6.3 Conversion Tricks

We give heuristic reasons that the following statements (A) to (D) are equivalent, and that each of them implies (E)—for a formal mathematical proof we don't have yet the exact definitions.

These implications also have practical relevance for constructing a basic function given another one. A coarse summary—for the discussion on regulations of cryptography that pop up from time to time—consists of the statements

- Who wants to prohibit encryption also must prohibit hash functions and pseudo-random generators.
- Who wants to make cryptography impossible must prove that $\mathbf{P} = \mathbf{NP}$.

(A) There is a one-way function $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$.

($\tilde{\mathbf{A}}$) There is a one-way function $\tilde{f}\colon \mathbb{F}_2^{2n} \longrightarrow \mathbb{F}_2^n$.

(B) There is a weak hash function $h\colon \mathbb{F}_2^* \longrightarrow \mathbb{F}_2^n$.

(C) There is a strong symmetric cipher $F\colon \mathbb{F}_2^n \times \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$ (where "strong" means secure under a known-plaintext attack).

(D) There is a perfect pseudo-random generator $\sigma\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^{p(n)}$.

(E) $\mathbf{P} \neq \mathbf{NP}$.

Remark 1 Making the statements precise in terms of complexity theory we have to state (A) – (D) for families of functions that are parametrized by n.

Remark 2 A pseudo-random generator is perfect if for unknown $x \in \mathbb{F}_2^n$, given some bits of the output $\sigma(x)$, there is no efficient way to predict some more bits of the output, or to compute x. In the specification p is a polynomial with integer coefficients—from a "seed" of length n the generator produces $p(n)$ bits.

We omit reasoning about the implication "(D) $\Longrightarrow$ (E)".

"(C) $\Longrightarrow$ (D)": Set $\sigma(x) = (s_1, \ldots, s_{p(n)/n})$ with $s_0 := x$ and $s_i := F(s_{i-1}, z)$ for $i \geq 1$, where the key z is a secret constant parameter. Note the similarity with the OFB mode for bitblock ciphers. For no block s_i of the sequence the attacker is able to determine the previous block s_{i-1}—otherwise the cipher wouldn't be secure. It is not obvious that this property suffices to show perfectness, we'll show this in Chapter IV.

"(D) $\Longrightarrow$ (C)": Consider the bitstream cipher that uses $\sigma(x)$ as bitstream and x as key.

"(A) $\Longrightarrow$ (C)": There is a simple approach by E. BACKUS: Set $F(a,k) = a + f(k)$. Under a known-plaintext attack a and $c = F(a,k)$ are known. Hence also $f(k) = c - a$ is known. So the attack reduces to inverting f.

[Other approaches: MDC (= Message Digest Cryptography) by P. GUTMANN, or the FEISTEL scheme.]

"(C) $\Longrightarrow$ (A)": See the example in Section 6.1.

"(A) $\Longrightarrow$ ($\tilde{\text{A}}$)": Define $\tilde{f}$ by $\tilde{f}(x,y) := f(x+y)$. Assume we can compute a pre-image (x,y) of c for $\tilde{f}$. Then this gives also the pre-image $x+y$ of c for f.

"($\tilde{\text{A}}$) $\Longrightarrow$ (B)": Pad $x \in \mathbb{F}_2^*$ with (at most $n-1$) zeroes, giving $(x_1, \ldots, x_r) \in (\mathbb{F}_2^n)^r$. Then set

$$\begin{aligned} c_0 &:= 0, \\ c_i &:= \tilde{f}(c_{i-1}, x_i) \quad \text{for } 1 \leq i \leq r, \\ h(x) &:= c_r. \end{aligned}$$

This defines $h\colon \mathbb{F}_2^* \longrightarrow \mathbb{F}_2^n$.

Let $y \in \mathbb{F}_2^n$ be given. Assume the attacker finds a pre-image $x \in (\mathbb{F}_2^n)^r$ with $h(x) = y$. Then she also finds a $z \in (\mathbb{F}_2^n)^2$ with $\tilde{f}(z) = y$, namely $z = (c_{r-1}, x_r)$ (where $y = c_r$ in the construction of h).

"(B) $\Longrightarrow$ (A)": Restricting h to $\mathbb{F}_2^n$ also gives a one-way function.