Analysis of Boolean Functions **(CMU 18-859S, Spring 2007)**

Lecture 18: Hardness Amplification continued

Mar. 22, 2007

Lecturer: Ryan O'Donnell *Scribe: Moritz Hardt*

1 Connection to random restrictions and Expected Bias

Assume $g : \{-1,1\}^m \to \{-1,1\}$ is $1-\epsilon$-hard for size s. Further assume g is balanced, i.e., $\mathbf{E}[g] = 0$. Although one can get rid of the last assumption, we will use it for the sake of simplicity.

Let H be a γ-hard-core set for g against size $s' = \Omega(\gamma^2 \log \frac{1}{\gamma\epsilon})$ of cardinality $\epsilon 2^m$ s.t. g is balanced on H. Notice, g is also balanced on $\bar{H}$.

Let $f : \{-1,1\}^n \to \{-1,1\}$. Suppose a circuit C of size at most s'' tries to compute $f(g(x^1), \ldots, g(x^n)$ on uniformly drawn inputs $x^1, \ldots, x^n$.

$$\text{Is } x^1, \ldots, x^n \in H?$$

- Conditioned on $x^i \in H$, $g(x^i)$ is computationally indistinguishable (up to γ) from a random bit to C.
- Conditioned on $x^i \notin H$, $g(x^i)$ is still a uniformly distributed random bit, but C might exactly know $g(x^i)$. (Remember, the example from last lecture.)

The point is, think of $(g(x^i), \ldots, g(x^n))$ as a random restriction $(y, \bar{I})$ with $*$-probability ϵ. The best thing C could do is look at $f_{y\to\bar{I}}$ and output the more common value.

Definition 1.1 *The* expected bias of f at ϵ *is*

$$\mathrm{ExpBias}_\epsilon(f) = \mathbf{E}[|\widehat{f_{y\to\bar{I}}}(\emptyset)|$$

where the expectation is over random restrictions $(y, \bar{I})$ with $*$-probability ϵ.

Theorem 1.2 *Let $g : \{-1,1\}^m \to \{-1,1\}$ which is $1-\epsilon$-hard for size s, assume g is balanced and let $f : \{-1,1\}^n \to \{-1,1\}$. Let $\gamma > 0$. Then, $f \otimes g$ is $\frac{1}{2} + \frac{1}{2}\mathrm{ExpBias}_\epsilon(f) + \gamma$-hard for circuits of size*

$$s'' = \Omega\left(\frac{\gamma^2 \log \frac{1}{\gamma\epsilon}}{n}\right) \cdot s.$$

Proof:[Sketch] We will not prove this result, although the proof is not too difficult. It follows from a hybrid argument. This is why we lose the factor of n in s''. □

Since $\mathrm{ExpBias}_\epsilon(\chi_{[n]}) = (1-\epsilon)^n$, we get Yao's XOR Lemma as a corollary at least in the case of balanced g.

Corollary 1.3 (Yao's XOR Lemma for balanced functions)

Remark 1.4 *The theorem is essentially tight.*

Proposition 1.5

$$\mathbb{S}_{1-\epsilon}(f) \leq \mathrm{ExpBias}_\epsilon(f) \leq \sqrt{\mathbb{S}_{1-\epsilon}(f)}$$

Proof:

$$\mathop{\mathbf{E}}_{y,\bar{I}}[\widehat{f_{y\to\bar{I}}}(\emptyset)^2] \leq \mathop{\mathbf{E}}_{y,\bar{I}}[|\widehat{f_{y\to\bar{I}}}(\emptyset)|] \leq \sqrt{\mathop{\mathbf{E}}_{y,\bar{I}}[\widehat{f_{y\to\bar{I}}}(\emptyset)^2]}$$

The following proposition concludes our proof. □

Proposition 1.6

$$\mathop{\mathbf{E}}_{y,\bar{I}}[\widehat{f_{y\to\bar{I}}}(\emptyset)^2] = \mathbb{S}_{1-\epsilon}(f)$$

Proof:

$$\begin{aligned}
\mathop{\mathbf{E}}_{y,\bar{I}}[\widehat{f_{y\to\bar{I}}}(\emptyset)^2] &= \mathop{\mathbf{E}}_{\bar{I}}[\mathop{\mathbf{E}}_{y}[F_\emptyset(y)^2]] \\
&= \mathop{\mathbf{E}}_{\bar{I}}[\sum_{S\subseteq\bar{I}} \widehat{F_\emptyset}(S)^2] \\
&= \mathop{\mathbf{E}}_{\bar{I}}[\sum_{S\subseteq\bar{I}} \hat{f}(S)^2] \\
&= \sum_S \hat{f}(S)^2\mathbf{Pr}[S\subseteq\bar{I}] \\
&= \sum_S (1-\epsilon)^{|S|}\hat{f}(S)^2
\end{aligned}$$

□

2 Very noise sensitive *monotone* functions

Our goal is now clear. We want to find very *noise sensitive* monotone functions.

Definition 2.1 *The* noise sensitivity of f at $\epsilon \in [0, 1/2]$, *denoted* $\mathrm{NS}_\epsilon(f)$ *is*

$$\mathbf{Pr}_{x,y=N_\epsilon(x)}[f(x) \neq f(y)]$$

where $y = N_\epsilon(x)$ means that y is formed by flipping each bit of x independently with probability ϵ.

