9

Dealing with Non-Perfect Randomness

Great Ideas in Theoretical Computer Science
Saarland University, Summer 2014

In the last lecture we discussed pseudorandom generators, which are deterministic polynomial-time algorithms that, given a truly random binary sequence of length n, output a pseudorandom sequence of length $\ell(n) > n$, which is computationally indistinguishable by any polynomial-time algorithm. However, in reality truly random sources are rare: in most situations we do not know how to obtain such perfect random sequences, i.e. bits that are unbiased and completely independent. Instead, we can obtain random sequences through physical devices: These sequences are not perfect random, but have certain "randomness" inside. Can we use these non-perfect random sequences in randomized algorithms? Can we further bound the error probability of one randomized algorithm when using this non-perfect random source? In today's lecture we will address these questions, and introduce another combinatorial object, called *extractors*. We will show how extractors can be used to deal with non-perfect randomness in algorithm design.

9.1 Extractors

In Information Theory, entropy is a measure of uncertainty in a random variable. We look at two definitions of entropy: *Shannon entropy* quantities the expected value of the information contained in a message, and *min-entropy* quantities how far in the worst case the distribution is from the uniform distribution.

Definition 9.1 (Entropy)**.** *Let X be a random variable. Then the entropy $\mathbf{H}(X)$ of X is defined as*

$$\mathbf{H}(X) \triangleq \mathbf{E}\left[-\log\left(\mathbf{Pr}[\,X = x\,]\right)\right].$$

Definition 9.2 (min-Entropy)**.** *Let X be a random variable. Then the min-entropy $\mathbf{H}_\infty(X)$ of X is defined by*

$$\mathbf{H}(X) \triangleq \min\left\{-\log\left(\mathbf{Pr}[\,X = x\,]\right)\right\}.$$

It is not hard to see that if X is a random variable over $\{0,1\}^n$, then $\mathbf{H}_\infty(X) \leq n$ with $\mathbf{H}_\infty(X) = n$ if and only if X is distributed according to the uniform distribution $\mathcal{U}_n$. Hence,

the bigger the value of $\mathbf{H}_\infty(X)$, the more randomness inside X.

Lemma 9.3. *It holds that $\mathbf{H}_\infty(X) \leq \mathbf{H}(X)$ for any X.*

For practical applications, randomness is from physical devices. These devices generate random sequences with certain randomness. To measure the randomness of the sequence generated by these devices, we introduce the notion of k-source.

Definition 9.4 (k-source)**.** *We call X a k-source if $\mathbf{H}_\infty(X) \geq k$.*

Example 9.5. *Here are some examples of k-sources:*

- *k random and independent bits, together with $n-k$ fixed bits (in an arbitrary order). They are called oblivious bit-fixing sources.*
- *k random and independent bits, and $n-k$ bits that depend arbitrarily on the first k bits. They are called adaptive bit-fixing sources.*
- *Uniform distribution on $S \subseteq \{0,1\}^n$ with $|S| = 2^k$. These are called flat k-sources.*

Proposition 9.6. *Every k-source is a convex combination of flat k-sources (provided that $2^k \in \mathbb{N}$), i. e. $X = \sum_i p_i X_i$ with $0 \leq p_i \leq 1, \sum p_i = 1$ and all the X_i are flat k-sources.*

Definition 9.7 (statistical difference)**.** *For random variables X and Y taking values in U, their statistical difference is defined by*

$$\Delta(X,Y) \triangleq \max_{T \subseteq U} |\mathbf{Pr}[X \in T] - \mathbf{Pr}[Y \in T]|.$$

We say that X and Y are ε-close if $\Delta(X,Y) \leq \varepsilon$.

Lemma 9.8.

$$\Delta(X,Y) = \frac{1}{2} \cdot \sum_{z \in T} |\mathbf{Pr}[X = z] - \mathbf{Pr}[Y = z]|$$

After these preparations, we are ready to define extractors. Informally, extractors are functions which extract random bits from any distribution which contains sufficient randomness.

