Harmonic Analysis of Boolean Functions, and applications in CS

Lecture -8

April 28, 2008

Lecturer: Guy Kindler Scribe by: Shira Kritchman Updated: May 11, 2008

Last lecture we finished proving Kalai's theorem, and started to talk about codes and code testing. This lecture we continue talking about codes. More specifically, we define the *Hadamard code* and examine the *linearity test* for it. Finally, we define the notion of *testing constraints*, and suggest a test for equality on the Hadamard code.

1 The long code - Revision

In this section we mostly revise material from the previous lecture. We define the notions of *code family* and *code testing*, and we learn about the *long code*.

Definition 1 (Code and code family) *A code is a set of strings over some alphabet Σ (in this course $\Sigma = \{\pm 1\}$). A code family is a set of codes with unbounded size (where size refers to the number of words in a code).*

We will refer to an element of a code as a *code word*. To a general string we will sometimes refer as just *word*.

Note that usually we speak of a family of codes without explicitly indexing each member. That is, we discuss some code C which implicitly depends on n, and should be properly denoted $C^{(n)}$.

Definition 2 (Code tester) *A q-query code tester for a code C, with completeness parameter c and soundness parameter s, is a random procedure T, which reads q locations from a string f and either accepts or rejects the string, and satisfies:*
(completeness) $f \in C \Rightarrow \Pr[T^f \text{ accepts}] > c$
(soundness) $\Pr[T^f \text{ accepts}] > s \Rightarrow$ *"There exists a reasonable decoding for f".*

Note that 'reasonable decoding' wasn't well defined. It can mean that the word is very close to some unique code word, or that it has high correlation with some code word and with not too many code words, or even more generally, that there is a not too long list of code words which are somehow related to f.

In theoretical computer science, code testers are important as a tool often used in the analysis of hardness of approximation.

The first code that we define is the *long code*. The strings in the set $\{\pm 1\}^{2^n}$ can be understood as truth tables describing all the Boolean functions on n variables. The long code is the set of all $f \in \{\pm 1\}^{2^n}$ that describe dictatorships:

Definition 3 (Long code)

$$C = \{\chi_i\}_{i \in [n]} \subseteq \{\pm 1\}^{2^n}$$

We now describe a test for a code similar to the long code - the set of all dictatorships and minus dictatorships,

$$\widetilde{C} = \{\chi_i\}_{i\in[n]} \cup \{-\chi_i\}_{i\in[n]}$$

Before we describe the test, we should recall some more definitions.

Definition 4 (The Not All Equal function) *NAE is the boolean function on* $\{\pm 1\}^3$ *which equals* -1 *on* (x, y, z) *iff* $x = y = z$.

Definition 5 (The set ψ) *ψ is the set of all legal votes,*

$$\psi = \left\{ (\mathbf{x}, \mathbf{y}, \mathbf{z}) \in (\{\pm 1\}^n)^3 \middle| \forall i \ NAE(x_i, y_i, z_i) = 1 \right\}$$

We are now ready to define the NAE test for the code $\widetilde{C}$.

Definition 6 (NAE test) *Given* $f \in \{\pm 1\}$*, pick uniformly at random* $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \psi$*, and accept iff* $NAE(f(\mathbf{x}), f(\mathbf{y}), f(\mathbf{z})) = 1$.

Denote the test by T_{NAE}. Clearly, the NAE test has completeness $c = 1$:

(Completeness) $f \in \widetilde{C} \Rightarrow \Pr[T^f_{NAE} \text{ accepts}] = 1$.

From Kalai's theorem (which we phrased in lecture #7 and proved in lectures #7 and #8), we know that if $\Pr[T^f_{NAE} \text{ accepts}] > 1 - \varepsilon$ then f is $k\varepsilon$-close to a code word. From this it follows easily that:

(Soundness) $\forall \delta$ there exists $s(\delta)$ such that if $\Pr[T^f_{NAE} \text{ accepts}] > s(\delta)$ then $\exists i \in [n]$ for which $\left|\hat{f}(i)\right| > 1 - \delta$.

