Part I:

Pseudo-randomness from Hardness

Valentine Kabanets
(Simon Fraser University, Canada)

Pseudorandom Objects:

- expander graphs
- error-correcting codes (ECC)
- incompressible strings (Boolean fns of high circuit complexity)
- pseudorandom generators (PRG)

Insight (1980's): Hardness $\Longleftrightarrow$ PRG

Plan:

- Def'n of PRG
- Yao's "distinguisher into next-bit predictor"
- Hybrid argument
- NW PRG
- Play to Win & Play to Lose (applications)

- PRG

$G : \{0,1\}^\ell \to \{0,1\}^k$ "efficiently" computable

$C : \{0,1\}^k \to \{0,1\}$ $C \in \mathcal{C}$ class of tests

C is ε-fooled by G if

$$\left| \Pr_{z \sim U_k}[C(z)=1] - \Pr_{\sigma \sim U_\ell}[C(G(\sigma))=1] \right| \leq \varepsilon$$

Task: Given a class $\mathcal{C}$ of tests $C : \{0,1\}^k \to \{0,1\}$, construct a PRG $G : \{0,1\}^\ell \to \{0,1\}^k$ that ε-fools all $C \in \mathcal{C}$.

<u>Want:</u> $k >> \ell$ (large stretch)

If G fails to ε-fool some Δ, i.e.,

$$\left| \Pr[\Delta(z)=1] - \Pr[\Delta(G(\sigma))=1] \right| > \varepsilon$$

we call this Δ a ε- distinguisher.

- <u>Toward NW PRG</u>

Let $f : \{0,1\}^n \to \{0,1\}$ be a "hard" function

Define

$$\underbrace{G(x)}_{n} = \underbrace{x, f(x)}_{n+1} \text{ bits}$$

Suppose G is <u>not</u> a PRG for a class $\mathcal{C}$ of tests.

Then $\exists$ ε- distinguisher Δ :

$$\Pr_{x \sim U_n}[\Delta(G(x))=1] - \Pr_{z \sim U_{n+1}}[\Delta(z)=1] > \varepsilon$$

$$\Pr_{x \sim U_n}\left[\Delta(x, f(x))=1\right] - \Pr_{\substack{x \sim U_n \\ b \sim U_1}}\left[\Delta(x, b)=1\right] > \varepsilon$$

Algo A^{δ} to compute $f(x)$ [Yao]:

pick $r \sim U_1$
if $\delta(x, r) = 1$ then output r
else output $\neg r$

Claim: $\Pr_{x}\left[A^{\delta}(x) = f(x)\right] - \frac{1}{2}$

$= \Pr_{x}\left[\delta(x, f(x)) = 1\right] - \Pr_{x,b}\left[\delta(x,b) = 1\right]$

Proof: condition on each $x \in \{0,1\}^n$

- Case 1: $\delta(x,0) = \delta(x,1)$
- Case 2: $\delta(x,0) = \neg\delta(x,1)$ $\square$

So, $\Pr_{x,A}\left[A(x) = f(x)\right] \geq \frac{1}{2} + \varepsilon$

Upshot: f is ave-case hard for $\mathcal{C}$
$\left(\forall C \in \mathcal{C} \quad \Pr_{x}\left[C(x) = f(x)\right] < \frac{1}{2} + \frac{\varepsilon}{2} \right)$

$\Rightarrow$ $G(x) = x, f(x)$
ε-fools all δ s.t. $A^{\delta} \in \mathcal{C}$

Ave-case Hardness for $\mathcal{C}$ $\Rightarrow$
Pseudorandomness for a slightly smaller class $\{D\}$

Example: $\mathcal{C} = \text{Size}[2^{\frac{n}{10}}]$
$\{D\} = \text{Size}[2^{\frac{n}{20}}]$

$G(x) = x, f(x)$ $\quad n \longmapsto n+1$ bits only!

