Pseudo Random Bit Generators

Prof. Zeph Grunschlag
Pseudo Random Generator

PRG’s also known as stream ciphers because they correspond to pseudo-random one-time pads. Intuitively, these are deterministic functions whose outputs cannot be differentiated from random bitstreams.
PRG Definition

DEF: $U_k$ denotes the uniform distribution on bitstrings of length $k$.

NOTE: Security is built-in following definition:

DEF: A PRG with expansion $l(k)$ is a deterministic poly-time algorithm $g$ from bitstrings to bitstrings s.t.:

- $l(k)$ is a polynomial in $k$ s.t. $l(k) > k$
- $|g(x)| = l(|x|)$
- No PPT distinguisher $D$ exists with $\text{Prob}(D(g(U_k)) = 1) - \text{Prob}(D(U_{l(k)}) = 1)$ non-negligible in terms of $k$. 
Blum-Blum-Shub
Official PRG

- \( l(k) \) is any polynomial > \( k \)

INPUT: random seed \( x \) of length \( k \)

OUTPUT: bitstring \( s \) of length \( L \)

Use 1st \( \frac{1}{4} \) of \( x \) to generate \( p \) deterministically

Use 2nd \( \frac{1}{4} \) of \( x \) to generate \( q \) deterministically

Let \( n = p \cdot q \), and \( r = 2\text{nd } \frac{1}{2} \) of \( x \).

Return \( \text{BBS-PRG}(n, r, l(k)) \)  // slide #5 from

// “probabilistic encryption”
PRG ⇔ Stateful Private Encryption

THM: A pseudo random bit generator exists iff a stateful symmetric encryption scheme exists with $|M| > |K|$ that is computationally secure.

½ proof: PRG $g \Rightarrow$ Encryption $E_K$: Use the pseudo random one time pad defined by

- security parameter $k$ chosen so $l(k) \geq |m|
- G: K = U_k$ (key $K$ a rand. $k$-bit string)
- $E_K(m) = g(K) \oplus m$
Construction of PRG’s

THM: Suppose there is a one way permutation, then there is a PRG with arbitrary polynomial expansion.

Need the following ideas:
• one way function
• one way permutation
• hard core bit