Extractors and Pseudorandom Generators

Luca Trevisan

Columbia University

Contents

- We present a new approach to construct extractors.
- Extractors transform a weakly random (realistic) source of randomness into an almost uniform (useful) one.
  - Extractors have a variety of other applications.
- Our construction
  - is stronger and simpler than previous ones;
  - reveals a new connection between extractors and pseudorandomness.
    (New direction: *from* pseudorandomness *to* extractors.)

Randomness in Computation

- Randomness is useful in designing efficient algorithms and data structures, and is essential in cryptography and in some distributed protocols.

- General tools to "manipulate" randomness are typically of the greatest interest.

- Extractors are a prime such tool.

Definitions

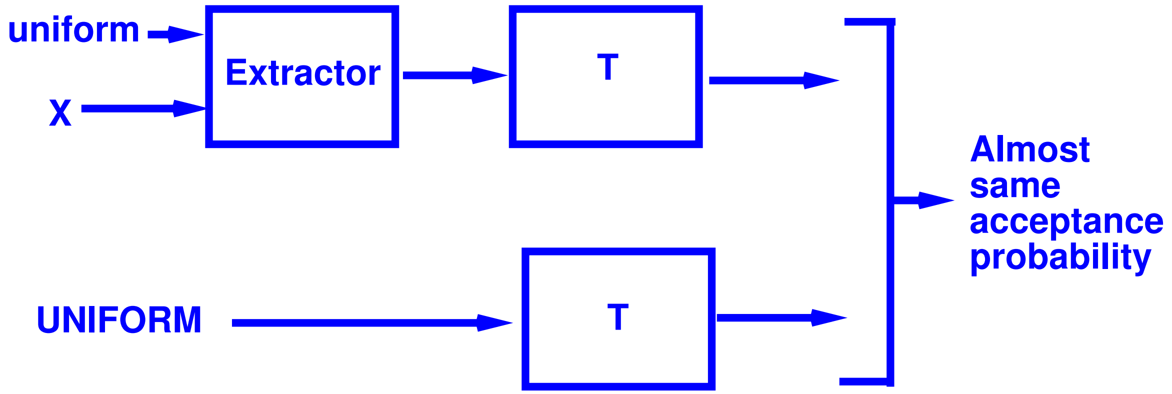
- A random source is modeled as a random variable X with range $\{0,1\}^n$.
  - X has min-entropy k if for every x, $\mathbf{Pr}[X = x] \leq 2^{-k}$. Then X contains "k bits of randomness"
- Y and Z are ϵ-close if for all "tests" $T : \{0,1\}^n \rightarrow \{0,1\}$

$$|\mathbf{Pr}[T(Y) = 1] - \mathbf{Pr}[T(Z) = 1]| \leq \epsilon$$

Extractor

A (k, ϵ)-extractor transforms an input of min-entropy k into a distribution ϵ-close to uniform.

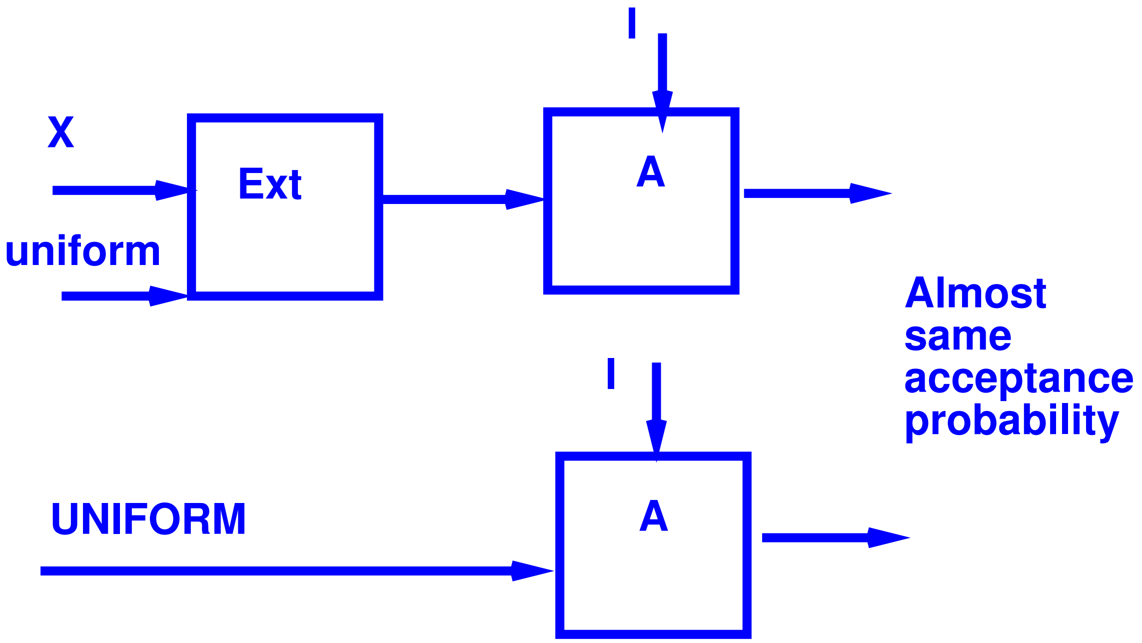
It uses a (small) amount of randomness to do the transformation.