Want: poly or exp stretch

Attempt 1 (direct product of f):

$$G(x_1, \dots, x_k) = x_1, \dots, x_k, f(x_1), \dots, f(x_k)$$

$k \cdot n \longmapsto kn + k$ bits

Suppose G is not a PRG: $\exists\, D$ $\exists\, \varepsilon > 0$

$$\Pr[D(G(x_1,\dots,x_k))=1] - \Pr[D(U_{kn+k})=1] > \varepsilon$$

$$\begin{array}{llllll}
\vec{x} & f(x_1) \dots & f(x_{k-1}) & f(x_k) & =: Q_0 & = G \\
\vec{x} & f(x_1) \dots & f(x_{k-1}) & U_1 & =: Q_1 & \\
\vec{x} & f(x_1) \dots & U_1 & U_1 & =: Q_2 & \\
 & & \vdots & & & \\
\vec{x} & U_1 \dots & U_1 & U_1 & =: Q_k & = U_{kn+k}
\end{array}$$

$$\Pr[D(Q_0)=1] - \Pr[D(Q_1)=1]$$
$$+ \Pr[D(Q_1)=1] - \Pr[D(Q_2)=1]$$
$$+ \quad \cdots$$
$$\Pr[D(Q_{k-1})=1] - \Pr[D(Q_k)=1] > \varepsilon$$

By averaging, $\exists\ 0 \le i < k$

$$\Pr[D(Q_i)=1] - \Pr[D(Q_{i+1})=1] > \frac{\varepsilon}{k}$$

Say $i=0$:

$$\Pr[D(x_1,\dots,x_k, f(x_1),\dots,f(x_{k-1}), f(x_k))=1]$$
$$- \Pr[D(x_1,\dots,x_k, f(x_1),\dots,f(x_{k-1}), b)=1] > \frac{\varepsilon}{k}$$

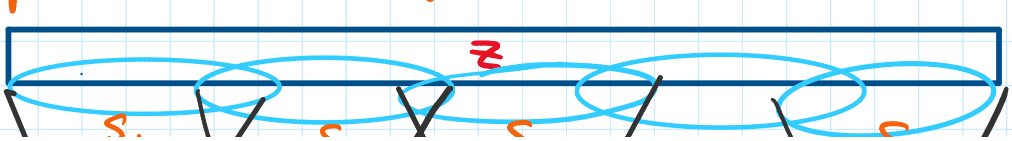
can fix & preserve $> \frac{\varepsilon}{k}$ (by averaging)

By Yao, get a predictor A^{D} (with advice)

$$\Pr_x[A^{D}(x)=f(x)] \ge \frac{1}{2} + \frac{\varepsilon}{k}$$

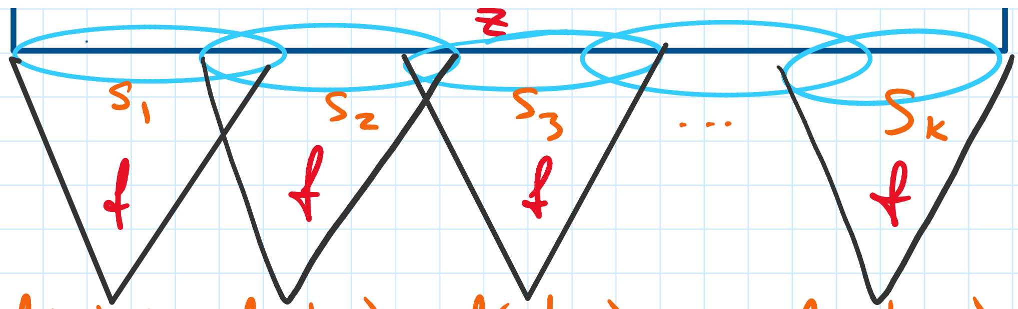
Note: need a "harder" f to get a better stretch.

attempt 2 (NW Designs):



