CAS CS 538. Lecture notes on Indistinguishability (September 12, 2019 lecture by Leo Reyzin)

The The Joy of Cryptography defines indistinguishability (Definition 4.5) by saying that the advantage of every polynomial-time algorithm in distinguishing is negligible. It's a good definition, but sometimes you want to be more precise. Precision is particularly important when you are deciding on setting parameters such as key lengths, because you have to choose a specific length in order to provide security against a specific class of adversaries.

So we will offer a more refined definition of indistinguishability, which takes into account the specific running time and distinguisher advantage. This saves us from having to worry about what "polynomial" and "negligible" mean. (We will, eventually, switch to just "polynomial" and "negligible" instead of keeping track of t and ϵ for every theorem we do, but I want us to build up some intuition first.)

Definition 1. We say $\mathcal{L}_{\mathsf{left}} \approx_{t,\epsilon} \mathcal{L}_{\mathsf{right}}$ if for all A whose running time[1] is at most t,

$$|\Pr[A \diamond \mathcal{L}_{\mathsf{left}} \Rightarrow 1] - \Pr[A \diamond \mathcal{L}_{\mathsf{right}} \Rightarrow 1]| \leq \epsilon\,.$$

This notion is called (t,ϵ)-indistinguishability. We can use in various security definitions (such as Definition 5.1 for PRG security or Definition 5.3 for one-time-security of encryption); then instead of saying simply "secure" (which means there is a negligible ϵ for all polynomial time t) you'll say (t,ϵ)-secure (which means a specific t and ϵ).

We can now make the Chaining Lemma (Lemma 4.7) more precise. (Note: I like to call it the post-processing lemma, because it says that post-processing of library outputs by a program $\mathcal{L}^*$ cannot help a distinguisher by any more than the running time of $\mathcal{L}^*$.)

Lemma 1. *If $\mathcal{L}_{left} \approx_{t,\epsilon} \mathcal{L}_{right}$ and $\mathcal{L}^*$ runs in time at most s, then $\mathcal{L}^* \diamond \mathcal{L}_{left} \approx_{t-s,\epsilon} \mathcal{L}^* \diamond \mathcal{L}_{right}$.*

Proof. Suppose some adversary A runs in time at most $t-s$. Let $A' = A \diamond \mathcal{L}^*$. Then A' runs in time at most $(t-s)+s = t$. Therefore, A' cannot distinguish $\mathcal{L}_{\mathsf{left}}$ from $\mathcal{L}_{\mathsf{right}}$ with advantage greater than ϵ. Thus,

$$\begin{aligned}
&|\Pr[A \diamond (\mathcal{L}^* \diamond \mathcal{L}_{\mathsf{left}}) \Rightarrow 1] - \Pr[A \diamond (\mathcal{L}^* \diamond \mathcal{L}_{\mathsf{right}}) \Rightarrow 1]| = \\
&|\Pr[(A \diamond \mathcal{L}^*) \diamond \mathcal{L}_{\mathsf{left}} \Rightarrow 1] - \Pr[(A \diamond \mathcal{L}^*) \diamond \mathcal{L}_{\mathsf{right}} \Rightarrow 1]| = \\
&\qquad |\Pr[A' \diamond \mathcal{L}_{\mathsf{left}} \Rightarrow 1] - \Pr[A' \diamond \mathcal{L}_{\mathsf{right}} \Rightarrow 1]| \leq \epsilon\,.
\end{aligned}$$

□

Even though this lemma seems trivial, it is extremely useful. It says that if you had and adversary who could distinguish $\mathcal{L}^* \diamond \mathcal{L}_{\mathsf{left}}$ from $\mathcal{L}^* \diamond \mathcal{L}_{\mathsf{right}}$, then you could have another adversary to distinguish $\mathcal{L}_{\mathsf{left}}$ from $\mathcal{L}_{\mathsf{right}}$ with just s additional time. $\mathcal{L}^*$ is called a *reduction*: it *reduces* the problem of indistinguishability of something more complicated (namely $\mathcal{L}^* \diamond \mathcal{L}_{\mathsf{left}}$ and $\mathcal{L}^* \diamond \mathcal{L}_{\mathsf{right}}$) to the problem of indistinguishability of something simpler (namely $\mathcal{L}_{\mathsf{left}}$ and $\mathcal{L}_{\mathsf{right}}$). Building reductions—that is, figuring the right $\mathcal{L}^*$ for your problem—is often the crucial step in proofs of indistinguishability.