Definition 9.9 (seeded extractors, [3])**.** *A (k,ε)-extractor is a function $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \mapsto \{0,1\}^m$ such that, for every distribution X on $\{0,1\}^n$ with $\mathbf{H}_\infty(X) \geq k$, the distribution $\mathsf{Ext}(X,\mathcal{U}_d)$ is ε-close to $\mathcal{U}_m$.*

By definition, a (k,ε)-extractor has five parameters: (1) The length of the source n; (2) the output length m; (3) the length of seeds d; (4) the min-entropy threshold k; (5) the error of the extractor ε. Here, we refer to the ratio k/n as *the entropy rate* of the source X, and to the ratio m/k as *the fraction of randomness* extracted by Ext. The goal of constructing good extractors is to minimize d and maximize m. Notice that a trivial case is when $d \geq m$, in which case the extractor can simply ignore its first input and output the seed.

Extractors have various applications in Theoretical Computer Science, including random sampling using few random bits; constructions of expanders that beat the eigenvalue bound;

explicit constructions of error-correcting codes. In designing randomized algorithms, extractors are used to simulate randomized algorithms with non-perfect random sequences.

The framework of using extractors to simulate randomized algorithms is as follows: Assume that algorithm A uses m random bits. Since we do not have truly random bits, algorithm A uses the "almost random" strings to instead the perfect random ones. That is, the random strings for A come from the output of $\mathsf{Ext}(X, \mathcal{U}_d)$. Since the seed length d is small, we can eliminate this part of randomness by running all the possible seeds and taking the majority value. In particular, an explicit extractor with logarithmic seed length can be used to simulate any polynomial-time randomized algorithm given access to a weak random source of sufficient high min-entropy.

Theorem 9.10. *Let $A(w; r)$ be a randomized algorithm such that $A(w; \mathcal{U}_m)$ has error probability at most γ, and let $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be a (k, ε)-extractor. Define*

$$A' = \mathrm{maj}_{y \in \{0,1\}^d} \{A(w, \mathsf{Ext}(x, y))\}.$$

Then for every k-source X on $\{0,1\}^n$, $A'(w; X)$ has error probability at most $2(\gamma + \varepsilon)$.

9.2 Extractors as Hash Functions

Throughout the rest of the note, capital variables are 2 raised to the power of the corresponding lower variable, e.g. $D = 2^d$. We first review the definition of pairwise independent hash functions.

In this section we present one construction of the extractors by simply using hash functions. This construction is also called the *Leftover Hash Lemma* by Impagliazzo.

Definition 9.11 (pairwise independent hash functions)**.** *A family of pairwise hash functions is a set of functions $h : D \mapsto R$ such that for any distinct $x_1, x_2 \in D$ and all (not necessarily distinct) $y_1, y_2 \in R$, it holds that*

$$\mathbf{Pr}\left[\, h(x_1) = y_1 \wedge h(x_2) = y_2 \,\right] = \frac{1}{|R|^2}.$$

Theorem 9.12 (Leftover Hash Lemma,[1, 2])**.** *Let $\mathcal{H} = \{H \mid H : \{0,1\}^n \to \{0,1\}^m\}$ be a family of pairwise independent hash functions, where $m = k - 2\log(1/\varepsilon)$. Then $\mathsf{Ext}(X, H) = (H, H(X))$ is a (k, ε)-extractor.*

Proof. Let X be an arbitrary k-source on $\{0,1\}^n$ and d be the seed length of Ext. Choose H randomly from $\mathcal{H}$. We show that $(X, H(X))$ is ε-close to $\mathcal{U}_d \times \mathcal{U}_m$ in the following three steps:

- Step 1: We show that the collision probability of $(H, H(X))$ is close to that of $\mathcal{U}_d \times \mathcal{U}_m$.
- Step 2: We note that this is equivalent to saying that the ℓ_2-distance between $(H, H(X))$ and $\mathcal{U}_d \times \mathcal{U}_m$ is small.
- Step 3: Then we deduce that the statistical difference is small, by recalling that the statistical difference equals half of the ℓ_1 distance, which can be (loosely) bounded

by the ℓ_2 distance.