$z \in \{0,1\}^{\ell}$

$z \leftarrow \{0,1\}^{\ell}$



$$f(z|_{S_1}) \quad f(z|_{S_2}) \quad f(z|_{S_3}) \quad \cdots \quad f(z|_{S_k}) \quad =: NW^{f}_{S_1,\dots,S_k}(z)$$

<u>Thm</u>: $\forall \gamma > 0$, $\forall n$ $\exists$ (n,d)-design $S_1,\dots,S_k$

- each $S_i \subseteq [\ell]$, $|S_i| = n$
- $|S_i \cap S_j| \leq d = \gamma \cdot \log k$

with $\ell \leq O\left(\frac{n^2}{d}\right)$. Can be constructed in time $\mathrm{poly}(k,\ell)$ (or $2^{O(\ell)}$).

<u>Example</u>: $k = 2^{\gamma n}$, $\ell \leq O(n)$

$NW^f : O(n) \longmapsto 2^{\gamma n}$ bits

<u>Security analysis</u>: Say D is an ε-distinguisher

$$\Pr_z\left[D(z, f(z|_{S_1}),\dots, f(z|_{S_k})) = 1\right] -$$
$$\Pr_{z,\vec{b}}\left[D(z, b_1, \dots, b_k) = 1\right] > \varepsilon$$

Hybrid argument $\Rightarrow$ $\exists$ $0 \leq i \leq k$

$$\Pr\left[D(z, f(z|\ \ ) \quad f(z|\ \ ) \quad b_1 \quad b_{k-i}) = 1\right] -$$

$$\Pr[\Delta(z, f(z|_{S_1}), \ldots, f(z|_{S_i}), b_1, \ldots, b_{k-i}) = 1] -$$
$$\Pr[\Delta(z, f(z|_{S_1}), \ldots, b, b_1, \ldots, b_{k-i}) = 1]$$
$$> \varepsilon/k$$

- fix $z|_{\overline{S_i}}$ (z outside S_i)
- $x := z|_{S_i}$ is free.
- $f(z|_{S_1}), \ldots, f(z|_{S_{i-1}})$ almost fixed

  each depends on $\leq d = \gamma n$ bits of x,

  $2^{\gamma n}$ bits of advice suffice

Total advice: $\leq 2^{(3\gamma)n} \ll 2^n$ if $\gamma \ll \frac{1}{3}$

Upshot: If $f : \{0,1\}^n \to \{0,1\}$ cannot be computed on more than $\frac{1}{2} + \frac{\varepsilon}{2^{\gamma n}}$ inputs by circuits of size $2^{3\gamma n}$, then $NW^f_{S_1, \ldots, S_{2^{\gamma n}}} : O(n) \to 2^{\gamma n}$ is PRG for linear-size circuits.

Rescaling: $G : O(\log n) \mapsto n$ bits

Rescaling: $G : O(\log n) \mapsto n$ bits

secure against linear-size tests

If $f \in E = \mathrm{Time}(2^{O(n)})$, then $G^f : O(\lg n) \mapsto n$ is computable in poly(n) time.

Thm [Nisan-Wigderson]: $\exists f \in E$ that is exp-hard on average by exp-size circuits, then

$\exists$ PRG $G^f : O(\lg n) \mapsto n$

computable in poly(n) time.

Worst-Case to Average-Case Reduction via locally list-decodable ECCs

$$f^{\in E} : \{0,1\}^n \to \{0,1\}$$

$$\Big\downarrow \text{ECC}$$

$$\hat{f}^{\in E} : \{0,1\}^{O(n)} \to \{0,1\}$$

(Reed-Muller + Hadamard)

s.t.

if can compute $\hat{f}$ on $\frac{1}{2}+\varepsilon$ inputs

s.t. if can compute $\hat{f}$ on $\frac{1}{2}+\varepsilon$ inputs with size s circuits
then can compute f everywhere with size $\mathrm{poly}(s, \frac{1}{\varepsilon})$ circuits

Thm [Impagliazzo-Wigderson]: If E requires size $2^{\Omega(n)}$ circuits, ~~then~~ $\Longleftrightarrow$ $\exists$ PRG $G: O(\lg n) \mapsto n$.

for size(n) circuits

Hence, BPP = P.

