1 Hardcore bits

To motivate the construction of pseudorandom generators from one-way permutations, we start with the observation that since $f\colon \{0,1\}^n \to \{0,1\}^n$ is a permutation, the distribution $f(x)$ when x is uniformly random is already uniform. To get a pseudorandom generator, it is therefore sufficient to find just one more bit $h(x)$ so that $(f(x), h(x))$ is pseudorandom.

To understand what this $h(x)$ should look like, let us go back to the definition of pseudorandom generator. The definition requires that for every distinguisher D of a given size,

$$\left|\Pr_{x\sim\{0,1\}^n}[D(f(x), h(x)) = 1] - \Pr_{y\sim\{0,1\}^{n+1}}[D(y) = 1]\right| \leq \varepsilon.$$

Since $f(x)$ is uniformly distributed, we can rewrite this as

$$\left|\Pr_{x\sim\{0,1\}^n}[D(f(x), h(x)) = 1] - \Pr_{x\sim\{0,1\}^n, b\sim\{0,1\}}[D(f(x), b) = 1]\right| \leq \varepsilon.$$

This equation says that given $f(x)$, D is just about as likely to output 1 when its second input is $f(x)$ as when it is a completely random bit. Intutively, this means $f(x)$ should give almost no information about the value $h(x)$. Such a function f is called a *hardcore bit* for f.

Definition 1. Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be a function and $h\colon \{0,1\}^n \to \{0,1\}$ be a predicate. We say h is an (s,ε) *hardcore bit* for f if for every circuit P of size s (called a predictor)

$$\Pr_{x\sim\{0,1\}^n}[P(f(x)) = h(x)] \leq 1/2 + \varepsilon.$$

We now show that in order to construct a pseudorandom generator, it is sufficient to have a hardcore bit.

Claim 2. *Suppose f is a permutation and h is an (s,ε) hardcore bit for f. Then $(f(x), h(x))$ is an $(s - O(1), \varepsilon)$ pseudorandom generator.*

The proof is unfortunately somewhat technical. As usual, we will assume that $(f(x), h(x))$ is not a pseudorandom generator and show that $h(x)$ cannot be a hardcore bit for $f(x)$. But instead of working with probabilities, it will be more convenient to convert them into expectations. Usually we think of an event as a random variable Z that takes values 1 (when the event is true) or 0 (when it is false). But in this proof instead of working with the expectation of Z directly, the expression will simplify quite a bit by working with its "Fourier transform" $(-1)^Z$. In this setting probabilities and expectations are related by this formula:

$$\mathrm{E}[(-1)^Z] = \Pr[Z = 0] - \Pr[Z = 1] = 1 - 2\Pr[Z = 1] = 2\Pr[Z = 0] - 1.$$

Proof. Suppose $(f(x), h(x))$ is not an (s,ε) pseudorandom generator. By our previous discussion, there is a circuit D of size s so that

$$\left|\Pr_{x\sim\{0,1\}^n}[D(f(x), h(x)) = 1] - \Pr_{x\sim\{0,1\}^n, b\sim\{0,1\}}[D(f(x), b) = 1]\right| > \varepsilon.$$

Let us remove the absolute value (by possibly adding a negation gate at the output of D) and introduce the shorthand notation $D_x(a) = D(f(x), a)$. With this notation, we have

$$\Pr_{x,b}[D_x(b) = 1] - \Pr_{x,b}[D_x(h(x)) = 1] > \varepsilon.$$

Since $D_x(b)$ is $D_x(h(x))$ with probability $1/2$ and $D_x(\overline{h(x)})$ with probability $1/2$, we have

$$\frac{1}{2}\Pr_x[D_x(\overline{h(x)}) = 1] - \frac{1}{2}\Pr_x[D_x(h(x)) = 1] > \varepsilon.$$

Using the relation $\mathrm{E}[(-1)^Z] = 1 - 2\Pr[Z = 1]$ applied to $Z = D_x(h(x))$ and $Z = D_x(\overline{h(x)})$ we get

$$\frac{1}{2}\mathrm{E}_x[(-1)^{D_x(h(x))}] - \frac{1}{2}\mathrm{E}_x[(-1)^{D_x(\overline{h(x)})}] > 2\varepsilon.$$

