On the derandomization of BPTIME[sublinear]

Marius Zimand
Towson University

June 2007
- Derandomization = deterministic simulation of a prob. algorithm without a large time overhead.

- Practical application? - Usually there is some overhead, sometimes the simulation does not work for all inputs.

- Theoretical interest: Is randomness necessary? When is it? When is it not?
Best of Derandomization I - Conditional Results

➤ (Impagliazzo, Wigderson '97) If: \( \exists \alpha > 0, \exists \) problem in E that requires circuits of size \( 2^{\alpha n} \).
   Then: \( \text{BPP} = \text{P} \).

➤ (Impagliazzo, Wigderson '98) If: \( \text{EXP} \neq \text{BPP} \).
   Then: any problem in \( \text{BPP} \) can be solved in determ. subexponential time at infinitely many input lengths, and at these length the determ. alg. is correct on all inputs except a negligible fraction.

➤ (Goldreich, Zuckerman'97) If: \( \exists \alpha > 0, \exists \) problem in E that requires circuits of size \( 2^{\alpha n} \).
   Then: \( \text{MA} = \text{NP} \).

➤ (Miltersen, Vinodchandran’ 99) If: \( \exists \alpha > 0, \exists \) problem in \( \text{NE} \cap \text{co-NE} \) that requires nondeterministic circuits of size \( 2^{\alpha n} \).
   Then: \( \text{AM} = \text{NP} \).
Best of Derandomization II - Unconditional Results

- (Rabin’63) 1-way pfa = dfa. (Freivalds’81: 2-way pfa ≠ dfa.)
- (Kaneps, Freivalds’90 and Dwork, Stockmeyer’92) 2-way pfa & subexp. time = dfa.
- (Nisan’91) BP.AC$^0 \subseteq \text{TIME}[n^{\text{poly} \log n}]$.
- (Nisan’92) PRG for small space:
  For every $S$-space $R$-random bits prob. alg $A$ and every input $x$, there is

$$G : \{0,1\}^{S \log R} \rightarrow \{0,1\}^R$$

so that $A(x, U_R)$ and $A(x, G(U_{S \log R}))$ are very close. $G$ can be computed from $A$ and $x$ in time $\text{poly}(R)$ and space $O(S)$.
THIS PAPER: Unconditional result for a weak computational model: BPTIME[sublinear].

BPTIME[sublinear] = decision problems, solvable by prob. algorithms in sublinear time, with bounded error.
Sublinear time used to be illegal

\[ [\text{Hopcroft} & \text{ Ullman'79, pp. 288}]: \text{"time complexity } T(n) \text{ means max}(n + 1, \lceil T(n) \rceil)." \]
Times have changed...

Things that can be done in probabilistic sublinear time:

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

Marius Zimand
Derand. of BPTIME(sublinear)
MAIN THEOREM:

There exist two constant natural numbers $a$ and $c$ such that for all $T(N) < N^{1/a}$, any alg. in $\text{BPTIME}[T(N)]$ 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 $N$.

- the non-uniform version is trivial.
- Current estimation: $a = 45$.
- Derandomization works also for promise-$\text{BPTIME}[T(N)]$: the simulator is correct on all inputs on which the probabilistic algorithm is correct except at most a $2^{-\Omega(T(N) \log T(N))}$ fraction.
Generic approach

- $L \in \text{BPTIME}[T(n)]$
- There is a prob. alg. $A$ running in time $T(n)$ such that
  \[ \forall x \ \text{Prob}_\rho[A(x, \rho) = L(x)] > 2/3. \]
- $W^x = \{ \rho \mid A(x, \rho) = L(x) \}$, set of witnesses for $x$.
- $\text{density}(W^x) > 2/3.$
Generic approach

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

$$\frac{|Z \cap W^x|}{|Z|} \approx \text{density}(W^x) > \frac{2}{3}.$$ 

- we only need $|Z \cap W^x| > \frac{1}{2}|Z|$.

