

Lecture 5: PRG Construction (Cont.)

Notes by Yael Kalai

MIT - 6.5620

Lecture 5 (September 17, 2025)

Warning: This document is a rough draft, so it may contain bugs. Please feel free to email me with corrections.

Recap

Last lecture we started to show how to construct a PRG from any OWP f .

1. We first showed that it suffices to construct a PRG with a single bit stretch, i.e. a PRG

$$G : \{0,1\}^\lambda \rightarrow \{0,1\}^{\lambda+1}.$$

This is the case since we showed that we can stretch any such PRG into one that has a polynomial stretch $k = \text{poly}(\lambda)$ by concatenating G k times

$$G^k : \{0,1\}^\lambda \rightarrow \{0,1\}^{\lambda+k}.$$

2. We then defined the notion of a *hardcore predicate*

Definition 1. $P : \{0,1\}^\lambda \rightarrow \{0,1\}$ is a hardcore predicate of f if for every poly-size \mathcal{A} there exists a negligible function μ such that for every $\lambda \in \mathbb{N}$,

$$\Pr_{x \leftarrow \{0,1\}^\lambda} [\mathcal{A}(f(x)) = P(x)] \leq \frac{1}{2} + \mu(\lambda).$$

3. We showed that if the OWP f had a hardcore predicate $P : \{0,1\}^\lambda \rightarrow \{0,1\}$ then the following function is a PRG:

$$G(x) = f(x) \circ P(x).$$

Thus, it remains to argue that every OWP has a hardcore predicate. Goldreich and Levin proved that every one-way function has a randomized hardcore predicate.

Goldreich-Levin Theorem

Theorem 2. [1] If f is a one-way function, then the following randomized predicate

$$P(x, r) := x \cdot r \mod 2 = \sum_{i \in \lambda} x_i r_i \mod 2$$

is a hardcore predicate for f . Namely, for every poly-size \mathcal{A} there exists a negligible function $\mu : \mathbb{N} \rightarrow [0, 1]$ such that for every $\lambda \in \mathbb{N}$

$$\Pr_{U_\lambda \leftarrow \{0,1\}^\lambda} [\mathcal{A}(f(U_\lambda), r) = P(U_\lambda, r)] \leq \frac{1}{2} + \mu(\lambda)$$

This theorem implies that

$$G(x, r) = f(x) \circ r \circ P(x, r)$$

is a PRG. In this lecture our focus is on proving Theorem 2.

Proof of Theorem 2 Suppose for contradiction that there exists a poly-size adversary \mathcal{A} and a non-negligible function $\epsilon : \mathbb{N} \rightarrow [0, 1]$ such that for every $\lambda \in \mathbb{N}$,

$$\Pr_{U_\lambda \leftarrow \{0,1\}^\lambda} [\mathcal{A}(f(U_\lambda), r) = P(U_\lambda, r)] \geq \frac{1}{2} + \epsilon(\lambda)$$

We will show how to use \mathcal{A} to construct an adversary \mathcal{B} that breaks the one-wayness of f .

Since \mathcal{A} is only assumed to predict with small advantage ϵ , we cannot expect \mathcal{B} to invert every given image $y = f(x)$. Indeed, it may be that \mathcal{A} always fails when given an input whose first λ -bits is y . So which y 's can we hope to invert?

Define

$$\text{GOOD} = \{x \in \{0,1\}^\lambda : \Pr[\mathcal{A}(f(x), r) = x \cdot r] \geq \frac{1}{2} + \frac{\epsilon(\lambda)}{2}\}$$

By Markov's inequality

$$\Pr_{x \leftarrow U_\lambda} [x \in \text{GOOD}] \geq \frac{\epsilon(\lambda)}{2}$$