By the conditional probabilities formula, we have

$$\begin{aligned}
\mathrm{E}_{x,b}[(-1)^{D_x(b)+b+h(x)}] &= \Pr[b = h(x)]\,\mathrm{E}[(-1)^{D_x(b)+b+h(x)} \mid b = h(x)] \\
&\quad + \Pr[b = \overline{h(x)}]\,\mathrm{E}[(-1)^{D_x(b)+b+h(x)} \mid b = \overline{h(x)}] \\
&= \frac{1}{2}\mathrm{E}[(-1)^{D_x(h(x))}] + \frac{1}{2}\mathrm{E}[(-1)^{D_x(\overline{h(x)})+1}] \\
&= \frac{1}{2}\mathrm{E}_x[(-1)^{D_x(h(x))}] - \frac{1}{2}\mathrm{E}_x[(-1)^{D_x(\overline{h(x)})}]
\end{aligned}$$

From where we get that

$$\mathrm{E}_{x,b}[(-1)^{D_x(b)+b+h(x)}] > 2\varepsilon.$$

This expression says that $D_x(b) + b$ as a random function in x correlates with $h(x)$, so it suggests using the following predictor P:

> P: On input z, choose a random $b \sim \{0, 1\}$ and output the bit $D(z, b) + b$.

Then $\mathrm{E}[(-1)^{P(f(x))+h(x)}] > 2\varepsilon$, and by using the relation $\mathrm{E}[(-1)^Z] = 2\Pr[Z = 0] - 1$, we get that

$$\Pr[P(f(x)) = h(x)] = \Pr[P(f(x)) + h(x) = 0] = \frac{1}{2} + \frac{1}{2}\mathrm{E}[(-1)^{P(f(x))+h(x)}] > \frac{1}{2} + \varepsilon$$

contradicting the assumption that h is an (s, ε) hardcore bit for f. □

We are now left with the task of constructing a hardcore bit for f. How should we go about this?

One thing we can try is to take the first bit x_1 of the input x as a hardcore bit for f. After thinking about it for a moment we see that, in general, this may not be a good idea: Perhaps $f(x_1 \cdots x_n)$ is of the form $(x_1, g(x_2 \ldots x_n))$ where g is a one-way permutation on $n - 1$ bits. Then x_1 is certainly not a hard-core bit for f as it can be determined completely by looking at the output.

We can extend this counterexample to conclude that for any specific input bit x_i, there may be one-way permutations f for which x_i is not a hardcore bit. This suggests that to get a hardcore bit for f, we may want to involve not one but all the input bits of f. An easy way to do this is to calculate the XOR of all the input bits of f. Is $x_1 + \cdots + x_n$ then a hardcore bit for f? You can again come up with examples where f could be a one-way permutation which reveals the value of $x_1 + \cdots + x_n$, in which case this is not a hardcore bit for f.

2 The Goldreich-Levin theorem

The amazing insight of Goldreich and Levin is that this problem can be avoided by randomizing the hardcore bit. Specifically, they suggest the following construction. Given a one-way permutation $f\colon \{0,1\}^n \to \{0,1\}^n$, it is easy to see that the function $(x,r) \to (f(x),r)$, where $r \in \{0,1\}^n$ is also a one-way permutation. Then

Theorem 3. *(Goldreich-Levin theorem) If f is a (s,ε) one-way permutation, then the predicate $h_S(x,r) = \langle x,r\rangle$ is an $(s\cdot(\varepsilon/n)^{O(1)}, O(\varepsilon))$ hardcore bit for the permutation $(f(x),r)$, where*

$$\langle x,r\rangle = x_1r_1 + \cdots + x_nr_n \mod 2.$$

The proof of this theorem goes by contradiction. Let $s' = s\cdot(\varepsilon/n)^{O(1)}$ and $\varepsilon' = O(\varepsilon)$ and let's assume that $\langle x,r\rangle$ is not (s',ε') hardcore for $(f(x),r)$. Then there is a predictor P for which

$$\Pr_{x,r}\left[P(f(x),r) = \langle x,r\rangle\right] > \frac{1}{2} + \epsilon.$$