- simulate $A$ repeatedly 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.
Generic approach

- Use extractors; extractors exist unconditionally.
- Extractors vs. PRGs: some aspects are similar, some aspects are very different
Extractor vs. PRG

Pseudo-rand. generator

\[ f \xleftarrow{\text{hard (N-W)}} y \quad \xrightarrow{\text{one-way (B-M-Y)}} \quad (HILL) \]
\[ g(y) \quad \text{m bits} \]

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

\[ \left| \Pr(g(y) \in A) - \frac{|A|}{2^m} \right| < \epsilon \]

(k, \( \epsilon \))-Extractor

\[ X \text{ (weakly rand. string)} \quad \text{min-entropy}(X) = k \]
\[ \quad \xrightarrow{\text{y (seed)}} \quad \text{EXT} \]
\[ \text{EXT}(X, y) \quad \text{m bits} \]

for all \( A \subseteq \{0,1\}^m \),

\[ \left| \Pr(\text{EXT}(X, y) \in A) - \frac{|A|}{2^m} \right| < \epsilon \]
EXT : \{0, 1\}^n \times \{0, 1\}^d \rightarrow \{0, 1\}^m, (k, \epsilon)-extractor.

For any \( u \in \{0, 1\}^n \), \( Z_u = \{\text{EXT}(u, y) \mid y \text{ seed}\} \); the samples induced by \( u \).

For any \( B \subseteq \{0, 1\}^m \), with prob. of \( u \geq 1 - 2^{-(n-k-1)} \),

\[
\frac{|Z_u \cap B|}{|Z_u|} = \epsilon \ \text{density}(B).
\]

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 \( u \)?
Goldreich and Wigderson'2002- use extractors for derandomization

- In their applications, the set $W$ to be hit correctly was a large set of short advice strings;
- $W$ did not depend on $x$.
- They used $x$ itself to obtain $Z_x$ to hit $W$ correctly using a standard extractor.
The new idea:

- Here $W^x$ does depend on $x$.
- 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 $W^x$ correctly?
- Indeed we can! Use an exposure-resilient extractor.
Digression: exposure-resilient extractors

- **Extractor** = procedure that transforms imperfect randomness into (almost) perfect randomness (information-theoretic randomness).
Digression: exposure-resilient extractors

- **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.
Standard extractors

Review the def. of a (standard) extractor

- $\text{EXT} : \{0, 1\}^n \times \{0, 1\}^d \rightarrow \{0, 1\}^m$
- $\text{EXT}(x, y)$
- $x$ - “weakly random string,”, min-entropy$(x) = k$
- $y$ - “seed,” unif. distributed
- adversary - computationally unbounded
Standard extractor: the game view

- Adversary is given the challenge $Z$ which is either
  1. $\text{EXT}(x, y)$, OR
  2. $U_m$
- Adv. wants to distinguish (1) from (2) with bias $\epsilon$. $\text{EXT}$ is a $(k, \epsilon)$-extractor if no adv. succeeds.
Exposure-resilient extractor: the game view

- Adversary is given the challenge $Z$ which is either
  1. $\text{EXT}(x, y)$, 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 $\epsilon$.
- $\text{EXT}$ is a $(k, \epsilon)$-exposure-resilient extractor resistant to $q$ queries if no adv. succeeds.
(Standard) Extractor

m-bit strings

EXTRACT(X,y_1)

W

EXTRACT(X,y_D)

Exposure-resilient extractor

m-bit strings

EXTRACT(X,y_1)

W_X

EXTRACT(X,y_D)
How to build an exposure-resilient extractor

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

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

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

Here $W$ is an oracle circuit that can query $q$ bits of its oracle.

**DEF.** $x$ hits $W$ $\epsilon$-correctly if

$$\left| \left| \left| \frac{\{y \mid \text{EXT}(x, y) \in W^x\}}{D} - \frac{|W^x|}{2^m} \right| \right| < \epsilon$$

**Lemma.** Let $\text{EXT} : \{0, 1\}^n \times \{0, 1\}^d \to \{0, 1\}^m$. If $\forall W$, no. of $X$ that do not hit $W$ $\epsilon$-correctly is $\leq 2^t$

$\Rightarrow$ $\text{EXT}$ is a $(t + \log(1/\epsilon), 2\epsilon)$-extractor with $q$ query resistance.
Using the Håstad-Impagliazzo-Levin-Luby construction

\[ R : \{0, 1\}^n \rightarrow \{0, 1\}^n. \]

We take \( \text{EXT}(R, y) = g_R(y) \) (à la Trevisan).