Play to Win

Play to Lose

$$G^f : \{0,1\}^\ell \to \{0,1\}^k$$

$k \ll$ circuit size(f)

$\Downarrow$

G^f can't be broken

Derandomization

$k \gg$ circuit size(f)

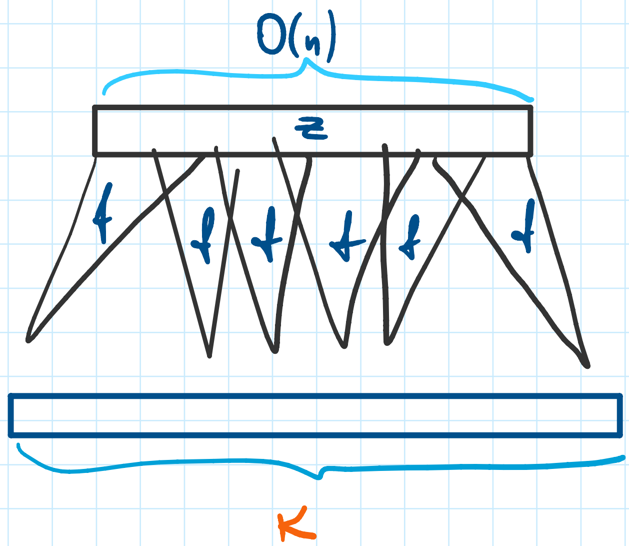
$\Downarrow$

G^f can be broken if can compute MCSP

Learning from MCSP (Natural Property)

- all easy fns, ACCEPT
- $\geq \frac{1}{2}$ of random fns, REJECT

$f:\{0,1\}^n \to \{0,1\}$ $\quad size(f) = 2^{\varepsilon n}$



$g:\{0,1\}^{\log K} \to \{0,1\}$

$K = 2^{\frac{\varepsilon}{10} n}$

$K = 2^n$, $g:\{0,1\}^n \to \{0,1\}$

$size(g) \lesssim size(f) = 2^{\varepsilon n}$

but, $size(rand_n) \approx 2^n$

So, $MCSP(-, 2^{\varepsilon n})$ breaks G^f!

$\Rightarrow$ Can learn f by the Reconstruction Property of NW

Reconstruction Property of NW
[Carmosino, Impagliazzo, Kabanets, Kolokolová]

Min Circuit Size Problem (MCSP): Given x, s, is x computable by a circuit of size $\leq s$?

Part II :

Meta-Complexity

MCSP : Given x, s, is x computable by a circuit of size $\leq s$?

MK^tP : Given x, s, t, is there $d \in \{0,1\}^{\leq s}$ s.t. UTM $U(d)$ outputs x within t steps?

What is the computational complexity of MCSP and MK^tP ???

- both are in NP. Are they NP-complete?
- are they easy on average? (No, if Crypto exists [Razborov, Rudich])

Connections to Crypto, Learning, Complexity, ...

Pseudorandomness is an important tool for Meta-Complexity!

<u>Plan</u>:

- Hadamard Code DP Generator (Direct Product)
- Symmetry of Information for K^{poly} (under assumptions)
- Application to Complexity Theory

<u>Hadamard Code</u>

message $x \in \{0,1\}^n \longmapsto \langle x, 0^n\rangle \langle x, 0^{n-1}1\rangle \dots \langle x, 1^n\rangle$

$\left[\langle x,y\rangle = \sum x_i y_i \mod 2 \right]$

<u>Thm</u> [Goldreich, Levin]

$\exists$ poly-time (randomized) algo A that given n, ε

Thm [illegible]

$\exists$ polytime (randomized) algo $\mathcal{A}$ that, given n, ε and $C : \{0,1\}^n \to \{0,1\}$ s.t., for some $x \in \{0,1\}^n$,

$$\Pr_r [C(r) = \langle x, r \rangle] \geq \frac{1}{2} + \varepsilon ,$$

the algo $\mathcal{A}^C$ outputs (w. prob. $\geq \frac{1}{2}$) a list of $O(\frac{1}{\varepsilon^2})$ strings that contains x (in time $O(\frac{n^2}{\varepsilon^4} \cdot \log n)$).

