Define $h(x) = T_{1-2\delta}(f*f)(\pi^{-1}(x))$ for $x \in \{0,1\}^\ell$.

$$\dots = E[g(x)h(x)] = \sum_{T\subseteq[\ell]} \hat{g}(T)\cdot\hat{h}(T) \underset{\text{Homework 1, Q2i}}{=} \sum_{T\subseteq[\ell]} \hat{g}(T)\cdot \sum_{\substack{S\subseteq[k]\\ \pi_2(S)=T}} \widehat{T_{1-2\delta}(f*f)}(S) =$$

$$= \sum_{S\subseteq[k]} \hat{g}(\pi_2(S))\cdot \widehat{T_{1-2\delta}(f*f)}(S) = \sum_{S\subseteq[k]} \hat{g}(\pi_2(S))\cdot \hat{f}(S)^2\cdot(1-2\delta)^{|S|}. \quad \blacksquare$$

<u>Cor</u>: If $f = \chi_{\{i\}}$ and $g = \chi_{\{j\}}$ with $\pi(i)=j$, then $Pr[\text{accepts}] = 1-\delta$.

<u>Proof</u>: Either from lemma or directly.

<u>Def</u>: For a function $f:\{0,1\}^k \to \{-1,1\}$ with $\hat{f}(\phi)=0$, define Q_f to be the distribution on $i\in[k]$ obtained by choosing $S\subseteq[k]$ with prob. $\hat{f}(S)^2$ and then choosing $i\in S$ uniformly. In other words, $Pr[Q_f = i] = \sum_{S\ni i} \frac{\hat{f}(S)^2}{|S|}$.

<u>Cor</u>: Assume $Pr[\text{test accepts } f,g,\pi] \geq \frac{1}{2}+\varepsilon$ and moreover $\hat{f}(\phi)=\hat{g}(\phi)=0$.

Then, $\underset{\substack{i\sim Q_f\\ j\sim Q_g}}{Pr}[\pi(i)=j] \geq \delta\cdot\varepsilon^3$.

<u>Proof</u>: By the assumption, $2\varepsilon \leq \sum_{S\subseteq[k]} \hat{f}(S)^2\cdot\hat{g}(\pi_2(S))\cdot(1-2\delta)^{|S|}$

Let $\mathcal{F} = \{S\subseteq[k] : \hat{g}(\pi_2(S))\cdot(1-2\delta)^{|S|} > \varepsilon\}$. Then, $\sum_{S\notin\mathcal{F}} \hat{f}(S)^2\cdot\hat{g}(\pi_2(S))\cdot(1-2\delta)^{|S|} \leq \varepsilon\cdot\sum\hat{f}(S)^2 = \varepsilon$.

Therefore, $\varepsilon \leq \sum_{S\in\mathcal{F}} \hat{f}(S)^2\cdot\underbrace{\hat{g}(\pi_2(S))\cdot(1-2\delta)^{|S|}}_{\leq 1} \leq \sum_{S\in\mathcal{F}} \hat{f}(S)^2$.

Now, $\underset{\substack{i\sim Q_f\\ j\sim Q_g}}{Pr}[\pi(i)=j] \geq \sum_S \hat{f}(S)^2\cdot\hat{g}(\pi_2(S))^2\cdot\frac{1}{|S|}$, where the "$\frac{1}{|S|}$" appears because for each $j\in\pi_2(S)$ there is at least one $i\in S$ s.t. $\pi(i)=j$.

Using the ineq. $i\cdot(1-\delta)^i \leq \frac{1}{\delta}$, $\hat{g}(\pi_2(S))^2\cdot\frac{1}{|S|} \geq \delta\cdot\hat{g}(\pi_2(S))^2\cdot(1-\delta)^{|S|}$, which is at least $\delta\cdot\varepsilon^2$ for all $S\in\mathcal{F}$. Hence, $Pr[\dots] \geq \delta\cdot\varepsilon^2\sum_{S\in\mathcal{F}}\hat{f}(S)^2 \geq \delta\cdot\varepsilon^3$. $\blacksquare$

<u>Thm</u>: $\forall\eta>0$ it is NP-hard to tell whether a given MAX3LIN2 instance has value $\geq 1-\eta$ or $\leq \frac{1}{2}+\eta$.

28.2.2008 <u>Proof</u>: Let $\lambda = \frac{\eta^5}{16}$. By the PCP + ParRep theorems, there exists a $k=k(\lambda)$, $\ell=\ell(\lambda)$ s.t. the following is NP-hard: given a label cover instance that is bipartite with assignments $[k]$ on the left side and $[\ell]$ on the right side, and a projection constraint $\pi_{u,v}:[k]\to[\ell]$ associated to each edge (u,v), decide whether value $=1$ or value $\leq\lambda$. We will show a reduction from this to MAX3LIN2.

