Computer Science 2426F | Fall, 2020
St. George Campus | University of Toronto

Notes #1.5
Removing Randomness From Nonuniform Adversaries

Probabilistic Nonuniform Adversaries

The goal of this note is to prove the exercise from the bottom of Page 1 of Notes #1. We want to show that for adversaries against pseudo-randomness, nonuniform adversaries that use randomness are no more powerful than nonuniform adversaries that are deterministic – that is, that do not use randomness. (A similar theorem will be true about nonuniform adversaries in other settings.)

So let G be a number generator, where $|G(s)| = l(|s|)$. What do we mean by a nonuniform adversary that uses randomness? We mean a family $D = \{D_1, D_2, \dots\}$ of circuits; D_n has $l(n)$ input bits and one output bit; for some c and sufficiently large n, D_n has size $\leq n^c$. In addition to the usual gates, D_n is allowed to use *coin-tossing* gates, where a coin-tossing gate has no inputs and chooses its output bit *randomly* whenever the circuit is run. $p_D(n)$ and $r_D(n)$ now mean the obvious things. For example, to define $p_D(n)$ we consider the following experiment:
Choose a random $n-$bit string s; compute $G(s)$; run D_n on $G(s)$, choosing the outputs of the coin-tossing gates randomly.
Then $p_D(n)$ is the probability that D accepts (that is, outputs 1).

Say (w.l.o.g) that $p_D(n) - r_D(n) > 0$. We wish to show that there is a *deterministic* circuit D' that is no bigger than D, such that $p_{D'}(n) - r_{D'}(n) \geq p_D(n) - r_D(n)$. We will do this by fixing the outputs of the coin-tossing gates appropriately.

Say that D_n has m coin-tossing gates. For each $m-$bit string u, define $p_D(n, u)$ to be the probability that D accepts in the above experiment when the coin-tossing gates are fixed to output u (that is, the first coin-tossing gate always outputs the first bit of u, the second one always outputs the second bit of u, etc.). Define $r_D(n, u)$ similarly. We now have

$$p_D(n) = E_u(p_D(n, u)) \qquad \text{and} \qquad r_D(n) = E_u(r_D(n, u))$$

where $E_u(\alpha)$ is the expected (or average) value of α as u varies randomly over $m-$bit strings. We therefore have (by the additivity of expectations)

$$p_D(n) - r_D(n) = E_u(p_D(n, u) - r_D(n, u))$$

So there must be some u – call it u_0 – such that

$$p_D(n, u_0) - r_D(n, u_0) \geq E_u(p_D(n, u) - r_D(n, u)) = p_D(n) - r_D(n)$$

We now form the deterministic circuit circuit D' by fixing the output wires of the coin-tossing gates to be u_0. We have

$$p_{D'}(n) - r_{D'}(n) = p_D(n, u_0) - r_D(n, u_0) \geq p_D(n) - r_D(n)$$

How can we find an appropriate u_0. In fact, we have no efficient, deterministic way to do this. The whole point of nonuniformity is that we don't *have* to have a way of finding u_0; u_0 is hardwired into the circuit.