DP Generator $DP_k : \{0,1\}^{n+nk} \to \{0,1\}^{nk+k}$

$$DP_k(x, z_1, \dots, z_k) = z_1 \dots z_k \langle x, z_1 \rangle \dots \langle x, z_k \rangle$$

($k \leq \text{poly}(n)$)

DP Reconstruction

Lemma [Hirahara]: Suppose E requires size $2^{\Omega(n)}$ circuits. $\exists$ poly p s.t. if some time t algo D $(1/3)$-distinguishes $DP_k(x, \vec{z})$ from $\mathcal{U}_{nk+k}$, then

$$K^{p(t)}(x) \leq k + \log p(t) .$$

Proof:

$$z_1 \dots z_k \ \langle x, z_1 \rangle \dots \langle x, z_k \rangle \quad = DP_k$$

$$z_1 \dots z_k \ \langle x, z_1 \rangle \dots \ b_k$$

$z_1 \dots z_K \quad \langle x, z_1\rangle \dots \langle x, z_K\rangle$

$z_1 \dots z_K \quad \langle x, z_1\rangle \dots \quad b_K$

$z_1 \dots z_K \quad \langle x, z_1\rangle \dots b_{K-1} \ b_K$

$\dots$

$z_1 \dots z_K \quad b_1 \dots b_{K-1} \ b_K \quad = \mathcal{U}_{nK+K}$

δ

$\left(\frac{1}{3}\right)$- dist.

By Yao, get a Predictor for $\langle x, z_K\rangle$, for $\geq \frac{1}{2} + \frac{1}{3K}$ inputs

pK

with advice $\langle x, z_1\rangle, \dots, \langle x, z_{K-1}\rangle$

dependent on randomness $\underbrace{z_1, \dots, z_K \in \{0,1\}^{nK}}_{\text{too big to add to advice}}$

Solution: Use the NW PRG $G : O(\log n) \mapsto nK$ bits

(which exists by circuit complexity for E assumption)

to replace $z_1 \dots z_K$ with $G(\sigma)$ for $\sigma \in \{0,1\}^{O(\log n)}$

Add $\sigma \in \{0,1\}^{O(\log n)}$ to the advice.

Fixed $\sigma \Rightarrow$ fixed $z_1, \dots, z_K \Rightarrow$ fixed $\underbrace{\langle x, z_1\rangle, \dots, \langle x, z_K\rangle}_{\text{extra } K \text{ bits of advice}}$

Then by GL run on Yao's Predictor, get a list of size $O(K^2)$ containing x.

Use additional $O(\log K) \leq O(\log n)$ bits of advice to specify this x on the list.

Conclude: $\approx$ runtime of GL's algo on Yao's Predictor

$$K^{p(t)}(x) \leq K + \log p(t)$$

$\square$

$K \quad (x) \leq K + \log p(t)$ $\square$

Symmetry of Information

Thm [Hirahara; Goldberg, K.]:

Suppose E requires size $2^{\Omega(n)}$ circuits. $\Leftrightarrow$ PRG $G: O(\log n) \mapsto n$ bits, secure for size(n).

Suppose MK^tP is <u>easy on average</u> (in AvgP). (defined in the proof below)

Then $\exists$ poly q s.t.