It follows that

$$\Pr_x\left[\Pr_r\left[P(f(x),r) = \langle x,r\rangle\right] > \frac{1}{2} + \frac{\epsilon}{2}\right] > \frac{\epsilon}{2}$$

Let S be the set of all x such that

$$\Pr_r\left[P(f(x),r) = \langle x,r\rangle\right] > \frac{1}{2} + \frac{\epsilon}{2} \tag{1}$$

This suggests the following algorithm for inverting $f(x)$ when $x \in S$: On input $z = f(x)$, try to find all a's such that $\Pr_r\left[P(z,r) = \langle a,r\rangle\right] > 1/2 + \epsilon/2$. Since x satisfies (1), one of these a's must equal x. To find out which one, apply f to all of them and see which one maps to z. Since f is a permutation, if $f(a) = z$ it must be that $a = x$.

Can we implement this algorithm efficiently? At first, the idea seems unreasonable: It looks like there might be exponentially many x' such that $\Pr_r\left[P(z,r) = \langle x',r\rangle\right] > 1/2 + \epsilon/2$, so even listing all of them, much less computing them, may take too much time. However, this is not the case, and the search of x' can be completed efficiently:

Lemma 4. *There is a randomized algorithm $A^?$ which on input ε and given oracle access to $g : \{0,1\}^n \to \{0,1\}$ runs in time $(s/\varepsilon)^{O(1)}$ and with probability at least $1/2$ outputs a list that contains all x such that*

$$\Pr_r\left[g(r) = \langle x,r\rangle\right] \geq \frac{1}{2} + \frac{\varepsilon}{2}.$$

There are two proofs of the Goldreich-Levin theorem. We will present the proof of Kushilevitz and Mansour, which uses Fourier analysis of Boolean functions.

3 Fourier analysis

Fourier analysis is a general technique for the study of functions $f\colon \{0,1\}^n \to \mathbb{R}$. We can think of a function f from $\{0,1\}^n$ to the real numbers as a vector in 2^n dimensional space. The standard

basis for this space consists of those vectors that have entry 1 in exactly one of the 2^n positions, and entry 0 everywhere else. These are the 2^n functions δ_a ($a \in \{0,1\}^n$), which take value 0 everywhere except at a, where they take value 1.

Now any function f over $\{0,1\}^n$ can be written as a linear combination of the functions δ_a. The coefficient in front of δ_a – the value $f(a)$ – is then simply just the inner product between f and δ_a.

The Fourier transform allows us to express f in a different orthogonal basis. This is the basis consisting of the character functions $\chi_a \colon \{0,1\}^n \to \mathbb{R}$

$$\chi_a(x) = (-1)^{\langle x,a \rangle}, \qquad a \in \{0,1\}^n.$$

It is easy to verify that, viewed as vectors, the functions χ_a are orthogonal and each has norm $2^{n/2}$. Instead of normalizing these functions, it is more convenient to work with respect to the normalized inner product

$$\text{inner product of } f \text{ and } g = \mathrm{E}_{x \sim \{0,1\}^n}[f(x)g(x)] = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} f(x)g(x).$$

In this basis, a function $f \colon \{0,1\}^n \to \mathbb{R}$ can be written as a linear combination of the χ_as:

$$f(x) = \sum_{a \in \{0,1\}^n} \hat{f}_a \cdot \chi_a(x)$$

where the coefficient $\hat{f}_a$ is just the inner product of f and χ_a:

$$\hat{f}_a = \mathrm{E}_{x \sim \{0,1\}^n}[f(x)\chi_a(x)]. \tag{2}$$

Parseval's identity Since the bases $\{\delta_a\}$ and $\{\chi_a\}$ are related by an orthonormal transformation (up to the normalization factor $2^{n/2}$), the sum of the squares of the coefficients of f in these two bases should be the same. This is Parseval's identity:

$$\sum_{a \in \{0,1\}^n} \hat{f}_a^2 = \mathrm{E}_{x \sim \{0,1\}^n}[f(x)^2].$$

We can also prove this algebraically by a calculation:

$$\begin{aligned}
\mathrm{E}_{x \sim \{0,1\}^n}[f(x)^2] &= \mathrm{E}_{x \sim \{0,1\}^n}\left[\left(\textstyle\sum_{a \in \{0,1\}^n} \hat{f}_a \chi_a(x)\right)^2\right] \\
&= \mathrm{E}_{x \sim \{0,1\}^n}\left[\textstyle\sum_{a,b \in \{0,1\}^n} \hat{f}_a \hat{f}_b \chi_a(x)\chi_b(x)\right] \\
&= \textstyle\sum_{a,b \in \{0,1\}^n} \hat{f}_a \hat{f}_b \, \mathrm{E}_{x \sim \{0,1\}^n}[\chi_a(x)\chi_b(x)] \\
&= \textstyle\sum_{a \in \{0,1\}^n} \hat{f}_a^2
\end{aligned}$$

as the only surviving terms in the summation over a and b are those where $a = b$.

Corollary 5. *if f is a function from $\{0,1\}^n$ to $\{-1,1\}$, then $\sum_{a \in \{0,1\}^n} \hat{f}_a^2 = 1$.*

4 Proof of the Goldreich-Levin theorem

To prove Lemma 4, we now give an algorithm that given oracle access to $g\colon \{0,1\}^n \to \{0,1\}$ outputs all a's such that

$$\Pr[g(r) = \langle a, r\rangle] \geq \frac{1}{2} + \frac{\varepsilon}{2}.$$

We want to see what Fourier analysis tells us about the function g. Instead of working with g directly, it will be more convenient to look at the function $G(r) = (-1)^{g(r)}$ (which takes values -1 and 1) because

$$\mathrm{E}[G(r)\chi_a(r)] = 2\Pr[g(r) = \langle a, r\rangle] - 1$$

so now we simply need to find all $a \in \{0,1\}^n$ for which

$$\hat{G}_a = \mathrm{E}[G(r)\chi_a(r)] \geq \varepsilon. \tag{3}$$

Every a that satisfies (3) contributes at least ε^2 to the sum $\sum_{a\in\{0,1\}^n} \hat{G}_a^2$, which equals 1 by Corollary 5, so there can be at most $1/\varepsilon^2$ such a's.

How do we find these a's? We will generalize our objective a little bit and seek to find all a such that $\hat{G}_a^2 \geq \varepsilon^2$, and maybe even allow for a few a's that don't quite satisfy this condition. The idea is to try to locate these relevant as by a divide-and-conquer strategy. One nice way to visualize this strategy is as a search process along the following full binary tree of depth n. The root ε of this binary tree is labeled by the value $\sum_{a\in\{0,1\}^n} \hat{G}_a^2$. Its left and right children 0 and 1 are labeled by the partial sums

$$\sum_{a\colon a_1=0} \hat{G}_a^2 \quad \text{and} \quad \sum_{a\colon a_1=1} \hat{G}_a^2.$$

In general, a node at level i can be indexed by a string $v \in \{0,1\}^i$ and is labeled by the value

$$\ell(v) = \sum_{a\colon a_1=v_1,\ldots,a_i=v_i} \hat{G}_a^2$$

so that the leaf indexed by a is labeled by $\hat{G}_a^2$.

Say a node v is relevant if $\ell(v) \geq \varepsilon^2$. Although there are exponentially many nodes in the tree, there can be at most n/ε^2 relevant ones because the labels in each level sum to 1. If we could calculate the labels, it would be easy to identify all the relevant nodes via depth-first search starting at the root and pruning the search path at irrelevant nodes.