Step 1: By definition, the collision probability of $(H, H(X))$ is $\mathrm{CP}(H, H(X)) = \mathbf{Pr}\left[(H, H(X)) = (H', H'(X'))\right]$, where $(H', H'(X'))$ is independent of and identically distributed to $(H, H(X))$. Because $(H, H(X)) = (H', H'(X'))$ if and only if $H = H'$ and either $X = X'$ or $X \neq X'$ but $H(X) = H(X')$. Therefore

$$\begin{aligned}\mathrm{CP}(H, H(X)) &= \mathrm{CP}(H) \cdot \left(\mathrm{CP}(X) + \mathbf{Pr}\left[H(X) = H(X') | X \neq X'\right]\right) \\ &\leq \frac{1}{D} \cdot \left(\frac{1}{K} + \frac{1}{M}\right) \\ &= \frac{1+\varepsilon^2}{DM},\end{aligned}$$

where the last equality uses the fact that $m = k - 2\log(1/\varepsilon)$.

Step 2:

$$\begin{aligned}\|(H, H(X)) - \mathcal{U}_d \times \mathcal{U}_m\|^2 &= \mathrm{CP}(H, H(X)) - \mathrm{CP}(\mathcal{U}_d \times \mathcal{U}_m) \\ &\leq \frac{1+\varepsilon^2}{DM} - \frac{1}{DM} = \frac{\varepsilon^2}{DM}\end{aligned}$$

Step 3:

$$\begin{aligned}\Delta((H, H(X)), \mathcal{U}_d \times \mathcal{U}_m) &= \frac{1}{2} \cdot \|(H, H(X)) - \mathcal{U}_d \times \mathcal{U}_m\|_1 \\ &\leq \frac{\sqrt{DM}}{2} \cdot \|(H, H(X)) - \mathcal{U}_d \times \mathcal{U}_m\| \\ &\leq \frac{\sqrt{DM}}{2} \cdot \sqrt{\frac{\varepsilon^2}{DM}} \\ &= \frac{\varepsilon}{2}.\end{aligned}$$

Therefore, $\mathsf{Ext}(x, h(x))$ is a (k, ε)-extractor. □

9.3 Extractors versus Expanders

Extractors $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ can be considered as bipartite graphs $G = ([N], [M], E)$ where the nodes of the left side are strings of length n and the nodes of the right side are strings of length m. Every node x on the left side is connected to all 2^d nodes z for which there exists a $y \in \{0,1\}^d$ such that $\mathsf{Ext}(x, y) = z$. By the definition of extractors, for every set $S \subseteq \{0,1\}^n$ of size 2^k on the left hand side and for every set $T \subseteq \{0,1\}^m$ on the right hand side, the number of edges between S and T is close to what one expects in a random graph. More precisely, we have

$$\left|e(S, T) - |S| \cdot |T| \cdot 2^{d-m}\right| \leq \varepsilon,$$

where $e(S, T)$ is the number of edges between set S and T.

