PSEUDORANDOMNESS
AN OVERVIEW
by
Oded GOLDREICH
Weizmann Inst.

http://www.weizmann.ac.il/~oded/PS/prg-n06.ps
RANDOMNESS & COMPUTATION

• RANDOMNESS as a TOOL used in COMPUTATION

  Essential uses include
  + CRYPTOGRAPHY & DISTRIBUTED COMPUTING
  + Prob. Proof Systems (IP, ZK, PCP)
  + Sampling & Property Testing
  (Omitted: Use in standard ALGORITHMS)

• RANDOMNESS as an OBJECT viewed by COMPUTATION

  ⇒ Computational Indistinguishability

  ⇒ Different objects viewed as equiv. by resource-bounded computations.

  ⇒ Potential saving/elimination of RANDOMNESS in COMPUT. (because corresp. apply: cannot tell...)
Computational View of Randomness

**Computational Indistinguishability**

\[
X \equiv Y \quad \Leftrightarrow \quad \sum_x |\text{Prob}[X_n = x] - \text{Prob}[Y_n = x]| \\
\text{is negl}(n)
\]

\[
\Rightarrow \quad X \equiv Y \equiv \text{Efficient algorithms (and/or ALG of certain class)} \quad \text{"can not tell these apart"}
\]

Classes to consider:

- Poly-time alg.
  - Poly-size circuits (non-uniform)
- Quad-size circuits
- Space-bounded alg.
- Syn. restricted alg.
  - (projection, linear, hitting)
**Computational Indistinguishability**

\[ Z = \{ \Xi_k \} \text{, where } \Xi_k \in \{0,1\}^k \text{ or } \{0,1\}^{|k|} \]

**Def:** \( X \) and \( Y \) are \( \epsilon \)-indistinguishable by \( D \)

\[
\begin{align*}
D(X_k) & \quad \text{by D} \\
D(Y_k) & \quad \text{potential distinguished}
\end{align*}
\]

D's verdict is insignificant

\[
| \text{Prob}[D(X_k) = 1] - \text{Prob}[D(Y_k) = 1] | \leq \epsilon(k)
\]

Typically, \( \epsilon \) is negligible

\[
= \frac{1}{\text{complexity}(D)}
\]

When class of ALG is understood, we say that \( X \& Y \) are comput. indisting.
Notions of Pseudorandom Generators

\[ G: \Sigma_0^{13k} \rightarrow \Sigma_0^{13} \rho(k) \] is a PRG (generic) if

1. Stretch \( \rho(k) > k \) (\( \rho(k) \gg k \))
   - \( \rho \) is a polynomial
   - \( \rho \) is an exponential \( [\rho(k) = 2^{\Theta(k)}] \)

2. Efficient Generation
   - Each bit produced in POLY-TIME
   - Each bit produced in EXP-TIME

3. Pseudorandomness \( \equiv \) Computational Indist. from the Uniform (i.e. \{1,0\}^k)
   - By (Prob.) POLY-TIME Alg.
   - By QUAD-SIZE Circuits.

General Purpose PRG

Canonical Derandomizer

Gen more complex than D
Two popular notions of PRG

• **General-Purpose PRG**
  Can be used to save randomness in **any** (efficient) application.
  Output looks random also to observer that uses more resources than the PRG.

• **Canonical Derandomizer**
  May (and typically does) use more resource than the observer.
  Sufficient for derandomization of ALGs of specific complexity.

*) Essential to crypto/adversary applications.*
AMPLIFYING THE STRETCH OF GENERAL-PURPOSE PRGs

Suppose \( G : \{0,1\}^k \rightarrow \{0,1\}^{k+1} \) is a PRG, and \( p : \mathbb{N} \rightarrow \mathbb{N} \) is a polynomial (s.t. \( p(k) > k \)).

