On One-way Functions and Kolmogorov Complexity

Rafael Pass
Cornell Tech

Joint work with Yanyi Liu
The “Dark Ages” Crypto Cycle
(the last 2000 years)

"artist" invents scheme

scheme deployed

scheme broken

known attacks fail
One-way Functions (OWF) [Diffie-Hellman’76]

A function $f$ that is
• **Easy to compute**: can be computed in poly time
• **Hard to invert**: no PPT can invert it

**Ex [Factoring]**: use $x$ to pick to 2 random “large” primes $p,q$, and output $y = p \times q$
One-way Functions (OWF) [Diffie-Hellman’76]

A function $f$ that is
- **Easy to compute**: can be computed in poly time
- **Hard to invert**: no PPT can invert it

**Definition 2.1.** Let $f : \{0,1\}^* \rightarrow \{0,1\}^*$ be a polynomial-time computable function. $f$ is said to be a one-way function (OWF) if for every PPT algorithm $A$, there exists a negligible function $\mu$ such that for all $n \in \mathbb{N},$

$$\Pr[x \leftarrow \{0,1\}^n; y = f(x) : A(1^n, y) \in f^{-1}(f(x))] \leq \mu(n)$$
One-way Functions (OWF) [Diffie-Hellman’76]

A function $f$ that is
• **Easy to compute**: can be computed in poly time
• **Hard to invert**: no PPT can invert it

OWF both necessary [IL’89] and sufficient for:
• Private-key encryption [GM84,HILL99]
• Pseudorandom generators [HILL99]
• Digital signatures [Rompel90]
• Authentication schemes [FS90]
• Pseudorandom functions [GGM84]
• Commitment schemes [Naor90]
• Coin-tossing [Blum’84]
• ZK proofs [GMW89]
• ...

**Not included:**
public-key encryption, OT, obfuscation

Whether OWF exists is the most important problem in Cryptography
OWF v.s NP Hardness

Observation: OWF => NP \notin BPP

“Holy grail” [DH’76]

Prove: NP \notin BPP => OWF

Lots of partial BB “separations”: [Bra’79],[AGGM’06],[P’07],[MX’10]
In the absence of the holy-grail...

Factoring [RSA’83]

Discrete Logarithm Problem [DH’76]

Lattice Problems [Ajtai’96]

DES,
SHA,
AES...

So far, not broken...but for how long?
“Cryptographers seldom sleep well” - Micali’88

Have we really escaped from the “crypto cycle”?
In the absence of the holy-grail...

Discrete Logarithm Problem [DH’76]

Factoring [RSA’83]

Lattice Problems [Ajtai’96]

DES, SHA, AES...

**Central question:** Does there exist some natural average-case hard problem (a “mother problem”) that characterizes existence of OWF?
Main Theorem

For every polynomial $t(n) > 1.1n$:

OWFs exist iff t-bounded Kolmogorov-complexity is mildly hard-on-average
Kolmogorov Complexity [Sol’64,Kol’68,Cha’69]

Which of the following strings is more “random”:
• 1231231231231231231
• 1730544459347394037

$\mathbf{K}(x) = \text{length of the shortest program that outputs } x$

Formally, we fix a universal TM $U$, and are looking for the length of the shortest program $\Pi = (M,w)$ s.t. $U(M,w) = x$

Lots of amazing applications (e.g., Godel’s incompleteness theorem)
But uncomputable.
Which of the following strings is more “random”:
  • 1231231231231231231
  • 1730544459347394037

\[ K(x) = \text{length of the shortest program that outputs } x \]
\[ K^t(x) = \text{length of the shortest program that outputs } x \text{ within time } t(|x|) \]

Can \( K^t \) be **efficiently computed** when \( t(n) \) is a polynomial?
  • Studied in the Soviet Union since 60s [Kol’68,T’84]
  • Independently by Hartmanis [83], Sipser [83], Ko [86]
  • Closely related to **MCSP** (Minimum Circuit Size Problem) [T’84,KC’00]
Average-case Hardness of $K^t$

**Frequentional version** [60’s, T’84]
Does $\exists$ algorithm that computes $K^t(x)$ for a “large” fraction of $x$’s?

**Observation** [60’s, T’84]: $K^t$ can be approximated within $d \log n$ w.p $1-1/n^d$
Proof: simply output $n$.

**Def**: $K^t$ is **mildly-HOA** if there exists a polynomial $p$, such that no PPT heuristic $H$ can compute $K^t$ w.p $1-1/p(n)$ over random strings $x$ for inf many $n$.