$\forall\ x, y \in \{0,1\}^*$ and large t,

we have

$$K^t(x,y) \geq K^{q(t)}(x) + K^{q(t)}(y \mid x) - \log q(t)$$

<u>Remark</u>: SoI for $K^t \Rightarrow$ no Crypto (no OWF).
Hence, unconditional SoI for K^t is unlikely.

$\forall$ polytime $F: \{0,1\}^n \to \{0,1\}^n$ (candidate OWF/permutation)

$$\begin{aligned}
K^{q(t)}(x \mid \underset{y}{\underset{\shortparallel}{F(x)}}) &\leq K^t(x,y) - K^{q(t)}(F(x)) + O(\log n) \\
&\leq K^{p(t)}(x) - K^{q(t)}(F(x)) + O(\log n) \\
&\leq O(\log n) \quad \text{for most } x \sim U_n
\end{aligned}$$

$\leq O(\log n)$ for most $x \sim U_n$

Proof: For any k, k' (we'll choose their values later)

$$K^{2t}(\mathcal{D}P_k(x,z), \mathcal{D}P_{k'}(y,z')) \leq \underbrace{K^t(x,y) + |z| + |z'| + \log t}_{=: s}$$

Say a polytime algo B solves $MK^{2t}P$ on average:

For u, w, s such that $|u| + |w| > s$,

- $\forall\, u, w$ if $K^{2t}(u,w) \leq s$
  then $B(u, w, 1^{2t}, 1^s) = 1$
- $\Pr_{u,w \sim U}[B(u,w,1^{2t},1^s) = 1] \leq \frac{1}{2}$

Claim: For k, k' s.t. $k + k' = K^t(x,y) + \log t + 2$
and for $s = K^t(x,y) + |z| + |z'| + \log t$,

$$B(-,-,1^{2t},1^s)$$

is a $(1/2)$-Distinguisher for $\mathcal{D}P_k(x,z) \circ \mathcal{D}P_{k'}(y,z')$.

Proof of Claim: By def'n of S, B accepts <u>all</u> outputs of $\Delta P_K(x,z) \circ \Delta P_{K'}(y,z')$ (w. prob. 1 over z,z')

The output length of $\Delta P_K \circ \Delta P_{K'}$ is

$$K + K' + |z| + |z'| = S + 2 > S$$

Play to Lose

$$\Pr_{u,w \sim \mathcal{U}}\left[B(u,w,1^{2t},1^S)=1\right] \le \frac{1}{2} \quad (\text{by def'n of } B) \diamond$$

We'll choose K so that B <u>DOES NOT</u> $\frac{1}{4}$-distinguish between

$$\Delta P_K(x,z) \circ \mathcal{U} \quad \text{and}$$
$$\mathcal{U} \circ \mathcal{U}$$

Suppose B <u>does</u> $\frac{1}{4}$-distinguish them.
By ΔP Reconstruction Lemma, for some p,

$$(*) \qquad K^{p(t)}(x) \le K + \log p(t).$$

Set $K = K^{p(t)}(x) - \log p(t) - 1$ so that $(*)$ is false!

Play to Win

$$\left.\begin{array}{l} \mathcal{U} \circ \mathcal{U} \\ \end{array}\right\} \text{indistinguishable by } B$$

indistinguishable by B

distinguish. by B

$$\Delta P_{\kappa}(x,-) \circ \mathcal{U}$$

$$\Delta P_{\kappa}(x,-) \circ \Delta P_{\kappa'}(y,-)$$

must be distinguishable by B (Δ-inequality)

By ΔP Reconstruction for some p'

$$\begin{aligned} K^{p'(t)}(y \mid x) &\le \kappa' + \log p'(t) \\ &= K^t(x,y) + \log t + 2 - \kappa \\ &= K^t(x,y) + \log t + 2 \\ &\quad - K^{p(t)}(x) + \log p(t) + 1 \end{aligned}$$