There is one more very useful tool: transitivity of $\approx$ (also known as the "hybrid" argument). Again, we state a more precise version of it than the book does in Lemma 4.6.

[1] To cover adversaries that save on running simply because they precompute and hardwire a huge table, our notion of running time includes the time to load the program into memory—i.e., the length of the program

Lemma 2. *If* $\mathcal{L}_1 \approx_{t,\epsilon} \mathcal{L}_2$ *and* $\mathcal{L}_2 \approx_{s,\delta} \mathcal{L}_3$*, then* $\mathcal{L}_1 \approx_{\min(t,s),\epsilon+\delta} \mathcal{L}_3$*.*

Proof. Indeed, suppose A is an adversary whose running time is at most $\min(t, s)$. Then

$$\begin{aligned}
|\Pr[A \diamond \mathcal{L}_1 \Rightarrow 1] - \Pr[A \diamond \mathcal{L}_3 \Rightarrow 1]| \leq & \qquad (\text{because } |x+y| \leq |x| + |y|) \\
|\Pr[A \diamond \mathcal{L}_1 \Rightarrow 1] - \Pr[A \diamond \mathcal{L}_2 \Rightarrow 1]| + & \\
|\Pr[A \diamond \mathcal{L}_2 \Rightarrow 1] - \Pr[A \diamond \mathcal{L}_3 \Rightarrow 1]| \leq & \\
\epsilon + & \qquad (\text{because } \mathcal{L}_1 \approx_{t,\epsilon} \mathcal{L}_2) \\
\delta\,. & \qquad (\text{because } \mathcal{L}_2 \approx_{s,\delta} \mathcal{L}_3)
\end{aligned}$$

□

This lemma, like the previous, may seem trivial, but is also very useful. Often the trick to proving indistinguishability of two libraries (such as $\mathcal{L}_1$ and $\mathcal{L}_3$) is to find one (or more) intermediate library (such as $\mathcal{L}_2$) and then prove the whole sequence to be indistinguishable in pairs. The difficulty is in finding the correct $\mathcal{L}_2$ (or sequence of such libraries). It often combines some features of $\mathcal{L}_1$ and $\mathcal{L}_3$ and is therefore called a *hybrid*. The proof technique is called a *hybrid argument*.

An example of how both lemmas can be combined to give a non-trivial statement is the proof that pseudorandom one-time-pads give computational one-time security, per Claim 5.4. The more precise version of Claim 5.4 (i.e., with t and ϵ worked out) is the following.

Claim 1. *Let* pOTP *denote Construction 5.2. If, for some* t *and* ϵ*,* pOTP *is instantiated using a* (t, ϵ) *secure PRG* G*, then* pOTP *is* $(t - s, 2\epsilon)$*-secure, where* s *is the time required to compute* $\lambda + \ell$ *XOR operations (in big-O notation,* $s = \Theta(\lambda)$*).*

The proof of the claim is the same as the proof of Claim 5.4: you use the hybrid argument Lemma 4.6 (Lemma 2 here) seven times. Five of those uses uses are just interchangeability (i.e., $\equiv$). The other two (from $\mathcal{L}_{\mathsf{hyb\text{-}2}}$ to $\mathcal{L}_{\mathsf{hyb\text{-}3}}$ and from $\mathcal{L}_{\mathsf{hyb\text{-}5}}$ to $\mathcal{L}_{\mathsf{hyb\text{-}6}}$) are indistinguishability (i.e., $\approx$) steps, and each requires the chaining Lemma 4.7 (Lemma 1 here) to prove it. The chaining lemma results in the s term being subtracted from t, which means each of these two steps is $(t - s, \epsilon)$-indistinguishable, for a total of $(t - s, 2\epsilon)$.