**Def**: $K^t$ is **mildly-HOA to c-approximate** if there exists a polynomial $p$, such that no PPT heuristic $H$ can c-approximate $K^t$ w.p $1-1/p(n)$ over random strings $x$ for inf many $n$. 

Main Theorem

The following are equivalent:

1. **OWFs** exist
2. \( \exists \text{ poly } t(n) > 0, \text{ s.t. } K^t \text{ is mildly-HOA.} \)
3. \( \forall c > 0, \varepsilon > 0, \text{ poly } t(n) > (1 + \varepsilon) n, \)
   \( K^t \text{ is mildly-HOA to } (\text{clog } n) \)-approx.
Main Theorem

The following are equivalent:

1. **OWFs** exist

2. \( \exists \) poly \( t(n) > 0 \), s.t. \( K^t \) is mildly-HOA.

3. \( \forall \) \( c > 0, \varepsilon > 0 \), poly \( t(n) > (1 + \varepsilon) n \),
   \( K^t \) is mildly-HOA to \( (c \log n) \)-approx.

**Corr:** For all poly \( t(n) > (1 + \varepsilon)n \),
OWFs exists iff \( K^t \) is mildly hard-on-average.

**Corr:** For all \( c > 0, \varepsilon > 0 \), poly \( t(n) > (1 + \varepsilon) n \),
\( K^t \) is mildly hard-on-average to \( (c \log n) \)-approx iff \( K^t \) is mildly hard-on-average.
Earlier Connections between OWF and $K^t$

- [RR’97, KC00, ABK+06]: OWF $\implies$ exists poly $t$ s.t $K^t$ is worst-case hard
  - converse direction not known
  - this will be our starting point to showing OWF $\implies K^t$ is HOA

- [Santhanam’19]: Under a new conjecture, MCSP is “errorless-HOA” iff OWF exists
  - as mentioned, MCSP is closely related to $K^t$
  - in contrast, our results are unconditional.
Main Theorem

The following are equivalent:
1. OWFs exist
2. \( \exists \) poly \( t(n) > 0 \), s.t. \( K^t \) is mildly-HOA.
3. \( \forall \ c > 0, \ \epsilon > 0, \) poly \( t(n) > (1 + \epsilon) n \),
   \( K^t \) is mildly-HOA to \( (c \log n) \)-approx.

Proof: \((2) \implies (1) \implies (3)\)

Today: just sketch \((1) \iff (2)\)
Theorem 1

Assume there exists some poly $t(n) > 0$, s.t. $K^t$ is mildly-HOA. Then OWFs exist.

Theorem 2

Assume OWFs exists. Then there exists some poly $t(n) > 0$ s.t. $K^t$ is mildly-HOA.
Theorem 1

Assume there exists some poly \( t(n) > 0 \), s.t. \( K^t \) is mildly-HOA. Then OWFs exist.

Weak OWF: “mild-HOA version” of a OWF: efficient function \( f \) s.t. no PPT can invert \( f \) w.p. \( 1 - \frac{1}{p(n)} \) for inf many \( n \), for some poly \( p(n) > 0 \).

Lemma [Yao’82]. If a Weak OWF exists, then a OWF exists.

So, we just need to construct a weak OWF.
Let $c$ be a constant so that $K^t(x) < |x|+c$ for all $x$.

Define $f(\Pi', i)$ where $|\Pi'| = n$, $|i| = \log (n+c)$ as follows:
- Let $\Pi =$ first $i$ bits of $\Pi'$ (i.e., truncate $\Pi'$ to $i$ bits).
- Let $y =$ output of $\Pi$ after $t(n)$ steps.
- Output $i || y$.

Assume for contradiction that $f$ is not a Weak OWF. Then, for every inverse polynomial $\delta$, there exists a PPT attacker $A$ that inverts $f$ w.p $1-\delta$.