**Naive Iteration Method**

\[
G'(s) \equiv G^{p(151)-151}(s), \text{ where } G^i(x) \equiv G(G^{i-1}(x)) \text{ and } G^0(x) \equiv x
\]

```
G = 0
G = 0
```

\( \text{"Detected: sign detected" at this move.} \)

\( \text{Distinguish top line by applying } G \)

**Fixed-Length Iteration Method**

\[
G'(s) \equiv G_1 \cdot G_2 \cdots \cdot G_{p(151)}, \text{ where } s_0 \equiv s \text{ and } G_i s_i \equiv G(s_{i-1})
\]

```
G_1
```

\( \text{Distinguish top line by applying } G \)
**Def:** \( f: \{0,1\}^k \rightarrow \{0,1\}^k \) is a OWF if
1. poly-time computable
2. hard to invert on average-case
   \[ \forall \text{ppt } A, \quad \Pr_{x \leftarrow \{0,1\}^k} [A(f(x)) = f^{-1}(x)] = \text{negl}(k) \]

**Thm:** PRG exist iff OWF exist.

**PRG \rightarrow OWF:**
\[ G: \{0,1\}^k \rightarrow \{0,1\}^{2k} \text{ PRG} \]
\[ f(x,y) = G(x) \text{ for } |x| = |y| \]

Inverting \( f \) on \( f(u_k) = G(u_k) \)
implies
\[ \text{dist. } G(u_k) \text{ from } U_{2k} \]
(since the latter has \( f \)-preimage w. negl. prob).

**OWF \rightarrow PRG:**
We'll only show a special case.
OWF implies PRG

**DEF:** $b : \{0,1\}^* \rightarrow \{0,1\}^n$ is a **Hardcore** of $f$ if

1. $x \rightarrow b(x)$ is poly-time computable
2. $f(x) \rightarrow b(x)$ is hard to predict on average-case:
   \[ \forall \text{ppt } A, \quad \Pr_x[A(f(x)) = b(x)] < \frac{1}{2} + \text{negl}(n) \]

**Note:** $b(x)$ hard to predict from $f(x)$

- $f(x) \cdot b(x)$ is same
- $f(x) \cdot b(x)$ is independent

- Indivi. bits may not be hardcore;
  e.g., $f(x,y) = (f'(x), y)$
- If $f$ is 1-1 & easy to invert then it has no hardcore.

For a 1-1 OWF $f$, any hardcore $b$ yields a PRG $G(s) = f(s) \cdot b(s)$. 

**Amplifying stretch**
**OWF \rightarrow HARDCORE**

\[ \text{OWF } I \quad \Rightarrow \quad \text{OWF } \quad f(x, r) = (b(x), r) \]

\[ b(x, r) = \sum_{i=1}^{k} x_i \cdot r_i \mod 2 \]

**Lemma:**
Suppose, given \( B : \{0,1\}^k \rightarrow \{0,1\} \) s.t. \( \exists x \in \{0,1\}^k \)

\[ \text{Prob}_{r \in \{0,1\}^k} [ B(r) = b(x, r) ] > \frac{1}{2} + \varepsilon \]

Then, in \( \text{poly}(k/\varepsilon) \)-time, can guess \( x \) correctly w.p. \( \geq \text{poly}(\varepsilon/k) \)

\[ B_x(r) = A(f(x), r) \]

**Warm-up:**
Suppose \( P_x = \text{Prob} [ B(r) = b(x, r) ] > \frac{3}{4} + \varepsilon \)

\[ \Rightarrow \text{Recall } x_j \text{ w.p. } 1 - 2 \cdot (1 - P_x) > \frac{1}{2} + 2\varepsilon \]

by \( r \in \{0,1\}^k \) output \( B(r) \oplus B(\text{r} \oplus \text{e}^j) \)

\[ [ b(x, r) \oplus b(x, \text{r} \oplus \text{e}^j) ] = (\sum_{i=1}^{k} x_i \cdot r_i) + (x_j + \frac{1}{2} \cdot x_j \cdot r_j) = x_j \]

**Eliminate error-doubling**
Suppose \( r^{(1)}, \ldots, r^{(m)} \in \{0,1\}^k \) pairwise inder and we know \( b(x, r^{(i)}), \ldots, b(x, r^{(m)}), \)