To expand on this, denoting by

$$p = \Pr_{x \leftarrow U_\lambda} [x \in \text{GOOD}],$$

it holds that

$$\begin{aligned}
\frac{1}{2} + \epsilon(\lambda) &\leq \\
\Pr_{x \leftarrow U_\lambda} [\mathcal{A}(f(x), r) = P(x, r)] &= \\
\Pr_{x \leftarrow \text{GOOD}} [\mathcal{A}(f(x), r) = P(x, r)] \cdot \Pr_{x \leftarrow U_\lambda} [x \in \text{GOOD}] &+ \Pr_{x \leftarrow \neg \text{GOOD}} [\mathcal{A}(f(x), r) = P(x, r)] \cdot \Pr_{x \leftarrow U_\lambda} [x \notin \text{GOOD}] = \\
\Pr_{x \leftarrow \text{GOOD}} [\mathcal{A}(f(x), r) = P(x, r)] \cdot p &+ \Pr_{x \leftarrow \neg \text{GOOD}} [\mathcal{A}(f(x), r) = P(x, r)] \cdot (1 - p) \leq \\
p + \frac{1}{2} + \frac{\epsilon(\lambda)}{2}, &
\end{aligned}$$

which implies that indeed $p \geq \frac{\epsilon(\lambda)}{2}$.

We will construct an algorithm \mathcal{B} that succeeds in inverting every $f(x)$ for $x \in \text{GOOD}$ with non-negligible probability.

Special case 1: \mathcal{A} predicts $P(x, r)$ perfectly for every $x \in \text{GOOD}$

As a warmup, suppose that \mathcal{A} is a perfect predictor that for every $x \in \text{GOOD}$ and every $r \in \{0, 1\}^\lambda$

$$\mathcal{A}(f(x), r) = x \cdot r$$

With such a perfect predictor it is easy to invert the OWF f . Specifically, the inverter \mathcal{B} , given $y = f(x)$ where $x \leftarrow U_\lambda$, does the following:

1. For every $i \in [\lambda]$ compute $x'_i = \mathcal{A}(y, e_i)$.
2. Output $x = (x'_1, \dots, x'_\lambda)$.

By our assumption that \mathcal{A} is a perfect predictor, it follows that

$$x'_i = x \cdot e_i = x_i,$$

as desired.

Unfortunately, our predictor \mathcal{A} may not be perfect. Next, we relax the perfect condition as follows.

Special case 2: There exists a non-negligible $\epsilon = \epsilon(\lambda)$ such that for every $x \in \text{GOOD}$, \mathcal{A} predicts $P(x, r)$ with probability $3/4 + \epsilon$

We can use this predictor to construct an algorithm \mathcal{B} that for every $x \in \text{GOOD}$, given $f(x)$ finds an inverse with overwhelming probability (i.e., probability $1 - \mu$ for some negligible function μ). The inverter \mathcal{B} , given $y = f(x)$ where $x \leftarrow U_\lambda$, does the following:

1. Set $k = \lambda/\epsilon^2$.
2. For every $i \in [\lambda]$, do the following:

- (a) Choose at random $r_{i,1}, \dots, r_{i,k} \leftarrow \{0,1\}^\lambda$.
 - (b) For every $j \in [k]$ compute $b_{i,j} = \mathcal{A}(y, r_{i,j})$ and $b'_{i,j} = \mathcal{A}(y, r_{i,j} \oplus e_i)$, and let $x_{i,j} = b_{i,j} \oplus b'_{i,j}$.
 - (c) Let $x'_i = \text{majority}\{x_{i,1}, \dots, x_{i,k}\}$
3. Output $x' = (x'_1, \dots, x'_\lambda)$

Note that by our assumption for every $i \in [\lambda]$ and $j \in [k]$,

$$\begin{aligned}
 & \Pr_{r_{i,j} \leftarrow \{0,1\}^\lambda} [x'_{i,j} = x_i] \geq \\
 & \Pr_{r_{i,j} \leftarrow \{0,1\}^\lambda} [\mathcal{A}(y, r_{i,j}) = x \cdot r_{i,j} \wedge \mathcal{A}(y, r_{i,j} \oplus e_i) = x \cdot (r_{i,j} \oplus e_i)] = \\
 & 1 - \Pr_{r_{i,j} \leftarrow \{0,1\}^\lambda} [\mathcal{A}(y, r_{i,j}) \neq x \cdot r_{i,j} \vee \mathcal{A}(y, r_{i,j} \oplus e_i) \neq x \cdot (r_{i,j} \oplus e_i)] \geq \\
 & 1 - \Pr_{r_{i,j} \leftarrow \{0,1\}^\lambda} [\mathcal{A}(y, r_{i,j}) \neq x \cdot r_{i,j}] - \Pr_{r_{i,j} \leftarrow \{0,1\}^\lambda} [\mathcal{A}(y, r_{i,j} \oplus e_i) \neq x \cdot (r_{i,j} \oplus e_i)] \geq \\
 & 1 - 1/2 + 2\epsilon = 1/2 + 2\epsilon
 \end{aligned}$$