How do we calculate the values of the labels? Using the Fourier coefficient formula (2), we can obtain these values in time exponential in n. But if we are willing to settle for a probabilistic approximation, we can do much better. Let's start at the leaves. From the formula (2) we get

$$\hat{G}_a^2 = \mathrm{E}_x[G(x)\chi_a(x)]^2 = \mathrm{E}_x[G(x)\chi_a(x)]\,\mathrm{E}_y[G(y)\chi_a(y)] = \mathrm{E}_{x,y}[G(x)G(y)\chi_a(x+y)].$$

This suggests that to estimate $\hat{G}_a^2$, we ought to sample some number of random pairs (x, y) and output the average of the values $G(x)G(y)\chi_a(x+y)$.

Now let v be an arbitrary node in the tree and $FIX(v)$ be the set of those $a \in \{0,1\}^n$ with $a_1 = v_1, \ldots, a_i = v_i$. We want to estimate the value

$$\ell(v) = \sum_{a\in FIX(v)} \hat{G}_a^2 = \mathrm{E}\, G(x)G(y) \sum_{a\in FIX(v)} \chi_a(x+y).$$

The set $FIX(v)$ could be exponentially large so we have to be a bit careful here. Recall that $\chi_a(z) = (-1)^{\langle a,z\rangle}$ so:

$$\sum_{a \in FIX(v)} \chi_a(z) = \sum_{a \in FIX(v)} (-1)^{\langle a,z\rangle}$$

If z is nonzero along any of the coordinates $i+1$ up to n, this sum vanishes; otherwise, it equals $2^{n-i}\chi_v(z)$. So the only (x,y) pairs that contribute to the sum are those in which x and y agree on the last $n-i$ coordinates, and we can rewrite the identity as

$$\ell(v) = \mathrm{E}_{x',y' \sim \{0,1\}^i, u \sim \{0,1\}^{n-i}}[G(x'u)G(y'u)\chi_v(x'+y')].$$

Here, the first i bits x' and y' of x and y are chosen independently at random, while the last $n-i$ bits denoted by u are random but identical in x and y. (When $i = 0$ the right side equals $\mathrm{E}[G(u)^2] = 1$, which is consistent with Parseval's identity.)

We now have all the ingredients for the Kushilevitz-Mansour algorithm. First, we have a probabilistic procedure $\hat{\mathbf{S}}\mathbf{amp}^G(v)$ which estimates the label of node v as follows:

> $\hat{\mathbf{S}}\mathbf{amp}^G(v)$: Sample $O(n/\varepsilon^4)$ independent random triples (x', y', u) and output the average of the values $G(x'u)G(y'u)\chi_v(x'+y')$.

The following lemma follows directly from Corollary 9 below.

Lemma 6. *With probability at least $1 - \varepsilon^2/20n$, $\hat{\mathbf{S}}\mathbf{amp}^G(v)$ outputs a value between $\ell(v) - \varepsilon^2/6$ and $\ell(v) + \varepsilon^2/6$.*

Now here is the Kushilevitz-Mansour algorithm:

> **Algorithm $\mathbf{KM}^G(\varepsilon)$**, where G is a function from $\{0,1\}^n$ to $\{-1,1\}$ and $\varepsilon > 0$:
> Apply the following recursive procedure $\mathbf{P}(v)$ starting with v equal to the empty string:
> If $\hat{\mathbf{S}}\mathbf{amp}^G(v) \geq \varepsilon^2/2$:
> If v has length n, output v.
> Otherwise, call $\mathbf{P}(v0)$ and $\mathbf{P}(v1)$.

Theorem 7. *With probability at least $1/2$, the outputs of $\mathbf{KM}^G(\varepsilon)$ include all a such that $\hat{G}_a^2 \geq \varepsilon^2$, but it produces no more than $O(n/\varepsilon^2)$ outputs in total.*

Proof. Let v be any node such that $\ell(v) \geq \varepsilon^2$. By Lemma 6,

$$\Pr[\hat{\mathbf{S}}\mathbf{amp}^G < \varepsilon^2/2] \leq \varepsilon^2/20n$$

Since there are at most n/ε^2 such nodes v, by a union bound we have

$$\Pr[\hat{\mathbf{S}}\mathbf{amp}^G < \varepsilon^2/2 \text{ for some } v \text{ s.t. } \ell(v) \geq \varepsilon^2] \leq \frac{n}{\varepsilon^2} \cdot \frac{\varepsilon^2}{20n} \leq \frac{1}{20}.$$

Therefore, all $a \in \{0,1\}^n$ such that $\ell(a) = \hat{G}_a^2 \geq \varepsilon^2$ will be included in the output of $\mathbf{KM}^G(\varepsilon)$ with probability at least $1 - 1/20 = 19/20$.

