Exposure-resilient Extractors and Applications

Marius Zimand
Towson University

Facultatea de Matematica Bucuresti
Feb. 2009

A simple example for randomness extraction: von Neumann problem

- Source of randomness: a biased coin.
- Prob[coin = H] = p, Prob[coin = T] = (1-p), p unknown.
- Independent tosses: TT HH HT HT HH TH H...
- Can we get unbiased bits?

A simple example for randomness extraction: von Neumann problem

- Source of randomness: a biased coin.
- Prob[coin = H] = p, Prob[coin = T] = (1-p), p unknown.
- Independent tosses: TT HH HT HT HH TH H...
- Can we get unbiased bits?
- Yes. Prob[HT] = p(1-p), Prob[TH] = (1-p)p. So make HT $\rightarrow$ 0, TH $\rightarrow$ 1.
- So we get 001...

- Many times we need good random bits: cryptography, algorithms, simulation, ...
- Typically, the sources of randomness are not perfect: biases, correlations.
- Extractors improve the quality of randomness of a source.
- Pseudo-random generators handle a different problem: given a few random bits (the seed), produce a longer random string that "looks" random.
- Extractors and pseudo-random generators solve quite different problems; there are however surprising connections.

What's in this talk

- Extractors
- Exposure-resilient extractors introduced in **[ZIM'06]**
- A construction of exposure-resilient extractors based on the Håstad-Impagliazzo-Levin-Luby construction of a pseudo-random generator from a one-way function **[ZIM'06]**
- Applications in derandomization **[ZIM'07]**

A short history of randomness extractors

- 50's - Von Neumann: How to get unbiased bits from a biased coin.
- 80's: generalization to distributions were bits may have Markov-type correlations.
- 90's - present: theory of extractors that handle general imperfect distributions.

- **Extractor** = procedure that transforms imperfect randomness into (almost) perfect randomness (information-theoretic randomness).

- **Extractor** = procedure that transforms imperfect randomness into (almost) perfect randomness (information-theoretic randomness).
- **Exposure-resilient Extractor** = the above + the output looks random even to computationally unbounded adversaries that have adaptive but bounded access to the input.

Min-entropy; a way to assess the quality of randomness

Perfect Randomness = Unif. distribution

n-bit strings

- each string is equally likely to be produced
- for all x,
  $\Pr(X = x) = 2^{-n}$

Imperfect randomness = min-entropy $< n$

n-bit strings

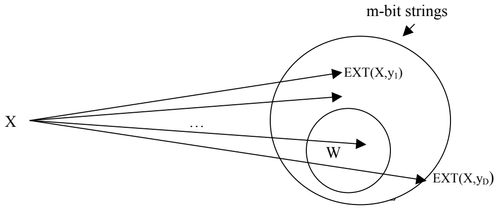
- some strings are more likely than other strings
- for all x,
  $\Pr(X = x) \leq 2^{-k} \overset{def}{\Leftrightarrow} \min - \mathrm{entropy}(X) = k$

(standard) seeded extractor

- $\mathrm{EXT} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$
- $\mathrm{EXT}(X, Y)$
  - X - "weakly random string,", min-entropy$(X) = k$
  - Y - "seed," unif. distributed
- For any $W \subseteq \{0,1\}^m$,

$$\mathrm{Prob}_{X,Y}(\mathrm{EXT}(X,Y) \in W) = \frac{|W|}{2^m} \pm \epsilon.$$

(Standard) Extractor



Another view

- Let's view W as an adversary – computationally unbounded.
- Adversary is given the challenge Z which is either
  (1) $\mathrm{EXT}(x, y)$, OR
  (2) U_m
- Adv. wants to distinguish (1) from (2) with bias ϵ. EXT is a (k, ϵ)-extractor if no adv. succeeds.

- Can we consider a more powerful adversary?
- Yes. Give him more information.

Strong extractor

- Adversary is given the seed y and the challenge Z which is either
  (1) $\mathrm{EXT}(x, y)$, OR
  (2) U_m
- Adv. wants to distinguish (1) from (2) with bias ϵ. EXT is a (k, ϵ)-strong extractor if no adv. succeeds.