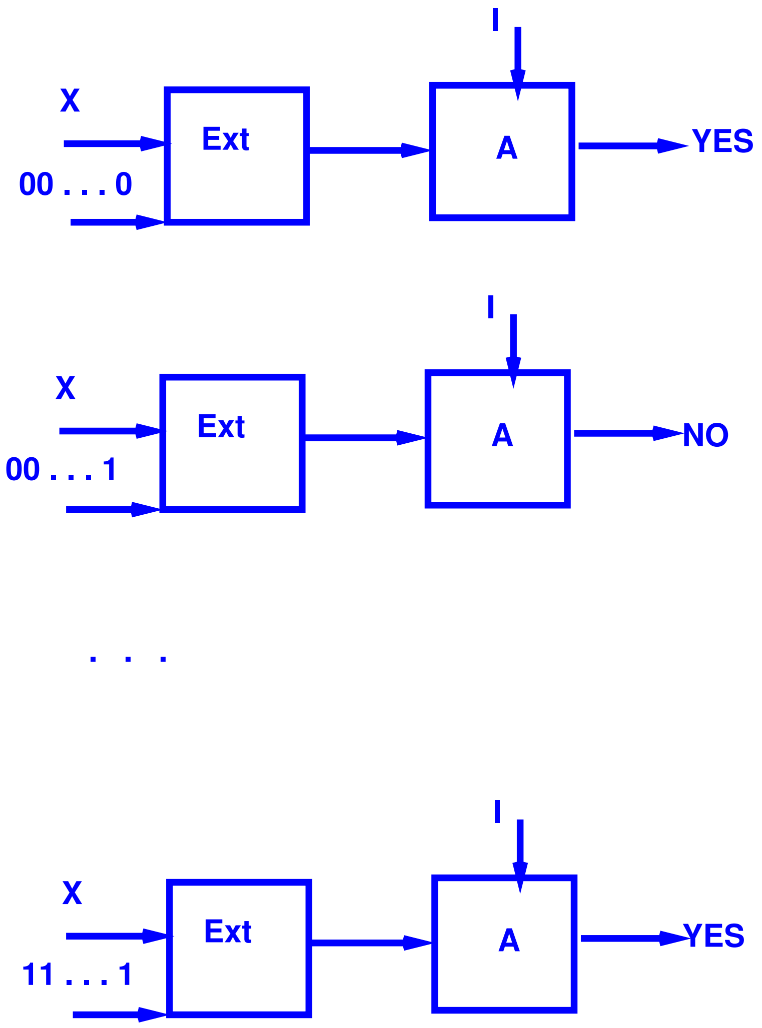
Applications of Extractors

- Simulation of randomized computations using weak sources of randomness.
- Randomness-efficient reduction of error in randomized algorithms.
- Construction of expanders, super-concentrators, sorting networks, and more.
- Miscellaneous applications in complexity theory.

