrace{(E_y[L_T(y)L_U(y)])}^{L_T \cdot L_U} (E_z L_U(z)) \\
&\quad = \sum_S \hat{A}_S^3 E_z[L_S(z)] \\
&\quad = \sum_S \hat{A}_S^3 \prod_{i \in S} E[z_i] \\
&\quad = \sum_S \hat{A}_S^3 (1-2\epsilon)^{|S|} \\
&\quad \leq \left(\max_S \hat{A}_S (1-2\epsilon)^{|S|}\right) \cdot \overbrace{\sum_S \hat{A}_S^2}^{=1}
\end{aligned}
$$

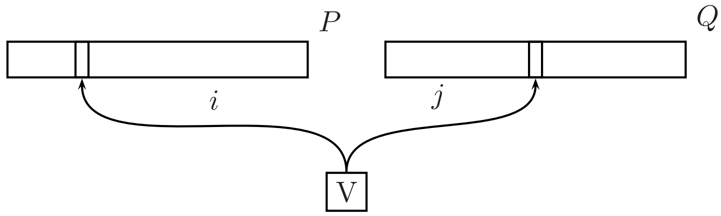
Now suppose that the probability that the test accepts is greater than $\frac{1}{2} + \delta$. Then there is some S where $\hat{A}_S(1-2\epsilon)^{|S|} \geq 2\delta$. Hence:

$$
\begin{aligned}
\hat{A}_S &\geq 2\delta \\
(1-2\epsilon)^{|S|} &\geq 2\delta \\
|S| \log \frac{1}{1-2\epsilon} &\leq \log \frac{1}{2\delta} \\
|S| &\leq \frac{\log \frac{1}{2\delta}}{\log \frac{1}{1-2\epsilon}}
\end{aligned}
$$

This is close to the the analysis of Håstad's proof system. The primary difference is that Håstad's system looks at two strings rather than one.

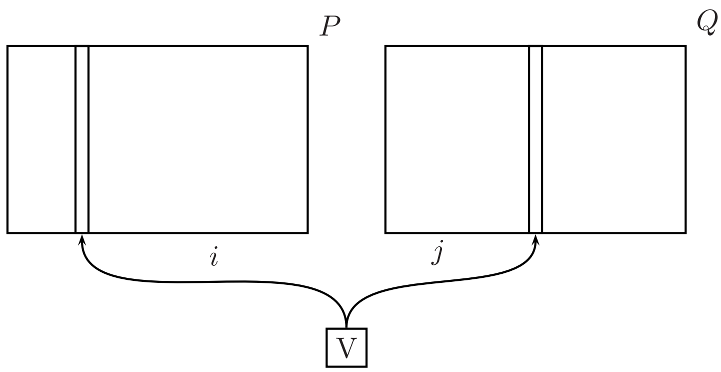
3 Håstad's verifier

First recall Raz's verifier:

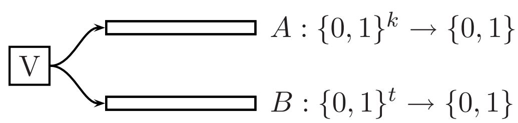


The proof P is over the alphabet $\Sigma, |\Sigma| = k$ and the proof Q is over the alphabet $\Gamma, |\Gamma| = t$. The verifier takes the character at position i from P and the character j from Q, where j is selected at random and i is selected at random from a few positions of P. The verifier accepts iff $P[i] = \pi(Q[i])$.

Now consider Håstad's verifier:



In the proofs for Håstad's verifier, the binary strings indexed by i, j are the long code encodings of the characters at those positions in Raz's Verifier. Håstad's verifer selects i, j the same way as Raz's so that the problem can be modeled as:



Here the function $\pi : \{1, \ldots, t\} \to \{1, \ldots, k\}$ is as before. The verifier will accept with probability greater than $1 - \epsilon$ if A is the long code of $P[i]$, B is the long code of $Q[j]$ and $\pi(Q[j]) = P[i]$. We will show that if the test accepts with probability $\frac{1}{2} + \epsilon$ then $\Pr[D(A) = \pi(D(B))] \geq \text{poly}(\epsilon)$.

We digress for a moment to consider a subtle point about the long code test. What if $S = \emptyset$? This is the case where the function is 0 everywhere. Then the test passes with probability 1. This would not be a problem if the function is balanced. A function A is self-dual if:

$$\begin{aligned} A(1-x) &= 1 - A(x) \text{ when } x, A(x) \in \{0,1\} \\ A(-x) &= -A(x) \text{ when } x, A(x) \in \{-1,1\} \end{aligned}$$

For example, $\oplus$ of an even number of inputs is not self-dual while $\oplus$ of an odd number of inputs is self-dual. If A is self-dual then $\hat{A}_S = 0$ if $|S|$ is even. Consider the following procedure called "folding" to convert any function $A : \{0,1\}^k \to \{0,1\}$ into a self dual function $A' : \{0,1\}^k \to \{0,1\}$:

$$A'(x) = \begin{cases} A(x) & \text{if } x[1] = 0 \\ 1 - A(1-x) & \text{if } x[1] = 1 \end{cases}$$

This is self-dual since:

$$A'(1-x) = \begin{cases} 1 - A(x) = 1 - A'(x) & \text{if } x[1] = 0 \\ A(1-x) = 1 - A'(x) & \text{if } x[1] = 1 \end{cases}$$

In our case of interest, the long code, then $A' = A$. In the following analysis, all testing and decoding is performed on A', B'.