Proposition 2.2

$$\mathrm{NS}_\epsilon(f) = \frac{1}{2} - \frac{1}{2}\mathbb{S}_{1-2\epsilon}(f) = \frac{1}{2} - \frac{1}{2}\sum_S (1-2\epsilon)^{|S|}\hat{f}(S)^2$$

Proof:

$$\mathbb{S}_{1-2\epsilon}(f) = \mathop{\mathbf{E}}_{x,y\sim_{1-2\epsilon}x}[f(x)f(y)]$$

That is, x is drawn uniformly at random and y is a $(1-2\epsilon)$-correlated copy of x. But, that is equivalent to choosing $y = N_\epsilon(x)$. So,

$$\mathop{\mathbf{E}}_{x,y\sim_{1-2\epsilon}x}[f(x)f(y)] = \mathop{\mathbf{E}}_{x,y=N_\epsilon(x)}[1 - 2\mathbb{P}[f(x) \neq f(y)]] = 1 - 2\mathrm{NS}_\epsilon(f)$$

□

Theorem 2.3 *If $f : \{-1,1\}^n \to \{-1,1\}$ is monotone (in NP) and $\mathrm{NS}_{n^{-\alpha}}(f) \geq \frac{1}{2} - n^{-\beta}$, then "NP is $1 - \frac{1}{\mathrm{poly}(n)}$-hard for poly-size circuits" implies "NP is $\approx \frac{1}{2} + n^{-\beta/2}$-hard for poly-size circuits".*

We can picture the following.

Proposition 2.4 *$\mathrm{NS}_\epsilon(f)$ is a concave, increasing function of ϵ. It is 0 when $\epsilon = 0$ and it is $\frac{1}{2}$ at $\epsilon = \frac{1}{2}$.*

Proof: Since $\mathrm{NS}_\epsilon(f)$ is a concave function of ϵ, 0 at 0, we have $\mathrm{NS}_\epsilon(f) \leq \epsilon \mathrm{NS}'_0(f)$. □

Therefore, $\mathrm{NS}_\epsilon(f) \leq O(\epsilon\sqrt{n})$ if f is monotone and $\mathrm{NS}_\epsilon(f) < \frac{1}{4}$, if $\epsilon < \Omega\left(\frac{1}{\sqrt{n}}\right)$.

3 Recursive Majority

Theorem 3.1

$$\mathrm{NS}_\epsilon(\mathrm{Maj}_n) \leq O(\sqrt{\epsilon})$$

Although, this theorem makes Majority a seemingly bad candidate for our purpose, we still try to recursively construct some good function starting with Majority.

$$p(\epsilon) := \mathrm{NS}_\epsilon(\mathrm{Maj}_3) = \frac{3}{2}\epsilon - \frac{3}{2}\epsilon^2 + \epsilon^3 =$$

Question: What is NS_ϵ of $\mathrm{Maj}_3(\mathrm{Maj}_3(\dots), \mathrm{Maj}_3(\dots), \mathrm{Maj}_3(\dots))$?

Observation 3.2 *If f is balanced, then*

$$\mathrm{NS}_\epsilon(f' \otimes f) = \mathrm{NS}_{\mathrm{NS}_\epsilon(f)}(f').$$

In particular,

- for small ϵ, $p(p(\epsilon)) \approx \left(\frac{3}{2}\right)^2 \epsilon$, $p(p(p(\epsilon))) \approx \left(\frac{3}{2}\right)^3 \epsilon$.
- for small δ, $p(p(1-\delta)) \approx \frac{1}{2} - \left(\frac{3}{4}\right)^2 \delta$.

So, define

$$\mathrm{Maj}_3^{(k)} = \underbrace{\mathrm{Maj}_3 \otimes \cdots \otimes \mathrm{Maj}_3}_{k \text{ times}}.$$

We get $\mathrm{NS}_\epsilon(\mathrm{Maj}_3^{(k)}) = p^{(k)}(\epsilon)$. The input length is 3^k.

Fact 3.3 *If depth* $k \geq (1 + o(1))(\log_{\frac{3}{2}}(\frac{1}{\epsilon}) + \log_{\frac{4}{3}}(\frac{1}{\delta}))$*, then* $\mathrm{NS}_\epsilon(\mathrm{Maj}_3^{(k)}) \geq \frac{1}{/}2 - \delta$.

Write $n = 3^k$ for the input length. We get $\mathrm{NS}_eps(\mathrm{Maj}_3^{(k)}) \geq \frac{1}{2} - \delta$, so long as

$$n \gtrapprox 3^{\log_{3/2}(1/\epsilon) + \log_{4/3}(1/\delta)} = \left(\frac{1}{\epsilon}\right)^{\log_{3/2} 3} \left(\frac{1}{\delta}\right)^{\log_{4/3} 3} \approx \left(\frac{1}{\epsilon}\right)^{2.71} \left(\frac{1}{\delta}\right)^{3.82}$$

So, if $\epsilon \geq \frac{1}{n^{1/\delta}}$, $\delta \leq \frac{1}{n^{1/8}}$, this holds. Finally, you get a monotone function f, computable in polynomial time, with $\mathrm{NS}_{n^{-1/\epsilon}}(f) \geq \frac{1}{2} - n^{-1/8}$.

Corollary 3.4 *If* $\exists L \in \mathrm{NP}$ *(balanced) which is* $1 - \frac{1}{\mathrm{poly}(n)}$*-hard for size* $s = \mathrm{poly}(n)$*, then* $\exists L \in \mathrm{NP}$ *which is* $\frac{1}{2} + \frac{1}{n^{\Omega(1)}}$*-hard (infinitely often) for size poly*(n)*.*

Remark 3.5 *Improvement (Healy-Vadhan-Viola): If there exists a balanced* $L \in \mathrm{NP}$*,* $1 - \frac{1}{\mathrm{poly}(n)}$*-hard for size* $2^{\Omega(n)}$*, then* $\exists L \in \mathrm{NP}$ *which is* $\frac{1}{2} + 2^{-\Omega(\sqrt{n})}$ *hard for size* $2^{\Omega(n)}$*.*