Note that for every $i \in [\lambda]$ it holds that $x'_{i,1}, \dots, x'_{i,k}$ are independent Boolean random variables such that $\Pr[x'_{i,j} = x_i] \geq 1/2 + 2\epsilon$. Therefore, by the Chernoff bound

$$\Pr[\text{majority}\{x_{i,1}, \dots, x_{i,\lambda}\} \neq x_i] \leq 2^{-O(\epsilon^2 \cdot k)}$$

Theorem 3 (Chernoff bound). *Let X_1, X_2, \dots, X_n be independent random variables taking values in $\{0, 1\}$ (Bernoulli trials). Let*

$$X = \sum_{i=1}^n X_i, \quad \mu = \mathbb{E}[X].$$

For any $0 < \delta < 1$:

$$\Pr[X \geq (1 + \delta)\mu] \leq \exp\left(-\frac{\delta^2}{3}\mu\right),$$

and

$$\Pr[X \leq (1 - \delta)\mu] \leq \exp\left(-\frac{\delta^2}{2}\mu\right).$$

Thus, for every $i \in [\lambda]$

$$\Pr[\mathcal{B}(f(x)) = x' : x'_i \neq x_i] \leq 2^{-O(\lambda)},$$

which together with the union bound implies that

$$\Pr[\mathcal{B}(f(x)) \neq x] \leq \lambda \cdot 2^{-O(\lambda)} = \text{negl}(\lambda).$$

Remark. Note that we could have chosen the same random $r_1, \dots, r_k \leftarrow \{0,1\}^\lambda$ for all the coordinates $i \in [\lambda]$. The reason is that we do not need independence for the union bound.

General case: There exists a non-negligible $\epsilon = \epsilon(\lambda)$ such that for every $x \in \text{GOOD}$, \mathcal{A} predicts $P(x, r)$ with probability $1/2 + \epsilon$

Note that in the above analysis the reason that we needed a success probability of greater than $\frac{3}{4}$ is because, in order to get a non-negligible probability of guessing a single bit x_i we needed to combine *two* guesses for two bits $x \cdot r_{i,j}$ and $x \cdot (r_{i,j} \oplus e_i)$.

The Goldreich-Levin reduction starts with the following (ridiculous!) idea. Suppose that for each pair r and $r \oplus e_i$, we simply guess the bit $x \cdot r$ ourselves, and we only use the adversary \mathcal{A} to guess the bit $x \cdot (r \oplus e_i)$. If we could somehow manage to always guess the bit $x \cdot r$ correctly then we could use the same argument as above since under this (ridiculous) assumption both bits $x \cdot r$ and $x \cdot (r \oplus e_i)$ are guessed correctly with probability at least $\frac{1}{2} + \epsilon$.

The problem is that we cannot guess $x \cdot r$ by ourselves with probability better than random. This is exactly the task that we needed \mathcal{A} to do in the first place! But we can guess each of these bits at random and have a success probability $\frac{1}{2}$. This seems useless since in the above procedure, for each $i \in [\lambda]$ we needed to choose a total of k different values of $x \cdot r_j$, and if we guess each of them at random then we will guess all of them correctly only with probability 2^{-k} . But it turns out that this random guessing is actually really useful, due to the following clever idea: suppose that we happen to correctly guess $x \cdot s_1$ and $x \cdot s_2$. Then we can also guess correctly

$$x \cdot (s_1 \oplus s_2) = (x \cdot s_1) \oplus (x \cdot s_2)$$

More generally, suppose we guessed correctly

$$x \cdot r_1, x \cdot r_2, \dots, x \cdot r_\ell$$

then we can use these ℓ bits to guess correctly

$$x \cdot (\oplus_{i \in I} r_i) = \oplus_{i \in I} (x \cdot s_i)$$

for every $I \subseteq [\ell]$. Since there are 2^ℓ such subsets we learned 2^ℓ inner products $x \cdot r_I$, where $r_I = \oplus_{i \in I} r_i$. Therefore, in order to learn k different inner products $x \cdot r_I$ we need to guess only $\ell = \log k$ many bits of the form $x \cdot r_j$! And we can guess all ℓ inner products correctly with probability $\frac{1}{k}$.