Given $A : \{0,1\}^k \to \{0,1\}, B : \{0,1\}^t \to \{0,1\}, \pi : \{1,\ldots,t\} \to \{1,k\}$ we pick $x \in \{0,1\}^k, y \in \{0,1\}^t$ uniformly. We also pick $z \in \{0,1\}^t$ where the bits are IID with $\Pr[z[i] = 1] = \epsilon$. The test accepts iff:

$$A'(x) \oplus B'(y) = B'((x \circ \pi) \oplus y \oplus z)$$

(From now on we will stop distinguishing between A and A' and between B and B'.)

Treating vectors as functions, $x \circ \pi \in \{0,1\}^t, (x \circ \pi)[i] = x[\pi(i)]$. In the case where the proof is based on a valid proof for Raz's verifier, we have $a = P[i], b = Q[j], A = l_{\{a\}}, B = l_{\{b\}}, \pi(b) = a$ and the test checks if:

$$x[a] \oplus y[b] = x[\pi(b)] \oplus y[b] \oplus z[b] = x[a] \oplus y[b] \oplus z[b]$$

And hence the test accepts with probability $1 - \epsilon$. Considering the equivalent setting of $A : \{-1,1\}^k \to \{-1,1\}, B : \{-1,1\}^t \to \{-1,1\}$. Using the Fourier analysis machinery we know that:

$$\Pr[\text{test accepts}] = \frac{1}{2} + \frac{1}{2} E_{xyz}[A(x)B(y)B((x \circ \pi)yz)].$$

Define $\pi_2(\beta) \triangleq \{i \in \{1,\ldots,k\} | \text{an odd number of } j \in \beta \text{ satisfy } \pi(j) = i\}$ so that $\prod_{j\in\gamma} x[\pi(j)] = \prod_{i\in\pi_2(\gamma)} x_i$. We can then calculate:

$$\begin{aligned}
& E_{x,y,z}[A(x)B(y)B((x \circ \pi)yz)] \\
= \; & E_{x,y,z}\left[\left(\sum_\alpha \hat{A}_\alpha L_\alpha(x)\right)\left(\sum_\beta \hat{B}_\beta L_\beta(y)\right)\left(\sum_\gamma \hat{B}_\gamma L_\gamma(x \circ \pi) L_\gamma(y) L_\gamma(z)\right)\right] \\
= \; & \sum_{\alpha,\beta,\gamma} \hat{A}_\alpha \hat{B}_\beta \hat{B}_\gamma \overbrace{E_x[L_\alpha(x) L_\gamma(x \circ \pi)]}^{L_\alpha \cdot L_{\pi_2(\gamma)}} \overbrace{E_y[L_\beta(y) L_\gamma(y)]}^{L_\beta \cdot L_\gamma} \overbrace{E_z[L_\gamma(z)]}^{(1-2\epsilon)^{|\gamma|}} \\
= \; & \sum_\beta \hat{A}_{\pi_2(\beta)} \hat{B}_\beta^2 (1-2\epsilon)^{|\beta|}
\end{aligned}$$

Now suppose that Pr[test accepts] $\geq \frac{1}{2} + \delta$ so that:

$$\begin{aligned}
2\delta &\leq \sum_{\beta} \hat{A}_{\pi_2(\beta)} \hat{B}_{\beta}^2 (1-2\epsilon)^{|\beta|} \\
&\leq \sum_{\beta} |\hat{A}_{\pi_2(\beta)}| \hat{B}_{\beta}^2 (1-2\epsilon)^{|\beta|} \\
&= \sum_{\beta : |\hat{A}_{\pi_2(\beta)}| \geq \delta} |\hat{A}_{\pi_2(\beta)}| \hat{B}_{\beta}^2 (1-2\epsilon)^{|\beta|} + \overbrace{\sum_{\beta : |\hat{A}_{\pi_2(\beta)}| < \delta} |\hat{A}_{\pi_2(\beta)}| \hat{B}_{\beta}^2 (1-2\epsilon)^{|\beta|}}^{<\delta}
\end{aligned}$$

Multiplying both sides by δ yields:

$$\begin{aligned}
\delta^2 &\leq \sum_{\beta : |\hat{A}_{\pi_2(\beta)}| \geq \delta} \hat{A}_{\pi_2(\beta)}^2 \hat{B}_{\beta}^2 (1-2\epsilon)^{|\beta|} \\
&\leq \sum_{\beta} \hat{A}_{\pi_2(\beta)}^2 \hat{B}_{\beta}^2 (1-2\epsilon)^{|\beta|}
\end{aligned}$$

Now consider the following decoding algorithm: $D(A)$ picks $\alpha \subseteq \{1, \ldots, k\}$ with probability $\hat{A}_{\alpha}^2$ and $a \in \alpha$ at random. Similarly $D(B)$ picks $\beta \subseteq \{1, \ldots, t\}$ with probability $\hat{B}_{\beta}^2$ and $b \in \beta$ at random. Using this strategy and noting that $\exists c : (1-2\epsilon)^k \leq \frac{c}{k}, c = O(\frac{1}{\epsilon})$ we find:

$$\begin{aligned}
\Pr[D(A) = \pi(D(B))] &\geq \sum_{\beta} \hat{A}_{\pi_2(\beta)}^2 \hat{B}_{\beta}^2 \frac{1}{|\beta|} \\
&\geq \frac{1}{c} \sum_{\beta} \hat{A}_{\pi_2(\beta)}^2 \hat{B}_{\beta}^2 (1-2\epsilon)^{|\beta|} \\
&\geq \frac{\delta^2}{c} = \Theta(\epsilon\delta^2)
\end{aligned}$$

This probability does not depend on t or k, so that we can adjust the soundness parameter of Raz's verifier and not have a circular relationship of the variables.