Simulation of Randomized Algorithms



Without Additional Randomness



Take the majority answer

Additional randomness and parameters

We want (k, ϵ) extractors $Ext : \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^m$ where (for constant ϵ):

- t (additional randomness) is small:
  but there is a lower bound $t \geq \Omega(\log n)$
  - Important for applications to have $t = O(\log n)$.
- m (output length) is large:
  but there is a lower bound $m \leq k + t - 2\log 1/\epsilon$.

Extractors with $m = k + t - O(\log 1/\epsilon)$ and $t = O(\log n/\epsilon)$ exist. Explicit constructions are hard.

Previous Results and Ours

Dispersers are weaker than extractors.

Reference	Min entropy k	Output length m	Additional randomness t	Type
Zuckerman'96	γn	$(1-\delta)k$	$O(\log n)$	Ext.
Ta-Shma'96	any k	k	$O((\log n)^9)$	Ext.
	n^γ	$k^{1-\delta}$	$O(\log n \log \cdots \log n)$	Ext.
Saks et al. '96	n^γ	$k^{1-\delta}$	$O(\log n)$	Disp.
Ta-Shma'98	any k	$k-(\log n)^{O(1)}$	$O(\log n)$	Disp.
This talk	n^γ	$k^{1-\delta}$	$O(\log n)$	Ext.
	any k	$k^{1-\delta}$	$O((\log n)^2/\log k)$	Ext.

δ, γ can be arbitrarily small.

Later improvements by Raz, Reingold and Vadhan.

Interlude: Pseudorandomness

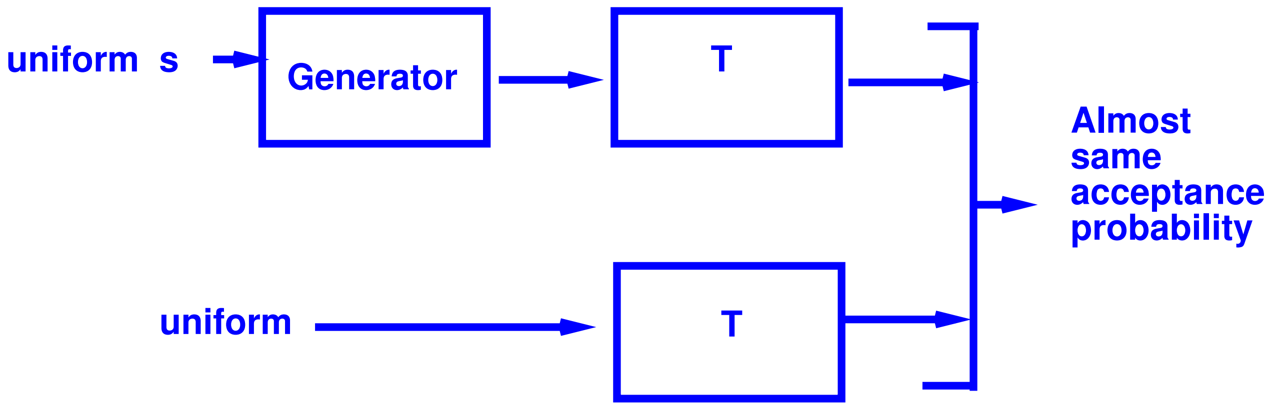
- Two distributions Y and Z over $\{0,1\}^m$ are (ϵ, S)-indistinguishable if
  - for every $T : \{0,1\}^m \to \{0,1\}$ computable by a circuit of size $\leq S$

$$|\mathbf{Pr}[T(Y) = 1] - \mathbf{Pr}[T(Z) = 1]| \leq \epsilon$$

- Recall: Y and Z are ϵ-close if for all $T : \{0,1\}^m \to \{0,1\}$
  $|\mathbf{Pr}[T(X) = 1] - \mathbf{Pr}[T(Y) = 1]| \leq \epsilon$

Pseudorandom Generator

$G : \{0,1\}^t \to \{0,1\}^m$ is a (S, ϵ) pseudorandom generator if, for a random input, the output is (S, ϵ)-indistinguishable from uniform. (Interesting when $m >> t$.)