Lu extractor

- ROUND 1: Adversary is given x and is allowed to calculate $f(x)$ with $|f(x)| = q$ bits.
- ROUND 2: Adversary loses access to x and is given the seed y and the challenge Z which is either
  (1) $\mathrm{EXT}(x, y)$, OR
  (2) U_m
- Adv. wants to distinguish (1) from (2) with bias ϵ. EXT is a (k, ϵ)-Lu extractor resistant to storage size q if no adv. succeeds.

Exposure-resilient extractor

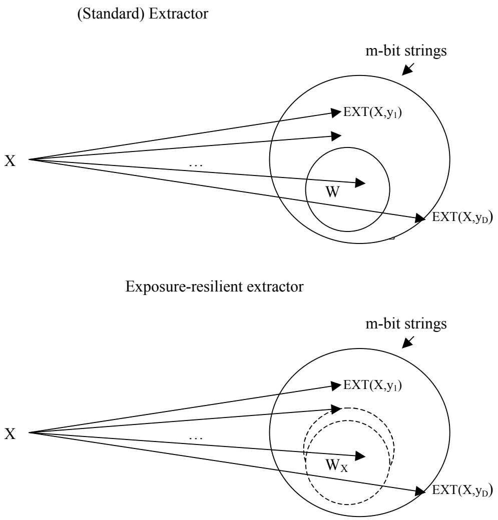
- Adversary is given the challenge Z which is either
  (1) $\mathrm{EXT}(x, y)$, OR
  (2) U_m

  and *simultaneously* oracle access to x to which it is allowed to make q queries.
- Adv. wants to distinguish (1) from (2) with bias ϵ.
- EXT is a (k, ϵ)-exposure-resilient extractor resistant to q queries if no adv. succeeds.

Formal definition

- adaptive test = oracle circuit
- $\mathrm{EXT} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ is a (k, ϵ)-extractor with query resistance q if for every distribution X with min-entropy k and for every adaptive test W with query complexity q

$$\mathrm{Prob}_{X,y}(\mathrm{EXT}(X,y) \in W^X) = \mathrm{Prob}_{X,U_m}(U_m \in W^X) \pm \epsilon.$$


(Standard) Extractor
m-bit strings
EXT(X,y1)
X
…
W
EXT(X,yD)
Exposure-resilient extractor
m-bit strings
EXT(X,y1)
X
…
WX
EXT(X,yD)

Motivation

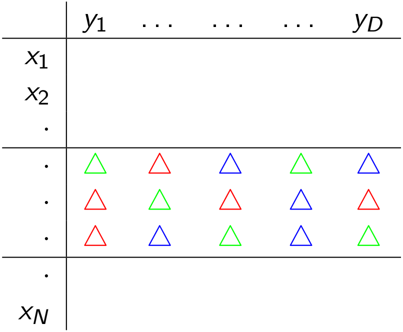
- Natural concept
- Exposure-resiliency is an important issue in cryptography.
- Sampling functions that have some dependency on the randomness of the sampler.
- Derandomization (later).

How to build an exposure-resilient extractor

An extractor $\mathrm{EXT} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ has nice combinatorial properties.

Using EXT, we color the $[N] \times [D]$ rectangle with colors from $[M]$.

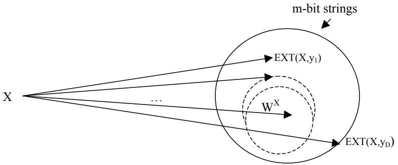
If in each each strip of height $\geq 2^k$ each color $c \in [M]$ appears a fraction of $(1 \pm \epsilon)/M$ times, then E is (k, ϵ)-extractor.



How to build an exposure-resilient extractor

- No similar property for exposure-resilient extractors.
- Techniques based on error-correcting codes, polynomials, designs, etc., are not enough.
- Use a reduction based on the HILL construction of a PRG from a one-way function.