You may be concerned that now all these r_I 's are not independent. Indeed, they are not! For example, $r_{\{1,2\}} = r_1 \oplus r_2$. As a result we cannot use the Chernoff bound in the analysis. However, they are *pairwise independent*. Therefore, instead of Chernoff bound we can rely on Chebyshev's inequality which is a weaker concentration bound.

Theorem 4 (Chebyshev's inequality). Let Z_1, \dots, Z_m be pairwise independent random variables obtaining values in $\{0, 1\}$, such that for every $j \in [m]$ $\Pr[Z_j = 1] = p$. Then, for every $\delta > 0$,

$$\Pr \left[\left| \frac{\sum_{j=1}^m Z_j}{m} - p \right| \geq \delta \right] \leq \frac{1}{4\delta^2 m}$$

So, the inverter \mathcal{B} , given $y = f(x)$, does the following:

1. Set $k = \lambda / \epsilon^2$.
2. For every $i \in [\lambda]$, do the following:
 - (a) Set $\ell \approx \log k$
 - (b) choose at random $r_1, \dots, r_\ell \leftarrow \{0, 1\}^\lambda$.
 - (c) For every $j \in [\ell]$, choose at random $b_j \leftarrow \{0, 1\}$.¹
 - (d) For every $J \subseteq [\ell]$, let

¹ b_j is a guess for $x \cdot r_j$.

$$r_J = \oplus_{j \in J} r_j \quad \text{and} \quad b_J = \oplus_{j \in J} b_j.$$

- (e) Similarly, for every $J \subseteq [\ell]$, let

$$b'_{i,J} = \mathcal{A}(y, r_J \oplus e_i) \quad \text{and} \quad x_{i,J} = b_J \oplus b'_{i,J}.$$

- (f) Let $x'_i = \text{majority}\{x_{i,J}\}_{J \subseteq [\ell]}$

3. Output $x' = (x'_1, \dots, x'_\lambda)$

Suppose that all our ℓ guesses were correct, which happens with probability $2^{-\ell} = \text{poly}(k)$. In this case, as above for every $i \in [\lambda]$ and $J \subseteq [\ell]$,

$$\begin{aligned} & \Pr_{r_1, \dots, r_\ell \leftarrow \{0,1\}^\lambda} [x'_{i,J} = x_i] \geq \\ & \Pr_{r_1, \dots, r_\ell \leftarrow \{0,1\}^\lambda} [\mathcal{A}(y, (r_J + e_i)) = x \cdot (r_J \oplus e_i)] = \\ & 1 - \Pr_{r_1, \dots, r_\ell \leftarrow \{0,1\}^\lambda} [\mathcal{A}(y, r_J \oplus e_i) \neq x \cdot (r_J \oplus e_i)] \geq \\ & 1/2 + \epsilon. \end{aligned}$$

This seems great, except that now we cannot apply the Chernoff bound since the random variables $\{r_J\}_{J \subseteq [\ell]}$ are not independent! However, as mentioned above they are pairwise independent, and hence we can rely on Chebyshev's inequality to argue that for every $x \in \text{GOOD}$ and every $i \in [\lambda]$,

$$\Pr[x'_i \neq x_i] \leq \frac{1}{4\epsilon^2 k}$$

In particular, setting $k = \lambda \cdot \epsilon^{-2}$, we can take a union bound, and guarantee that for every $x \in \text{GOOD}$,

$$\Pr[\mathcal{B}(f(x)) = x] \geq \frac{1}{4},$$

contradicting the fact that f is a one-way function.

□

References

- [1] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC)*, pages 25–32, Seattle, Washington, USA, 1989.