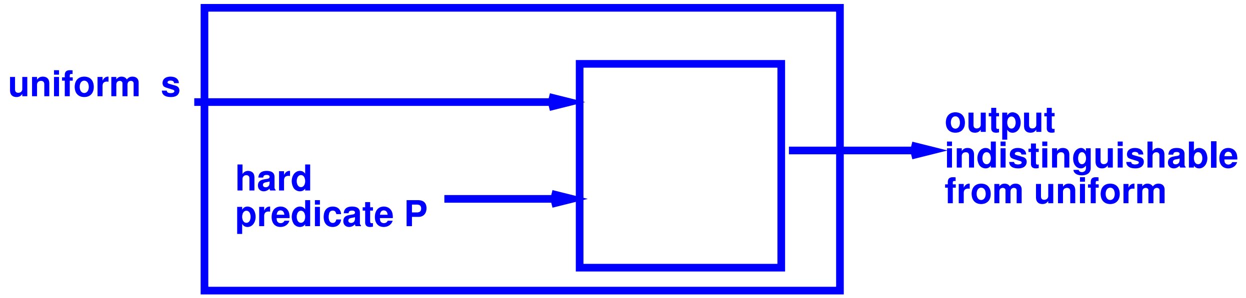


PRG Based on a Hard Predicate

Constructions by

- Nisan & Wigderson '88 (simpler but weaker) and
- Impagliazzo & Wigderson '97 (stronger but more complicated)

are based on a computationally hard predicate.



PRG Constructions by NW and IW

The generator has oracle access to the predicate.

Seed length $t = O(\log m)$, input length of the predicate $l = O(\log m)$, output length m, output $(O(m), 1/10)$-indistinguishable from uniform assuming:

- No circuit of size $2^{o(l)}$ computes P. [IW]
  Worst-case hardness assumption.

- No circuit of size $2^{o(l)}$ computes P on more than a fraction $1/2 + 2^{-o(l)}$ of inputs. [NW]
  Average-case hardness assumption.

Proof (for IW)

Let $G(\cdot)$ be IW generator with predicate P.

Suppose, for some T, $\mathbf{Pr}[T(G(U_t)) = 1] \not\approx \mathbf{Pr}[T(U_m) = 1]$,

Then IW show that there exists a small circuit A s.t.

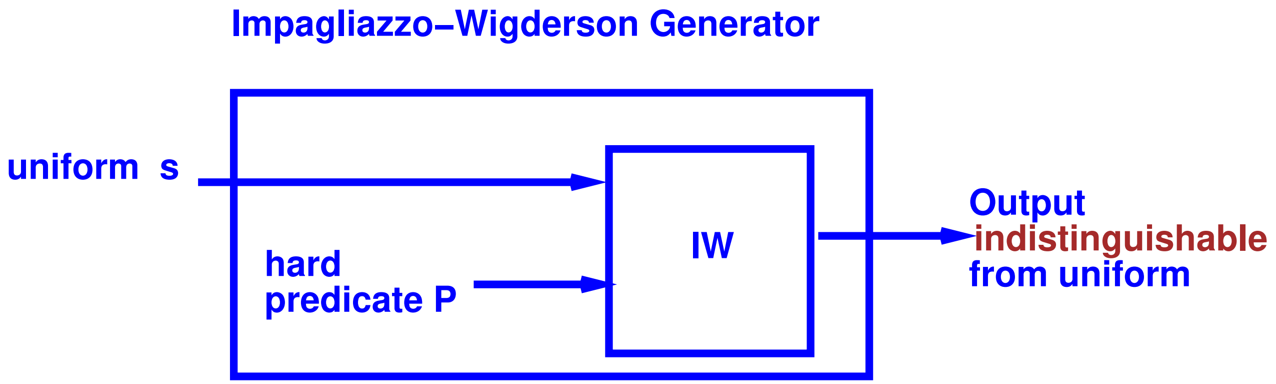
A with oracle access to T computes P.