The reduction replaces each left variable u with 2^{k-1} bits representing an odd function $f_u:\{0,1\}^k\to\{-1,1\}$ (i.e. $\forall x.\ f(x) = -f(x+(1,\dots,1))$). The bits are the evaluations of f_u on all inputs starting with 0. We can deduce the value of f_u on inputs starting with 1 because f_u is odd (this uses the fact that negation is allowed in MAX3LIN2). This trick is called "folding". We similarly replace each v on the right by $2^{\ell-1}$ bits representing g_v.

The MAX3LIN2 equations are given by the following tester: choose a constraint (u,v) uniformly and apply the $\text{Hastad}_{2,\eta}$ test to $f_u, g_v, \pi_{u,v}$.

Completeness: assume the value of the label cover is 1. Consider the following assignment to the MAX3LIN2. Set each f_u to be $\chi_{L(u)}$ and similarly for g. Since all constraints (u,v) are such that $\pi(L(u)) = L(v)$ Cor. 1 shows that our tester accepts w.p. $\geq 1-\eta$.

Soundness: assume that the tester accepts w.p. $> \frac{1}{2}+\eta$. By an averaging argument for $\frac{\eta}{2}$ of the ~~test~~ constraints the $\text{Hastad}_{2,\eta}$ test accepts w.p. $\geq \frac{1}{2}+\frac{\eta}{2}$. Consider the assignment L that for each u chooses a value in $[k]$ according to Q_{f_u}. Similarly, L assigns for each v a value from $[\ell]$ according to Q_{g_v}. Then, by Cor. (using the fact that f_u and g_v are odd) L satisfies each such constraint w.p. $> \eta\cdot\left(\frac{\eta}{2}\right)^3 = \frac{\eta^4}{8}$. So overall, L satisfies in expectation $> \frac{\eta^5}{16} = \lambda$ of the constraints.

Learning

Learning functions close to Parities (χ)

Prop: Given (query) access to a function $f:\{0,1\}^n \to \{-1,1\}$ that is $(\frac{1}{4}-\varepsilon)$-close to a parity χ_S, we can recover S with confidence $1-\delta$ using $O(n\log\frac{n}{\delta}/\varepsilon^2)$ queries.

Proof: Using local decoding we can get a guess for $\chi_S(e_i)$ that is correct w.p. $\geq \frac{1}{2}+2\varepsilon$ using 2 queries. By repeating this $O(\log\frac{n}{\delta}/\varepsilon^2)$ times, we can get an estimate that is correct w.p. $\geq 1-\frac{\delta}{n}$. If we repeat this for $i=1,\dots,n$, we get a guess for S that is correct w.p. $1-\delta$. ∎

For f that is farther than $\frac{1}{4}$ from parities, we can no longer find the closest parity because it might not be unique. $\chi_S \bullet \xleftrightarrow{1/2} \bullet \chi_T$

The Goldreich-Levin (1989) Algorithm

Our goal is to find all S s.t. $|\hat{f}(S)| \geq \gamma$ for some small γ (this is a (local) list decoding of Hadamard).

Claim: For $f:\{0,1\}^n \to \{-1,1\}$ $\#\{S : |\hat{f}(S)| \geq \gamma\} \leq \frac{1}{\gamma^2}$.

Thm [GL89]: Given (query) access to $f:\{0,1\}^n \to [-1,1]$ and $\gamma, \delta > 0$ there is a $\text{poly}(n, \frac{1}{\gamma}, \log\frac{1}{\delta})$-time algorithm that w.p. $\geq 1-\delta$ outputs a list $F = \{S_1, S_2, \dots, S_m\}$ s.t. every S with $|\hat{f}(S)| \geq \gamma$ is in F and also any S with $|\hat{f}(S)| < \frac{\gamma}{2}$ is not in F.

For this proof we sometimes think of S as an n-bit string.

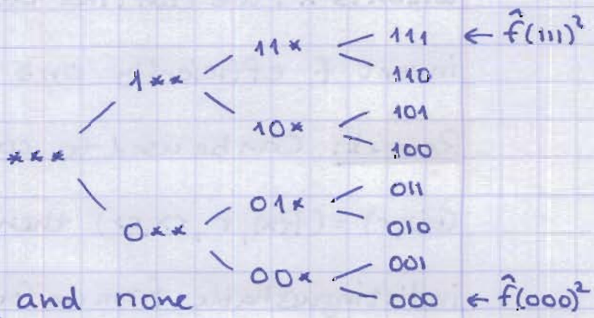
We saw in homework that we can estimate $\hat{f}(S)$ (and hence also $\hat{f}(S)^2$) to within $\pm\eta$ with confidence $\geq 1-\delta$ for any given f using $O(\log\frac{1}{\delta}/\eta^2)$ queries (or even random samples $(x, f(x))$. We can similarly estimate $\sum_S \hat{f}(S)^2$.