Note that if $\hat{f}(i) > 1 - \delta$, then f is close to the dictatorship function χ_i:

$$||f - \chi_i||_2^2 = \sum_{S \neq \{i\}} \hat{f}(S)^2 + (\hat{f}(i) - 1)^2 = 1 - \hat{f}(i)^2 + (\hat{f}(i) - 1)^2 = 2 - 2\hat{f}(i) \leq 2\delta$$

and thus the Hamming distance between f and χ_i is small:

$$\Pr_{\mathbf{x}}[f(\mathbf{x}) \neq \chi_i(\mathbf{x})] = \frac{||f - x_i||_2^2}{4} \leq \frac{\delta}{2}$$

(similarly, if $-\hat{f}(i) > 1 - \delta$, then f is close to $-\chi_i$).

2 The Hadamard code

In this section we define the *Hadamard code* and examine the *linearity test* for it.

The Hadamard code is the set of all Boolean functions on n variables which are characters:

Definition 7 (Hadamard code)

$$C = \{\chi_s\}_{s \subseteq n} \subseteq \{\pm 1\}^{2^n}$$

We suggest testing it using the 3-query linearity test:

Definition 8 (Linearity test) *Given* $f : \{\pm 1\}^n \to \{\pm 1\}$*, pick uniformly at random* $\mathbf{x}, \mathbf{y} \in \{\pm 1\}^n$*, and accept iff* $f(\mathbf{x})f(\mathbf{y}) = f(\mathbf{xy})$*.*

We now discuss the properties of the linearity test. We denote it by T_{lin}. Clearly, the linearity test has completeness $c = 1$:

(Completeness) f is a character $\Rightarrow \Pr[T_{lin}^f \text{ accepts}] = 1$.

As usual, we have to work harder to show soundness.

(Soundness) Here we wish to say something about f given that the probability of accepting f is high. We consider two cases: $\Pr[T_{lin}^f \text{ accepts}] > 1 - \delta$, where δ is small, and $\Pr[T_{lin}^f \text{ accepts}] \geq \frac{1}{2} + \delta$. We present two soundness lemmas dealing with the two cases.

Lemma 9 (Soundness lemma I for linearity test) *If* $\Pr[T_{lin}^f \text{ accepts}] > 1 - \delta$ *then* $\exists S \subseteq [n]$ *such that* $\hat{f}_S > 1 - \delta$*.*

Proof We wish to represent the acceptance probability as a function of the Fourier coefficients. Note that the test accepts iff $f(\mathbf{x})f(\mathbf{y})f(\mathbf{xy}) = 1$. Thus,

$$\mathbb{E}[f(\mathbf{x})f(\mathbf{y})f(\mathbf{xy})] = \Pr[T_{lin}^f \text{ accepts}] - \Pr[T_{lin}^f \text{ rejects}] = 2\Pr[T_{lin}^f \text{ accepts}] - 1$$

Repeating what we have seen in the second exercise,

$$\begin{aligned}
\mathbb{E}_{\mathbf{x},\mathbf{y}}[f(\mathbf{x})f(\mathbf{y})f(\mathbf{xy})] &= \mathbb{E}_{\mathbf{x},\mathbf{y}}\left\{\left[\sum_S \hat{f}(S)\chi_S(\mathbf{x})\right]\left[\sum_T \hat{f}(T)\chi_T(\mathbf{y})\right]\left[\sum_R \hat{f}(R)\chi_R(\mathbf{xy})\right]\right\} \\
&= \sum_{S,T,R} \hat{f}(S)\hat{f}(T)\hat{f}(R)\mathbb{E}[\chi_S(\mathbf{x})\chi_T(\mathbf{y})\chi_R(\mathbf{xy})] \\
&= \sum_S \hat{f}(S)^3
\end{aligned}$$

Thus, if $\Pr[T_{lin}^f \text{ accepts}] > 1 - \delta$ then $\sum_S \hat{f}(S)^3 = 2Pr[T_{lin}^f \text{ accepts}] - 1 > 1 - 2\delta$. Since $\sum_S \hat{f}(S)^3 \leq \max_S\{\hat{f}(S)\} \cdot \sum_S \hat{f}(S)^2 = \max_S\{\hat{f}(S)\}$, we get that if the acceptance probability is higher than $1 - \delta$, then there exists $S \subseteq [n]$ such that $\hat{f}(S) > 1 - 2\delta$.