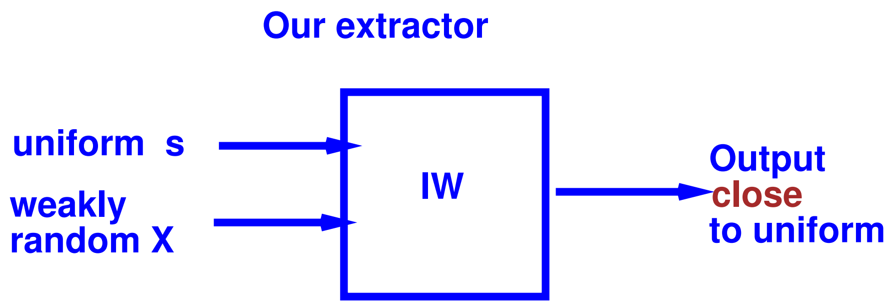
Contradition if P is hard and T is easy.

Important note: the IW proof works independently of the complexity of T.

An Extractor from the IW Generator

View the input of the extractor as the truth-table of a predicate. Use IW.





Our Analysis

Fix test T, of arbitrary complexity.

Every x such that $\mathbf{Pr}[T(Ext(x, U_t)) = 1] \not\approx \mathbf{Pr}[T(U_m) = 1]$ has a short description given T.

T is fixed, and X has large min-entropy.

There is low probability that x sampled from X has small description given T.

Then $\mathbf{Pr}[T(Ext(X, U_t)) = 1] \approx \mathbf{Pr}[T(U_m) = 1]$.

Consequence

Every construction of pseudorandom generators that

- is based on a worst-case predicate
- has a "black-box" analysis

is an extractor.

From IW we get for every $\epsilon, \gamma > 0$ a (k, ϵ)-extractor $Ext : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ where $k = n^\gamma$ and $m = k^{\Omega(1)}$

Better than previous constructions!

Structure of the Proof

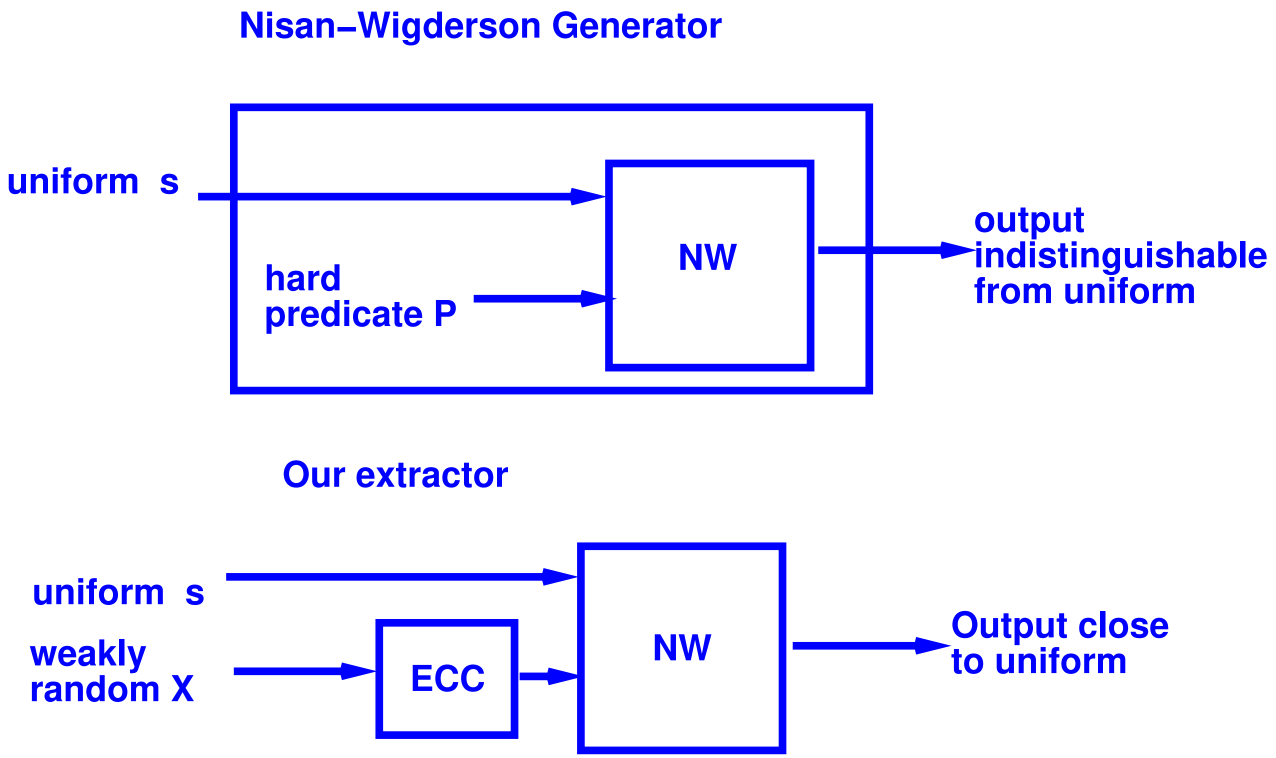
We want to prove that when we fix a statistical test, we will almost always "fool" it with the extractor (i.e. the test will not tell the difference between the output of the extractor and the uniform distribution).