For large enough $q \gg \max\{p, p'\}$, get

$$K^{q(t)}(y \mid x) \le K^t(x,y) - K^{q(t)}(x) + \log q(t).$$

$\square$

Symmetry of Information for pK^t

No derandomization assumption, but only assume

$$MK^tP \in \mathrm{Avg}\,P$$

$$\Rightarrow \quad pK^{q(t)}(y|x) \leq pK^{t}(x,y) - pK^{q(t)}(x) + \log q(t)$$

Application (cf. Igor's talk yesterday)

__Thm__ [Hirahara]: $\mathrm{Dist}\,NP^{NP} \subseteq \mathrm{Avg}\,P \Rightarrow P^{NP} \subseteq \mathrm{Time}[2^{O(\frac{n}{\log n})}]$.

__Proof Sketch__:

(1) $\mathrm{Dist}\,NP^{NP} \subseteq \mathrm{Avg}\,P \Rightarrow \forall$ NP-verifier $R(x,y)$

$$(*) \qquad K^{t}(x,y_x) \leq K^{t^{\varepsilon}}(x) + \log t$$

where y_x is the lex-first R-witness of x

(for large enough t, and const ε)

(2) By SoI ($\mathrm{Dist}\,NP \subseteq \mathrm{Avg}\,P \Rightarrow$ $E \notin$ io-Size$[2^{o(n)}]$ & $MK^tP \in \mathrm{Avg}\,P$)

$$K^{p(t)}(y_x|x) \leq K^{t}(x,y_x) - K^{p(t)}(x) + \log p(t)$$

(by $*$) $$\leq K^{t^{\varepsilon}}(x) - K^{p(t)}(x) + \log p'(t)$$

(3) By a simple averaging argument,
$\exists\ t \le 2^{o(n/\log n)}$

$$K^t(y_x | x) \le O\left(\frac{n}{\log n}\right)$$

Finally, enumerate all $O\left(\frac{n}{\log n}\right)$-bit candidate descriptions to look for y_x. $\square$

Remark: To get $P^{NP} \subseteq \mathrm{Time}\left[2^{O\left(\frac{n}{\log n}\right)}\right]$,
it suffices to assume:

(1) $E \not\subseteq$ io-$\mathrm{Size}[2^{o(n)}]$

(2) $MK^tP \in \mathrm{Avg}P$

(3) $MK^{t,SAT}P \in \mathrm{Avg}P$

} distinguish Easy from Random

This may not be exploiting fully the assumption $\mathrm{Dist}NP^{NP} \subseteq \mathrm{Avg}P$...

Claim: $\mathrm{Dist}NP^{NP} \subseteq \mathrm{Avg}P \Rightarrow \forall$ NP-verifier $R(x,y)$

$$K^t(x, y_x) \le K^{t^\varepsilon}(x) + \log t$$

$$K(x, y_x) \leq K(x) + \log t$$

where y_x is the lex-first R-witness of x for large enough t, and const ε)

<u>Proof sketch</u>: Play to Lose. $K^{2t,\,\mathrm{SAT}}(x, y_x) \lesssim K^t(x)$

$$K^{3t,\,\mathrm{SAT}}(\,\mathrm{DP}_k(x, y_x ; z)) \lesssim K^{2t,\,\mathrm{SAT}}(x, y_x) + |z|$$

$$\lesssim \underbrace{K^t(x) + |z|}_{=s}$$

Set k so that $k + |z| = s + 2$.

Then AvgP-algo for $\mathrm{MK}^{3t,\,\mathrm{SAT}}\mathrm{P}$ breaks the generator $\mathrm{DP}_k(x, y_x ; z)$.

By DP Reconstruction,

$$K^{p(t)}(x, y_x) \lesssim k \lesssim K^t(x) . \quad \diamond$$

Conclusions

Hardness-based PRG constructions are useful for:

- derandomization (play to win)
- learning (play to lose)
- K^t complexity properties (play to win + play to lose)
- average-case vs. worst-case complexity
  (Heuristica vs. Algorithmica)
- More ?