A simple and useful lemma

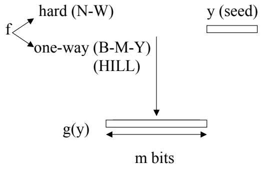


DEF. X hits W^X ϵ-correctly if

$$\frac{|\{y \mid \mathrm{EXT}(X,y) \in W^X\}|}{D} = \frac{|W^X|}{2^m} \pm \epsilon.$$

Lemma. Let $\mathrm{EXT} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$. If $\forall W$, no. of X that do not hit W ϵ-correctly is $\leq 2^t$
$\Rightarrow$ EXT is a $(t + \log(1/\epsilon), 2\epsilon)$-extractor

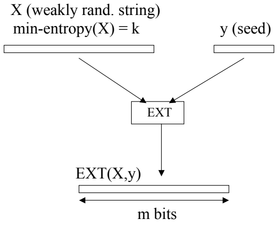
Pseudo-rand. generator



for all $A \subseteq \{0,1\}^m$, computable by poly-size circuits,

$$\Pr(g(y) \in A) = \frac{|A|}{2^m} \pm \epsilon.$$

(k, ϵ)-Extractor



for all A with query complexity q,

$$\Pr(\mathrm{EXT}(X, y) \in A) = \frac{|A|}{2^m} \pm \epsilon$$

Trevisan's method

$$\underset{\text{hard-function}}{f} \quad \xrightarrow{\text{N}-\text{W}} \quad \underset{\text{p.r.gen.}}{g_f}$$

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

Trevisan's method

$$\underset{\text{hard-function}}{f} \xrightarrow{\text{N}-\text{W}} \underset{\text{p.r.gen.}}{g_f}$$

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

- View the weakly random string as the truth-table of a function f

Trevisan's method

$$\begin{array}{ccc} f & \xrightarrow{\mathrm{N-W}} & g_f \\ \text{hard-function} & & \text{p.r.gen.} \end{array}$$

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

- View the weakly random string as the truth-table of a function f
- $\mathrm{EXT}(f, y) = g_f(y)$

Trevisan's method

$$\begin{array}{ccc} f & \xrightarrow{\text{N}-\text{W}} & g_f \\ \text{hard-function} & & \text{p.r.gen.} \end{array}$$

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

- View the weakly random string as the truth-table of a function f
- $\mathrm{EXT}(f, y) = g_f(y)$
- f does not hit D ϵ-correctly $\Leftrightarrow$ D disting. f from unif.

Trevisan's method

$$f \xrightarrow{\text{N-W}} g_f$$

hard-function p.r.gen.

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

- View the weakly random string as the truth-table of a function f
- $\mathrm{EXT}(f, y) = g_f(y)$
- f does not hit D ϵ-correctly $\Leftrightarrow$ D disting. f from unif.
- f can be calculated by small A with D-gates + small advice.

Trevisan's method

$$\underset{\text{hard-function}}{f} \quad \xrightarrow{\text{N-W}} \quad \underset{\text{p.r.gen.}}{g_f}$$

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

- View the weakly random string as the truth-table of a function f
- $\mathrm{EXT}(f, y) = g_f(y)$
- f does not hit D ϵ-correctly $\Leftrightarrow$ D disting. f from unif.
- f can be calculated by small A with D-gates + small advice.
- f has a small description, so there are few f's like this.

Trevisan's method

$$\begin{array}{ccc} f & \xrightarrow{\mathrm{N-W}} & g_f \\ \text{hard-function} & & \text{p.r.gen.} \end{array}$$

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

- View the weakly random string as the truth-table of a function f
- $\mathrm{EXT}(f, y) = g_f(y)$
- f does not hit D ϵ-correctly $\Leftrightarrow$ D disting. f from unif.
- f can be calculated by small A with D-gates + small advice.
- f has a small description, so there are few f's like this.
- $|\{f \mid f \text{ does not hit } D\ \epsilon\text{-correctly}\}|$ is small

Trevisan's method

$$\begin{array}{ccc} f & \stackrel{\mathrm{N-W}}{\longrightarrow} & g_f \\ \text{hard-function} & & \text{p.r.gen.} \end{array}$$

N-W: If circuit D distinguishes $g_f(y)$ from unif., then using some small advice, one can transform D into A such that A computes f.

Trevisan:

- View the weakly random string as the truth-table of a function f
- $\mathrm{EXT}(f, y) = g_f(y)$
- f does not hit D ϵ-correctly $\Leftrightarrow$ D disting. f from unif.
- f can be calculated by small A with D-gates + small advice.
- f has a small description, so there are few f's like this.
- $|\{f \mid f \text{ does not hit } D\ \epsilon\text{-correctly}\}|$ is small
- So, by Lemma, EXT is an extractor.

Mimic Trevisan? Not quite.