We now show how to estimate subsums like $\sum_{S=(1,0,1,1,*\dots*)} \hat{f}(S)^2$. Assume we want to estimate this sum over all S whose first k coordinates are $T \in \{0,1\}^k$. Define $g:\{0,1\}^{n-k} \to [-1,1]$ by $g(x) = \mathop{E}_{y\in\{0,1\}^k}[f(y,x)\cdot \chi_T(y)]$. As we saw in homework, $\forall S \in \{0,1\}^k$, $\hat{g}(S) = \hat{f}(T,S)$. Hence, our goal is to estimate $\sum_{S\in\{0,1\}^{n-k}} \hat{g}(S)^2 \overset{\text{Parseval}}{=} \mathop{E}_{x\sim\{0,1\}^{n-k}}[g(x)^2] =$

$= \mathop{E}_{x\in\{0,1\}^{n-k}}\left[\left(\mathop{E}_{y\in\{0,1\}^k}[f(y,x)\cdot\chi_T(y)]\right)^2\right] = \mathop{E}_{\substack{x\in\{0,1\}^{n-k}\\ y,y'\in\{0,1\}^k}}[f(y,x)\cdot\chi_T(y)\cdot f(y',x)\cdot\chi_T(y')]$ and we can estimate this to within $\pm\eta$ with confidence $\geq 1-\delta$ using $O(\log\frac{1}{\delta}/\eta^2)$ queries. ▨

Proof of [GL]:

We have a complete binary tree with a weight of $\hat{f}(S)^2$ associated to each leaf.



We want to find all leaves of weight $\geq \gamma^2$, and none with weight $< \frac{\gamma^2}{4}$. We can estimate the weight under any node to within $\pm\frac{\gamma^2}{4}$.

So the algorithm proceeds layer by layer, each time keeping the set of all nodes whose estimated weight is $\geq \frac{3}{4}\gamma^2$, and throwing away all nodes of estimated weight $< \frac{3}{4}\gamma^2$. At the end we output the set we found. Notice that the total weight at any level is ≤ 1, and hence our set is of size $\leq \frac{2}{\gamma^2}$ at all steps. In total, we perform $\leq \frac{2n}{\gamma^2}$ estimations of subsums. Finally, by performing these estimates with confidence $1 - \frac{\delta\cdot\gamma^2}{8n}$ we guarantee that w.p. $\geq 1-\delta$ all our estimates are correct. ▨

Application: Hard-core Predicates

Def: A permutation $f:\{0,1\}^n \to \{0,1\}^n$ is one-way if: (1). f is easy to compute.

(2). $\forall$ poly-time algorithm D and any poly p, $\Pr_x[D(f(x)) = x] < \frac{1}{p(|x|)}$.

Example: (RSA) The permutation $x \mapsto x^e \bmod N$ on $\mathbb{Z}_N^*$ for N a product of two large primes and e a random number in $\{1, 2, .., \phi(N)\}$.

We would like to have a hard bit (or hard predicate): this is an easy to compute function $B: \{0,1\}^n \to \{0,1\}$ s.t. given $f(x)$, no poly-time alg. can guess $B(x)$ w.p. $> \frac{1}{2} + \varepsilon$ for some inverse poly ε.

- Given $f: \{0,1\}^n \to \{0,1\}^n$, define $f': \{0,1\}^{2n} \to \{0,1\}^{2n}$ by $f'(x,r) = (f(x), r)$. Clearly, if f is a OWP, so is f'.

Thm: If f is one-way-permutation then $B(x,r) = (-1)^{\langle x,r \rangle}$ is a hard-core predicate for f'.

Proof: Assume by contradiction that A is a poly-time alg., that $\Pr_{x,r}[A(f(x),r) = (-1)^{\langle x,r \rangle}] \geq \frac{1}{2} + \varepsilon$ for some inverse polynomial ε. By an averaging argument, for $\frac{\varepsilon}{2}$ of all x, $\Pr_r[A(f(x),r) = (-1)^{\langle x,r \rangle}] \geq \frac{1}{2} + \frac{\varepsilon}{2}$.

Fix any such x and define $g(r) = A(f(x), r)$. Then the above says that $\hat{g}(x) \geq \varepsilon$. Using the GL algorithm we can recover a list of $O(1/\varepsilon^2)$ candidates, one of which is x. We can find out which one is x by computing f. So we managed to invert f efficiently on $\geq \frac{\varepsilon}{2}$ of the inputs, in contradiction. ∎

Remark: Can be used to construct PRGs: define $G: \{0,1\}^{2n} \to \{0,1\}^{2n+1}$ by $G(x,r) = (f(x), r, \langle x,r \rangle)$ then the output of G on a uniform input is indistinguishable from uniform on $\{0,1\}^{2n+1}$.