Remark 9.13. *Extractors correspond to* unbalanced *bipartite graphs, i.e., the sizes of the left- and right-hand sets are different. Moreover, when we view extractors as graphs, then the degree of every vertex from the left-hand side is not constant.*

From Extractors to Expanders. Let $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ be an extractor. Then, for any set $S \subseteq [N]$ of size K, we have that

$$\Delta(\mathsf{Ext}(U_S, \mathcal{U}_d), \mathcal{U}_m) \leq \varepsilon,$$

where U_S is the uniform distribution on S. Let $\mu(S) = |S|/N$ and $\mu(T) = |T|/M$. Then for any set $T \subseteq [M]$, it holds that

$$|\mathbf{Pr}\left[\,\mathsf{Ext}(U_S, \mathcal{U}_d) \in T\,\right] - \mu(T)| \leq \varepsilon,$$

i.e.,

$$\left|\frac{e(S,T)}{|S| \cdot D} - \mu(T)\right| \leq \varepsilon.$$

We rewrite the inequality above as

$$\left|\frac{e(S,T)}{ND} - \mu(S)\mu(T)\right| \leq \varepsilon\mu(S). \tag{9.1}$$

Proposition 9.14. *The function Ext is a (k,ε)-extractor if and only if the corresponding bipartite graph $G = ([N],[M],E)$ with left-degree D has the property that $\left|\frac{e(S,T)}{ND} - \mu(S)\mu(T)\right| \leq \varepsilon\mu(S)$ for every $S \subseteq [N]$ of size K and every $T \subseteq [M]$.*

From Expanders to Extractors. Comparing (9.1) with the Expander Mixing Lemma, which states that for any graph G with spectral expansion λ and for any sets $S, T \subseteq [N]$, we have

$$\left|\frac{e(S,T)}{N \cdot D} - \mu(S)\mu(T)\right| \leq \lambda\sqrt{\mu(S)\mu(T)},$$

it suffices that G is an extractor if $\lambda \cdot \sqrt{\mu(S)\mu(T)} \leq \varepsilon\mu(S)$ for all $S \subseteq [N]$ of size K and all $T \subseteq [N]$. So it suffices for $\lambda \leq \varepsilon\sqrt{K/N}$.

For the constructions of such expanders, we take an appropriate power of a constant degree expander. Specially, let G_0 be a D_0-regular expander on N vertices with bounded spectral expansion. We take the t-th power of G_0 and let $G = G_0^t$ where $t = O(\log((1/\varepsilon)\sqrt{N/K})) = O(n - k + \log(1/\varepsilon))$.

Theorem 9.15. *For every $n, k \in \mathbb{N}$ and $\varepsilon > 0$, there is an explicit (k,ε)-extractor $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^n$ with $d = O(n - k + \log(1/\varepsilon))$.*

Comparison. To conclude this lecture, we compare expanders and extracts. Although studying these two objects has different motivations and goals, expanders and extractors share a lot of similarity. Noticing that these two objects can be expressed by a function $f : \{0,1\}^n \times \{0,1\}^d \mapsto \{0,1\}^m$, constructions of expanders and extractors are to construct functions of the same form, aiming at optimizing different parameters.

Expanders	Extractors
Measured by vertex or spectral expansion	Measured by min-entropy/statistical difference
Typically constant degree	Typically logarithmic or poly-logarithmic degree
All sets of size at most K expand	All sets of size exactly (or at least) K expand
Typically balanced	Typically unbalanced, bipartite graphs

In addition to the close relationship between expanders and extractors, we can also relate extractors to other combinatorial objects. It is well understood that the following objects are very similar: pseudorandom generators, extractors, list-decodable error-correcting codes, and expander graphs. Constructions of any of these objects often give constructions of the others, see the survey by Vadhan [4].

References

[1] Charles H. Bennett, Gilles Brassard, and Jean-Marc Robert. Privacy amplification by public discussion. *SIAM J. Comput.*, 17(2):210–229, 1988.

[2] Russell Impagliazzo, Leonid A. Levin, and Michael Luby. Pseudo-random generation from one-way functions (extended abstracts). In *STOC*, pages 12–24, 1989.

[3] Noam Nisan and David Zuckerman. Randomness is linear in space. *J. Comput. Syst. Sci.*, 52(1):43–52, 1996.

[4] Salil P. Vadhan. The unified theory of pseudorandomness: guest column. *SIGACT News*, 38(3):39–54, 2007.