- For exposure-resilient extractors, we need to consider distinguishers that have bounded access to f (recall weakly-random string $=$ f's truth-table).
- But no function is hard to an adversary that has access to its truth-table.
- So the Nisan-Wigderson schema does not work.
- Use the HILL construction of a p.r. gen. from a O-W function!
- A function may be O-W even if the adversary has bounded access to its truth-table.

using the HILL construction

$R : \{0,1\}^n \to \{0,1\}^n$.

$$\underset{\text{O-W func.}}{R} \xrightarrow{\text{HILL}} \underset{\text{p.r.gen.}}{g_R}$$

LEMMA (Distinguisher $\Rightarrow$ Inverter) -Informal version:
For any oracle circuit C, we can build an oracle circuit A, making just a few extra queries, so that if for some R:
C^R distinguisher of $(g_R(x), U_m) \Rightarrow$
A^R inverts a large fraction of $\{R(x) \mid x \in \{0,1\}^n\}$ OR
R is a "a-lot"-to-1.

LEMMA (Formal version)

- Let EXT be the extractor obtained via the HILL method. EXT depends on the parameters β, P ϵ, and m. Let $\epsilon' = \frac{1}{2^{3\beta n+1}} \cdot (\epsilon/m - \sqrt{P/2^{\beta n-1}})$.
- Let C be a circuit with query complexity Q. There exists a circuit A with query complexity $(Q+m) \cdot \mathrm{poly}(1/\beta, 1/\epsilon', 2^{\beta n})$ with the following property. Suppose R does not hit C ϵ-correctly. Then either
  - (a) R is not $P - to - 1$ or
  - (b) for at least a fraction $(1/8) \cdot \epsilon'$ of the random coins ρ used by B, it holds that

$$|\{x \in \{0,1\}^n \mid A^R(R(x), \rho) \in R^{-1}(R(x))\}| \geq \frac{1}{4} \cdot (\epsilon')^2 \cdot N.$$

LEMMA (Informal version)

- For any oracle circuit A, there are few R's such that A^R with $|R|^\delta$ queries (for any $\delta < 1$) inverts a large fraction of $\{R(x) \mid x \in \{0,1\}^n\}$.
- There are few R's that are "a-lot"-to-1.

LEMMA (Formal version):

(a) Let E be the event (over random R) "R is not $\alpha N - to - 1$." The probability of E is bounded by $2^{-\Omega(n \cdot \alpha N)}$.

(b) Let A be oracle circuit with query complexity S. Let B be the event (over random pairs (R, ρ))"$|\{x \in \{0,1\}^n \mid A^R(R(x), \rho) \in R^{-1}(R(x))\}| \geq 2e \cdot \alpha N \cdot S \cdot T$." The probability of B is bounded by $2^{-T} + 2^{-\Omega(n \cdot \alpha N)}$.

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.
- Let D be an adaptive test.

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.
- Let D be an adaptive test.
- Suppose R does not hit D^R ϵ-correctly

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.
- Let D be an adaptive test.
- Suppose R does not hit D^R ϵ-correctly
- D^R distinguisher

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.
- Let D be an adaptive test.
- Suppose R does not hit D^R ϵ-correctly
- D^R distinguisher
- From D we build A s.t. A^R inverter or R is "a-lot"-to-1.

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.
- Let D be an adaptive test.
- Suppose R does not hit D^R ϵ-correctly
- D^R distinguisher
- From D we build A s.t. A^R inverter or R is “a-lot”-to-1.
- There are few such R's.

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.
- Let D be an adaptive test.
- Suppose R does not hit D^R ϵ-correctly
- D^R distinguisher
- From D we build A s.t. A^R inverter or R is "a-lot"-to-1.
- There are few such R's.
- Few R's do not hit D ϵ-correctly

Proof - main steps

- View the weakly-random string as the truth-table of a function R. Let g_R be the function constructed from R using the HILL schema.
- Take $\mathrm{EXT}(R, y) = g_R(y)$.
- Let D be an adaptive test.
- Suppose R does not hit D^R ϵ-correctly
- D^R distinguisher
- From D we build A s.t. A^R inverter or R is "a-lot"-to-1.
- There are few such R's.
- Few R's do not hit D ϵ-correctly
- So (by Lemma), exposure-resilient extractor.

Parameters

$\mathrm{EXT} : \{0,1\}^{\tilde{N}} \times \{0,1\}^d \rightarrow \{0,1\}^m$.