We construct a heuristic $H$ (using $A$) that computes $K^t$ w.p. $1-\delta O(n)$, which concludes that $K^t$ is not mildly HOA, a contradiction.
Heuristic $H(y)$ proceeds as follows given $x \in \{0,1\}^n$:

- For $i = 1$ to $n+c$
  - Run $A(i|y) \rightarrow \Pi$ and check if $\Pi$ outputs $y$ within $t(n)$ steps
- Output the smallest $i$ for which the check passed.

*Intuitively*, if $A$ succeeds with VERY high probability, then it should also succeed with high probability conditioned on length $i$, for EVERY $i \in [n+c]$

*But*: the problem is that $H$ is feeding $A$ the **wrong distribution** over $y$’s.
In OWF experiment (where A works):

\[ i \leftarrow U_{\log(n+c)} \]
\[ y \leftarrow \text{output of a random program of length } i \]

In the emulation by H in \( K^t \) experiment (where we need to prove that A works):

\[ i \leftarrow K^t(y) \]
\[ y \leftarrow U_n \]

No reason to believe that the output of a random program will be close to uniform!

But: using a counting argument, we can show that they are not too far in relative distance
**Key idea:**
- Assume for simplicity that $A$ is deterministic.
- Consider some string $y$ on which $H$ fails. $y$ has prob mass $2^{-n}$ in the $K_t$ exp.
- For $H(y)$ to fail, $A(w||y)$ must fail where $w = K_t(y)$.
- But the pair $w||y$ is sampled in the OWF exp w.p.
  
  \[
  \frac{1}{(n+c)} \times 2^{-w} > \frac{1}{(n+c)} \times 2^{-n+c} > \frac{1}{O(n)} \times 2^{-n}
  \]
- So, if $H$ fails w.p. $\varepsilon$, $A$ must fail w.p $\varepsilon / O(n) \leq \delta$
- Thus, $H$ fails w.p $\varepsilon \leq \delta O(n)$
Theorem 1

Assume there exists some poly \( t(n) > 0 \), s.t. \( K^t \) is mildly-HOA. Then OWFs exist.

Theorem 2

Assume OWFs exist. Then there exists some poly \( t(n) > 0 \) s.t. \( K^t \) is mildly-HOA.
**Theorem 1**

Assume there exists some poly \( t(n) > 0 \), s.t. \( K^t \) is mildly-HOA. Then OWFs exist.

**Theorem 2**

Assume OWFs exist. Then there exists some poly \( t(n) > 0 \) s.t. \( K^t \) is mildly-HOA.
Theorem 2

Assume OWFs exists.
Then there exists some poly $t(n)>0$ s.t. $K^t$ is mildly-HOA.

High-level Idea [KC’00,ABK+’06]:

• Use OWF $f$ to construct a PRG $G:\{0,1\}^n \to \{0,1\}^{2n}$ [HILL’99] (output of $G(U_n)$ is indistinguishable from $U_{2n}$ by PPT observers)

• Use algorithm $H$ for computing $K^t$ to distinguish output of PRG from random, where $t = \text{running time of } G$, which yields a contradiction.
So any algorithm $H$ that computes $K^t$ can break the PRG.

**Important:**
- Only works if $H$ computes $K^t$ w.p 1.
- if $H$ is just a heuristic (that works w.p 1-neg), then we have no guarantees: $H$ can fail on all pseudorandom strings, as they have tiny probability mass!
Entropy-preserving PRG (EP-PRG)

Efficiently computable function $G: \{0,1\}^n \rightarrow \{0,1\}^{n+c \log n}$

- **Pseudorandomness**: $G(U_n)$ indistinguishable from $U_{n+c \log n}$
- **Entropy-preserving**: $G(U_n)$ has Shannon entropy $n-O(\log n)$

**Lemma**: EP-PRG with running time $t$ implies $K^t$ is mildly-HOA
If $G$ is an EP-PRG, then $H(y) < n + O(1)$ w.p $O(1)/n^2$ given pseudo random samples