Then \( \text{MAJ}_{i \in [m]} [ \text{b}(x, r^{(i)}) \oplus B(r^{(i)} \oplus \text{e}^j) ] = x_j \)

with prob. \( \geq 1 - \frac{1}{2} \varepsilon \)

\[ \text{[simple call to } B, \text{per "vote"}] \]

**How?**
Generating PAIRWISE IND. samples in \( \mathbb{Z}_3^{13} \)
with known \( b(x, \cdot) \)-values

Select \( s^{(1)}, \ldots, s^{(m)} \in \mathbb{Z}_3^{13} \), where \( l = \log_2 (m+1) \)

Guess \( b(x, s^{(1)}), \ldots, b(x, s^{(m)}) \in \mathbb{Z}_3^{13} \)

\[ \text{[correct w.p.]} \quad 2^{-l} = \frac{1}{m+1} = \frac{1}{\text{poly}(k_k)} \]

Generate \( \langle \mathbb{Z}^{(1)} \rangle \) \( \forall I \) s.t. \( \mathbb{Z}^{(1)} = \bigoplus_{i \in I} s^{(i)} \)

and note

that

\[ b(x, \mathbb{Z}^{(1)}) = b(x, \bigoplus_{i \in I} s^{(i)}) \]

\[ = \bigoplus_{i \in I} b(x, s^{(i)}) \]

Thus, w.p. \( \frac{1}{m+1} \), we obtain the correct values for all \( b(x, \mathbb{Z}^{(1)}) \)'s.

**Note:** \( \mathbb{Z}^{(1)} \)'s are PAIRWISE (ND)
and uniformly dist. in \( \mathbb{Z}_3^{13} \).

**Hardness vs. Randomness, Act 2**

$G: \{0,1\}^k \rightarrow \{0,1\}^{p(k)}$ is a **canonical derandomizer** if $G$ is Exp-Time computable.

**Derandomization of $A$, where $A(x, 1^k)$ with**

$|x| = t_A(x) = poly(|1^k|)$

- $A'(x, s) = A(x, G(s))$
  - where $|s| = l^{-1}(t_A(x)) = k$

- $A''(x) = \max_{s \in 01^k} A'(x, s)^2$

**Running time**

$2^k \cdot (t_A(x) + t_G(k))$

**THM:** If $E$ can demand with $\ell(k) = 2^{2^k}$

then $BPP = P$. 

$k = l^{-1}(\text{poly}(n)) = O(\log n)$

$\Rightarrow 2^k = \text{poly}(n)$

**THM:** If $E = \text{Dtime}(2^{o(n)})$ contains a problem of circuit complexity $2^{2^{o(n)}}$ [in worst-case sense]

then $E$ can demand with $\ell(k) = 2^{2^k}$

$\exists c > 0 \text{ s.t. } \text{Dtime}(2^n) \not\subseteq \text{Size}(2^{c \cdot n})$
Constructing a CANON: DERANDOMIZER

Omitted: Worst-case Hardness \Rightarrow Average-case Hardness

\[ \exists f \in \mathcal{E} \text{ s.t. } \forall 2^{\Omega(n)} \text{-size circuit } C_n \]
\[ \Pr_{x \in \{0,1\}^n} [C_n(x) = f(x)] < \frac{1}{2} + 2^{-\Omega(n)} \]

\text{Constr.}

\[ G(s) = \prod_{I \in \{I_1, I_2, \ldots, I_{\ell(k)}\}} f(s|I) \quad \text{where} \]

- Compute \( I_1, I_2, \ldots, I_{\ell(k)} \)
- Evaluate \( f \) on \( f(x) \) points

\[ \text{time } \approx 2^{\Omega(n)} \Rightarrow f(k) = 2^{\Theta(k)} \approx \text{circuit-size}^{1/2} \]
\[ = \exp(o(k)) \]

Pseudorandomness \iff Unpredictability

\Rightarrow \text{Obvious (small uniform is unpredict.)}

see next 2 use this!