- length of the weakly rand. string: $\tilde{N} = n \cdot 2^n$.
- query resistance $\tilde{N}^{\delta}$, for any $\delta < 1$
- entropy $k = \tilde{N} - \tilde{N}^{\Omega(1)}$
- seed length $d = O(\log \tilde{N})$
- output length $m = \tilde{N}^{\Omega(1)}$

Application: Derandomization of BPTIME[sublinear]

$L \in \mathrm{BPTIME}[T(n)]$:
There is a prob. alg. A running in time $T(n)$ such that

$$\forall x \mathrm{Prob}_{\rho}[A(x, \rho) = L(x)] > 2/3.$$

Things that can be done in probabilistic sublinear time:

- approx. matrix multiplication
- approx. min. spanning tree
- a lot of property testing
- ...

Theorem

There exist two constant natural numbers a and c such that for all $T(N) < N^{1/a}$, *any alg. in BPTIME*$[T(N)]$ *can be simulated deterministically in* $(T(N))^c$ *time and the deterministic simulator is correct on* $\geq (1 - 2^{-\Omega(T(N) \log T(N))})$ *fraction of inputs of length* N*, for all* N.

- $L \in \mathrm{BPTIME}[T(n)]$
- There is a prob. alg. A running in time $T(n)$ such that

$$\forall x \mathrm{Prob}_{\rho}[A(x, \rho) = L(x)] > 2/3.$$

- $W_x = \{\rho \mid A(x, \rho) = L(x)\}$.
- density$(W_x) > 2/3$.

- generic derandomization scheme: use a pseudo-rand. tool to obtain a small set Z such that for all x,

$$\frac{|Z \cap W_x|}{|Z|} \approx \mathrm{density}(W_x) > 2/3.$$

- we only need $|Z \cap W_x| > (1/2)|Z|$.
- simulate A repeteadly using as randomness the elements of Z and take the majority vote.
- usually, a p.r.gen. is used to build Z.
- p.r.gens are known to exist only under some hardness assumptions.

- Use extractors; extractors exist unconditionally.
- $\mathrm{EXT} : \{0,1\}^n \times \{0,1\}^d \rightarrow \{0,1\}^m$, (k, ϵ)-extractor.
- For any $u \in \{0,1\}^n$, $Z_u = \{E(u,y) \mid y \text{ seed}\}$; the samples induced by u.
- For any $B \subseteq \{0,1\}^m$, with prob. of $u \geq 1 - 2^{-(n-k-1)}$,

$$\frac{|Z_u \cap B|}{|Z_u|} =_\epsilon \mathrm{density}(B).$$

- Take W_x in the role of B.
- For a fraction of $1 - 2^{-(n-k-1)}$ of u's,

$$|Z_u \cap W_x| > (1/2)|Z_u|.$$

- We still use randomness u; so no derand. so far.
- How to get rid of u?

- Use x itself to obtain Z_x to hit W_x correctly!
- x and W_x are not independent; so why would this work?
- A is sublinear; checking if $\rho \in W_x$ depends on just a few bits of x.
- the rest of x is indep. of "$\rho \in W_x$."
- maybe we can use the rest of x to produce samples that hit W_x corectly?
- Indeed we can! Use an exposure-resilient extractor.

- $T(N)$ - running time of the prob. A, with $T(N) < N^{1/a}$.
- $\mathrm{EXT} : \{0,1\}^N \times \{0,1\}^d \rightarrow \{0,1\}^{T(N)}$, $(N - \Omega(T(N) \cdot \log T(N)), 1/6)$ exposure-resilient extractor, resistant to $T(n)$ queries.
- View W_x as computed by an adversary that can query $T(N)$ bits of x.
- Take $Z_x = \{E(x,y) \mid y \text{ seed}\}$.
- For $(1 - 2^{-\Omega(T(N) \log T(N))})$ fraction of x,

$$\frac{|Z_x \cap W_x|}{|Z_x|} =_{1/6} \mathrm{density}(W_x) > 2/3.$$

- For these x's: $|Z_x \cap W_x| > (1/2)|Z_x|$.
- Exactly what we need!

Other applications

- Increasing the Kolmogorov complexity of infinite sequences
- Derandomization for interesting classes of constraint satisfaction problems

Multumesc.