Note that if δ is small (smaller than $\left(1 - 1/\sqrt{2}\right)/2 = 0.146$), then S is unique. In this case χ_S provides a unique decoding for f. ■

Lemma 10 (Soundness lemma II for linearity test) *If* $\Pr[T_{lin}^f \text{ accepts}] > \frac{1}{2} + \delta$ *then* $\exists S \subseteq [n]$ *such that* $\hat{f}(s) > 2\delta$*.*

Proof This follows immediately from the previous lemma: $\Pr[T^f_{lin} \text{ accepts}] \geq \frac{1}{2} + \delta = 1 - \left(\frac{1}{2} - \delta\right)$ and hence $\max_S\{\hat{f}(S)\} \geq 1 - 2\left(\frac{1}{2} - \delta\right) = 2\delta$. ■

It seems that the second soundness lemma doesn't say much - when δ is small, there could be many Fourier coefficients greater than 2δ. But since $\sum_S \hat{f}(S)^2 = 1$, there could be no more than $\frac{1}{4\delta^2}$ such coefficients. To each χ_S such that $\hat{f}(S) \geq 2\delta$ we call a *reasonable decoding*. The list of reasonable decodings provides a *list decoding* - a non-empty yet not too long list of reasonable decodings.

Remark What if we know that $\Pr[T^f_{lin} \text{ accepts}] \leq \frac{1}{2} - \delta$? Then $\Pr[T^{-f}_{lin} \text{ accepts}] = 1 - \Pr[T^f_{lin} \text{ accepts}] \geq \frac{1}{2} + \delta$, hence $\exists S \subseteq [n]$ such that $-\hat{f}(S) > 2\delta$.

In exercise #4 we will see an interesting code which has a 2-query test.

3 Testing constraints

In this section we define the notions of *constraint* and *constraint tester*, and suggest a test for equality on the Hadamard code.

Definition 11 (Constraint) *A constraint between two codes C_1 and C_2 is a relation R between them as sets. That is, $f \in C_1$ and $g \in C_2$ satisfy the constraint if fRg.*

A simple example for a constraint is $C_1 = C_2 = C$ the Hadamard code, and $f, g \in C$ satisfy the constraint if $f = g$.

Intuitively, a constraint tester is a procedure that receives as input two strings f and g, and checks whether (1) $f \in C_1$, (2) $g \in C_2$ and (3) fRg. Therefore, a constraint tester is similar to a Kind(l)er egg. To define things more rigorously, we need a few more definitions.

Definition 12 (Constraint collection) *A constraint collection on two codes C_1 and C_2 is a set of relations over C_1 and C_2,*

$$\mathcal{R} = \{R_\lambda\}_{\lambda \in \Lambda}$$

(again, we have an implicit dependence on n. That is, C_i is $C_i^{(n)}$, and $\mathcal{R}$ is $\mathcal{R}^{(n)}$).

Definition 13 (Decoding scheme) *A decoding scheme for a code C is a map D such that for any string f, $D(f)$ is a distribution over $C \cup \{\perp\}$.*

We are now ready to define the notion of a constraint tester.

Definition 14 (Constraint tester) *A q-query constraint tester for codes C_1 and C_2 and a constraint family $\mathcal{R} = \{R_\lambda\}_{\lambda \in \Lambda}$, with completeness parameter c and soundness parameter s, is a random procedure T, which reads $\lambda \in \Lambda$ and q locations from strings f and g, and either accepts or rejects, and satisfies:*
(completeness) $f \in C_1$, $g \in C_2$ and $fR_\lambda g \Rightarrow \Pr[T^{f,g}(\lambda) \text{accepts}] > c$

(soundness) T has soundness $s = s(\delta)$ for δ-satisfaction rate, if there exist decoding schemes D_1, D_2 for C_1, C_2 such that

$$\Pr_{\text{randomness of } T}[T^{f,g}(\lambda) \text{ accepts}] \geq s \Rightarrow \Pr_{f' \sim D_1(f), g' \sim D_2(g)}[f' R g'] \geq \delta.$$

We now construct a 3-query test for identity between Hadamard words. That is, given f and g, we want to test if they are characters, and if $f = g$, reading only three bits.