We prove so by showing that the cases when the test is not fooled have small descriptions.

Then there are few such cases and the probability that one of them happen is small.

An Extractor From the NW Generator

Encode the input with an error correcting code. Do as before.



Analysis

Fix test T.

If x is s.t. $\mathbf{Pr}[T(Ext(x, U_t)) = 1] \not\approx \mathbf{Pr}[T(Ext(U_m))] = 1$ then $ECC(x)$ is "approximated" by a string having short description given T.

T is fixed, and X has large min-entropy

There is low probability that x sampled from X is such that $ECC(x)$ is approximated by string with short description given T.

Then $\mathbf{Pr}[T(Ext(X, U_t)) = 1] \approx \mathbf{Pr}[T(Ext(U_m)) = 1]$.

Use of Error-Correcting Codes

- When we pick a string at random with large min-entropy, then with high probability the string does not have a short description.

- When we pick a string at random with large min-entropy, and then encode it with error-correcting code, then with high probability the encoding is not even close (in Hamming distance) to a string with short description).

(The error-correcting code must have the property that there are few codewords in any ball of large radius)

Advantages

The Nisan-Wigderson generator is simple to describe and analyze.

The whole construction can now be described from the ground up in a few lines without reference to previous work (except for standard error-correcting codes).

In particular, without reference to previous work on pseudorandomness.

The proof of correctness is also simple.

The Extractor — Abstract View

Primitives:

- we have an error correcting code $EC : \{0,1\}^n \rightarrow \{0,1\}^{\bar{n}}$ with $\bar{n} = \mathrm{poly}(n)$ and with the few-codewords-in-any-ball guarantee. (Standard.)
- we have m functions $\pi_1, \ldots, \pi_m$ where $\pi_i : \{0,1\}^t \rightarrow \bar{n}$, with certain properties. (NW.)

Construction:

$$Ext(x, s) = \bar{x}[\pi_1(s)]\bar{x}[\pi_2(s)] \cdots \bar{x}[\pi_m(s)]$$

where $\bar{x} = EC(x)$, and $\bar{x}[j]$ is the j-th entry of $\bar{x}$.

The Extractor — All the Details

Primitives:

- we have an error correcting code $EC : \{0,1\}^n \to \{0,1\}^{\bar{n}}$ as before.

- we have sets $S_1, \ldots, S_m \subseteq \{1, \ldots, t = O(\log n)\}$ s.t. $|S_i| = \log \bar{n}$ and $|S_i \cap S_j| \leq .01 \log \bar{n}$. (NW.)

Notation:

For a string $z \in \{0,1\}^t$ and a set $S \subseteq \{1, \ldots, t\}$, we denote by $z_{|S}$ the projection of z on the coordinates given by S.

Construction:

For a seed s, $\pi_i(s)$ is defined as the number whose binary representation is $s_{|S_i}$.

Extractor:

$$Ext(x, s) = EC(x)[s_{|S_1}] \cdots EC(x)[s_{|S_m}]$$

Conclusions

- High level view:
  - NW/IW: output of generator is indistinguishable from uniform if predicate is fixed and hard.
  - We: output is statistically close to uniform if predicate is random and has large min-entropy.

- Novelties in our approach
  - View the predicate in NW/IW as part of the input.
  - Show information-theoretic applications of pseudorandomness. (First time of a connection in this direction)