\text{HILL}: If \( g_R \) has a distinguisher, then \( R \) is not one-way.

\text{Here}: If \( \text{EXT}(R, \cdot) \) does not hit \( W^R \) correctly, then \( R \) belongs to a small set.
Using the HILL construction

\[ R : \{0, 1\}^n \rightarrow \{0, 1\}^n. \]

**LEMMA (Distinguisher ⇒ 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 \( R \):
\[ C^R \text{ distinguisher of } (g_R(x), U_m) \Rightarrow \]
\[ A^R \text{ inverts a large fraction of } \{R(x) | x \in \{0, 1\}^n\} \text{ OR} \]
\( R \) is a “a-lot”-to-1.
LEMMA (Formal version)

$R : \{0, 1\}^n \rightarrow \{0, 1\}^n$, $m$ - parameter.

Let $C$ be an oracle circuit with query complexity $m$.

There is a set of $2^{m+O(\log m)}$ circuits with query complexity $\text{poly}(m)$ such that if $C^R$ is not hit $\epsilon$-correctly by $\text{EXT}(R, \cdot)$ then either

- $R$ is not $m$-to-1, or
- $R$ is “good” for some circuit in the set.

(DEF: $R$ is “good” for circuit $A$ if for more than $1/2$ of $\rho$’s $|\{x \mid A^R(R(x), \rho) \in R^{-1}(R(x))\}| > \text{poly}(m)$. )
LEMMA (Informal version)

- For any oracle circuit $A$, there are few $R$'s that are “good” for $A$.
- There are few $R$’s that are “a-lot”-to-1.
LEMMA (Formal version):

$R : \{0, 1\}^n \rightarrow \{0, 1\}^n$, $m$ - parameter.

(a) Let $E$ be the event (over random $R$) “$R$ is not $m - to - 1$.”

The probability of $E$ is bounded by $2^{-\Omega(m \cdot \log m)}$.

(b) Let $A$ be oracle circuit with query complexity $m$. The

probability that $R$ is “good” for $A$ is bounded by $2^{-\Omega(m \cdot \log m)}$. 
$R$ does not hit $D^R \, \epsilon$-correctly

$D^R$ distinguisher

from $D$ obtain a list of circuits

$R$ is “good” for some circuit in the list or $R$ is “a-lot”-to-1.

There are few such $R$’s.

Few $R$’s do not hit $D \, \epsilon$-correctly

exposure-resilient extractor (by Useful Lemma).
Parameters

$\text{EXT} : \{0, 1\}^N \times \{0, 1\}^d \to \{0, 1\}^m$.

- output length $m \leq N^{1/45}$.
- query resistance $m$
- entropy $k = N - \Omega(m \log m)$
- seed length $d = O(\log m)$
- computable in $m \cdot \text{polylog}(m)$ time
Back to derandomization of $\text{BPTIME}$(sublinear)

- $A$ - Prob. alg. that $\text{BPTIME}[\text{sublinear}]$-computes $L$.
- $T(N)$ - running time of $A$, $T(N) < N^{1/45}$.
- $\text{EXT} : \{0, 1\}^N \times \{0, 1\}^{O(\log(T(N))} \rightarrow \{0, 1\}^{T(N)}$, $(N - \Omega(T(N) \cdot \log T(N)), 1/6)$ exposure-resilient extractor, resistant to $T(N)$ queries.
- Assume $x$ in $L$ (the other case is similar).
- $W^x = \{\rho \mid A(x, \rho) = 1\}$. 
Back to derandomization of BPTIME(sublinear)

- \( W^x = \{ \rho \mid A(x, \rho) = 1 \} \).
- 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 \( x \),
  \[
  \frac{|Z_x \cap W^x|}{|Z_x|} = \frac{1}{6} \text{ density}(W^x) > \frac{2}{3}.
  \]
- For these \( x \)'s: \( |Z_x \cap W^x| > \frac{1}{2}|Z_x| \).
- Exactly what we need!
Thank you.