**Idea:**
- If Shannon entropy is $n - O(\log n)$, then using an averaging argument, there exists a set $S$ of strings in the support of $G(U_n)$, s.t.
  - for every $y \in S$, $\Pr[G(U_n) = y] < 2^{-(n - O(\log n))}$
  - $\Pr[S] > 1/n$
- That is, conditioned on $S$, the relative distance from uniform is small, and we can use the same argument as for Thm 1 to argue that $H$’s failure probability will be small.
Constructing EP-PRG

**Good News:** GL’89 construction of a PRG from a **OWP** $f$ is entropy preserving.

$$G(s, r) = r, f(s), GL(s, r)$$

**Bad News:**
- HILL’99 construction of a PRG from **OWF** is not entropy preserving (as far as we can tell)
- Don’t know how to obtain an EP-PRG from OWF...

**Need to relax the notion of an EP-PRG.**
Entropy-preserving PRG (EP-PRG)

Efficiently computable function $G: \{0,1\}^n \rightarrow \{0,1\}^{n+c \log n}$

- **Pseudorandomness**: $G(U_n)$ indistinguishable from $U_{n+c \log n}$
- **Entropy-preserving**: $G(U_n)$ has Shannon entropy $n - O(\log n)$
Conditionally Entropy-preserving PRG (condEP-PRG)

Efficiently computable function $G: \{0,1\}^n \to \{0,1\}^{n+c \log n}$

- **Pseudorandomness:** $G(U_n | E)$ indistinguishable from $U_{n+c \log n}$
- **Entropy-preserving:** $G(U_n | E)$ has Shannon entropy $n-O(\log n)$

For some event $E$

**Lemma:** condEP-PRG with running time $t$ implies $K^t$ is mildly-HOA

Same proof as before works.
Constructing condEP-PRG from OWF

**Lemma**: OWF $\Rightarrow$ cond EP-PRG

**Proof**:
- Use a variant of PRG from *regular OWF* from [HILL’99,Gol’01,YLW’15]
- Show that it satisfies our notion of a cond EP-PRG when using *any OWF*.

$$G(s,r_1,r_2,r_3,i) = r_1,r_2,r_3, \left[\text{Ext}_{r_1}(s)\right]_{i-O(\log n)} \left[\text{Ext}_{r_2}(f(s))\right]_{n-i-O(\log n)} \text{GL}(s,r_3)$$

Shannon Entropy $n - O(\log n)$

Not a PRG. Not EP.
But is a PRG and EP *conditioned* on the event that $(i,s)$ is “good”

“good”: $s$ has regularity $r$ that is “common”, $i = r$
Ensures that extractors work.
Theorem 1
Assume there exists some poly $t(n) > 0$, s.t. $K^t$ is mildly-HOA. Then OWFs exist.

Theorem 2
Assume OWFs exists. Then there exists some poly $t(n) > 0$ s.t. $K^t$ is mildly-HOA.
Main Theorem

For all $\varepsilon > 0$, all poly $t(n) > (1+\varepsilon)n$ OWFs exist iff $K^t$ is mildly-HOA.

First natural avg-case problem characterizing the feasibility of the basic tasks in Crypto (i.e., private-key encryption, digital sigs, PRGs, PRFs, commitments, authentication, ZK...).
Recent Results on $K^t$ and Friends

- [Hirahara’18]: presents a **worst-case to average-case reduction** for $K^t$: $K^t$ is **errorless-HAO** if $K^t$ is **worst-case** hard to approximate. Similar results indep. obtained by [Santhanam’19] w.r.t. a variant of MCSP.

  *Our results to not extend to errorless-HAO...*

- [Ilango-Loff-Oliviera’20]: **Multi-MCSP** is NP-Hard

- [Oliviera-Santhanam]: Hardness magnification for MCSP
Towards the “holy-grail”

NP \[\rightarrow\] Multi-MCSP \[\stackrel{[\text{ILO’20}]}{\rightarrow}\] \(K_{\text{poly}}\) \[\rightarrow\] \(K_{\text{poly}}\) \[\rightarrow\] \(K_{\text{poly}}\) \[\rightarrow\] OWF

Hard for BPP

Hard for BPP

Hard to approx for BPP

Errorless-HOA (one-sided error)

mild-HOA (two-sided error)

[\text{today}]

\[\rightarrow\] Missing implications
Thank You