\text{Warm-up: Suppose } (I_j)\text{'s are disjoint.}

\text{Intuition to real case:}

Small intersections "bound" the gain
from \[ f(s|I_1) \ldots f(s|I_3) \]
towards predicting \[ f(s|I_{\ell+1}) \]
Suppose \( \{ Z_k \} \) is not pseudorandom; i.e.,\[ \exists k \text{ s.t. } Z_k \neq \chi \cdot U_{\ell(k)} \]

Consider HYBRIDS

\[ H_{k}^{(i)} = \underleftarrow{i} \underrightarrow{Z_k} \text{ RANDOM} \]

\[ H_{k}^{(i)} = U_{\ell(k)} \quad \text{and} \quad H_{k}^{(\ell(k))} = Z_k \]

\[ H_{k}^{(2)} = \underleftarrow{i} \quad \text{and} \quad H_{k}^{(3)} = \underleftarrow{i+1} \]

\[ H_{k} = \underleftarrow{i} \quad \text{and} \quad H_{k}^{(4)} = \underleftarrow{i+1} \]

Can emulate this...

Can distinguish 1st bit from a random value (when given \( i \)-prefix) of \( Z_k \)

Down arrow
**PRGs for Space-Bounded Distinguishers**

**THM:** Every Prob. Poly-Time algorithm can be emulated by a PPT algorithm of $\text{RANDOMNESS} = O(|\text{input}| + \text{original space complex})$.

**Conj:** Similar with $\text{RANDOMNESS} = O(\log |\text{input}| + \text{original space complex})$.

**Support**

1. A PRG with $|\text{seed}| = (\text{space complex})^2$
2. $\text{BPL} \subseteq \text{SC} \subseteq \text{TiSp}(\text{poly}, \text{polylog})$
3. $\text{UConn} \subseteq \text{L} \ [2005]$

$\text{RL} \ [1979]$

$(2') \text{BPL} \subseteq \text{DSPACE}((\log)^{1.5})$
SPECIAL-PURPOSE PRGs

**Projection Tests** ⇒ $t$-wise independent PRG

$G(s_0, \ldots, s_{t-1}) = \left( \frac{t-1}{j=0} s_j \cdot x_i^j \right)_{i=1, \ldots, d(k)}$

$s_0, s_{t-1}, x_i, \ldots$ are field elements

**Applications**

**Linear Tests** ⇒ Small-bias PRG

$G(s, f) = \text{LFSR}_f(s)$

**Feedback Rule**

START SEQ.

Application: "PCP of linear system" (dich. and quad. sys.)

```
  x x x = x
  x x x = x x x
  x x x = x
  x x x = x
```

(few) Linear combinations of the rows

**Hitting Tests** ⇒ Expander Walk PRG

$G(s, \tilde{v}_1, \ldots, \tilde{v}_{t-1}) = (v_0, \tilde{v}_1, \ldots, \tilde{v}_{t-1})$

∀ $s \subseteq \text{Vertex Set}$

of density $\geq \frac{1}{2}$

$\Pr[\text{sequence does not hit the set } s] < 2^{-\Omega(d)}$
Some Credits

Comput. Indist. \( \sim \) [Goldwasser+Micali]+[Yao]

Gen. Pur. PRG \( \sim \) [Blum+Micali]+[Yao]

Construction of gen.pur. PRG
- hardcore + iterations [BM]
- hardcore for any OWF [Goldreich+Levin]
- The Char. THM. [Hastad, Impagliazzo, Levin+Luby]

Canonical Derandomizers [Nisan+Wigderson]
\( E \neq \text{Size}(2^{\mathsf{O}(n)}) \Rightarrow \text{BPP}=\text{P} \) [Impag. + Wigderson]

PRG for space-bounded disting. [Nisan+Zuckerman], [Nisan], [Reingold]

Special-Purpose PRGs
- k-nise [Chor+Goldreich]+[Alon, Babai+Itai]
- small-bias [Naor\(^2\)]+[Alon, Goldreich, Hastad] +Peralta
- Expander Walk [Ajtai, Komlos+Szemeredi]

More details/material @
http://www.weizmann.ac.il/~oded
/pp_pseudo.html