Definition 15 (Testing linearity between Hadamard words) *Given f and g pick uniformly at random $\mathbf{x}, \mathbf{y} \in \{\pm 1\}^n$, and accept iff $f(\mathbf{x})f(\mathbf{y}) = g(\mathbf{xy})$.*

We now discuss the properties of this test. We denote it by T_{eq}. Clearly, the test has completeness parameter $c = 1$:

(Completeness) $f, g \in C$ and $f = g \Rightarrow \Pr[T_{eq}^{f,g} \text{ accepts}] = 1$.

As for soundness, we show the following:

(Soundness) If $\Pr[T_{eq}^{f,g} \text{ accepts}] > 1 - \delta$ then there exist (reasonable) decoding schemes D_1 and D_2 such that (when δ is small enough) $\Pr_{f' \sim D_1, g' \sim D_2}[f' = g']$ is 'large'.

Proof Suppose $\Pr[T_{eq}^{f,g} \text{ accepts}] > 1 - \delta$, then

$$\sum_S \hat{f}(S)^2 \hat{g}(S) = \mathbb{E}[f(\mathbf{x})f(\mathbf{y})g(\mathbf{xy})] = 2\Pr[T_{eq}^{f,g} \text{ accepts}] - 1 > 1 - 2\delta$$

This is a weighted mean of the coefficients of g, and hence there exists some $S \subseteq [n]$ such that $\hat{g}(S) > 1 - 2\delta$. If δ is small enough, then S is unique and we denote it S^*. Let D_2 be the decoding scheme which returns χ_{S^*}. As for f, we suggest two different decoding schemes:

(1) Note that by the Cauchy-Schwarz inequality, $\sum_S \left|\hat{f}(s)\hat{g}(S)\right| \leq ||f^2|| \cdot ||g^2|| = 1$, and therefore

$$\max_S\{\hat{f}(S)\} \geq \max_S\{\hat{f}(S)\} \sum_S \left|\hat{f}(S)\hat{g}(S)\right| \geq \sum_S \hat{f}(S)^2 \hat{g}(S) > 1 - 2\delta$$

So again, if δ is small enough, $\exists! S^{**}$ such that $\hat{f}(S^{**}) > 1 - 2\delta$. Hence, let D_1 be the decoding scheme which returns $\chi_{S^{**}}$. If δ is small enough then we must have $S^* = S^{**}$ (because otherwise $\sum_S \hat{f}(S)^2 \hat{g}(S)$ is too small). Hence we get that $\Pr_{f' \sim D_1(f), g' \sim D_2(g)}[f' R g'] = 1$.

(2) $\sum_S \hat{f}(S)^2 = 1$, hence it defines a probability distribution over $2^{[n]}$. Let D_1 be the decoding scheme returning χ_S with probability $\hat{f}(S)^2$. Then

$$\Pr_{f' \sim D_1, g' \sim D_2}[f' = g'] = \Pr[f' = \chi_{S^*}] = \hat{f}(S^*)^2$$

This is large since $\hat{g}(S^*) > 1 - 2\delta$, and therefore $\max_{S \neq S^*}\{\hat{g}(S)\} < \sqrt{1-(1-2\delta)^2} = 2\sqrt{\delta - \delta^2}$, and therefore

$$\hat{f}(S^*)^2\hat{g}(S^*) > 1 - 2\delta - \sum_{S \neq S^*} \hat{f}(S)^2\hat{g}(S) \geq 1 - 2\delta - \max_{S \neq S^*}\{\hat{g}(S)\} > 1 - 2\delta - 2\sqrt{\delta - \delta^2}.$$

This gives

$$\Pr_{f' \sim D_1, g' \sim D_2}[f' = g'] = \hat{f}(S^*)^2 > \frac{1 - \delta - 2\sqrt{\delta - \delta^2}}{\hat{g}(S^*)} \geq 1 - 2\delta - 2\sqrt{\delta - \delta^2}.$$

■