Let B be the set of nodes whose label exceeds $\varepsilon^2/3$ and B' be the set of nodes outside B whose parent node is in B. Since the nodes in B form a tree, we must have $|B'| \leq |B| + 1$. There must be fewer than $3n/\varepsilon^2$ nodes in B, so B' can have at most $3n/\varepsilon^2 + 1$ nodes. By a very similar calculation as above,

$$\Pr[\hat{\mathbf{Samp}}^G \geq \varepsilon^2/2 \text{ for some } v \text{ in } B'] \leq \left(\frac{3n}{\varepsilon^2} + 1\right) \cdot \frac{\varepsilon^2}{20n} \leq \frac{1}{5}.$$

Therefore, with probability at least 4/5, $\hat{\mathbf{Samp}}$ will output a value smaller than than $\varepsilon^2/2$ on all nodes in B', so $\mathbf{KM}^G(\varepsilon)$ will not make any recursive calls to $\mathbf{P}$ on a node outside $B \cup B'$. Since there are at most $O(n/\varepsilon^2)$ nodes inside $B \cup B'$, $\mathbf{KM}^G(\varepsilon)$ can produce at most this many outputs.

With probability at least $1 - 1/20 - 1/5 \geq 1/2$, both of these conditions are met. □

A Chebyshev's inequality

Recall the variance of a random variable X is given by the formula

$$\mathrm{Var}[X] = \mathrm{E}[X^2] - \mathrm{E}[X]^2 = \mathrm{E}[(X - \mathrm{E}[X])^2].$$

Let $X_1, \ldots, X_m$. Large deviation inequalities give tail bounds on the distribution of $X = X_1 + \cdots + X_m$. Recall that $X_1, \ldots, X_m$ are pairwise independent if for every pair $i \neq j$, X_i and X_j are statistically independent. If $X_1, \ldots, X_m$ are pairwise independent, then

$$\mathrm{Var}[X] = \mathrm{Var}[X_1] + \cdots + \mathrm{Var}[X_m].$$

However, this is not true in general. Chebyshev's inequality gives deviation bounds on the random variable X. It always holds, but it is most useful when X is a sum of pairwise independent random variables (or almost so):

Theorem 8 (Chebyshev inequality)**.** *For every random variable X*

$$\Pr\left[|X - \mathrm{E}[X]| \geq t\sqrt{\mathrm{Var}[X]}\right] \leq \frac{1}{t^2}.$$

Proof. We apply Markov's inequality to the random variable $Y = (X - \mathrm{E}[X])^2$. □

In particular we have the following corollary, which allows us to get estimates for the average of a random variable from random sampling.

Corollary 9. *Let $X_1, \ldots, X_m$ be pairwise independent and identically distributed random variables taking values in $[-1, 1]$. Let $X = (X_1 + \cdots + X_m)/m$ and $\mu = \mathrm{E}[X_i] = \mathrm{E}[X]$. For every $\delta > 0$,*

$$\Pr\left[|X - \mu| \geq \delta\right] \leq \frac{1}{\delta^2 m}.$$

Proof. We apply Chebyshev's inequality to X. Notice that

$$\mathrm{Var}[X] = \frac{1}{m^2}(\mathrm{Var}[X_1] + \cdots + \mathrm{Var}[X_m]) \leq \frac{\mathrm{Var}[X_1]}{m} \leq \frac{\mathrm{E}[X_1^2]}{m} \leq \frac{1}{m}$$

and the corollary follows by choosing $t = \delta\sqrt{m}$ in Chebyshev's inequality. □