Indistinguishability Obfuscation

and its Connections to Proof Complexity

Christopher Brzuska

Aalto University, Finland

Goals

If your only take-away from this talk is the definition, that's not a bad outcome.

Russell & Ivy

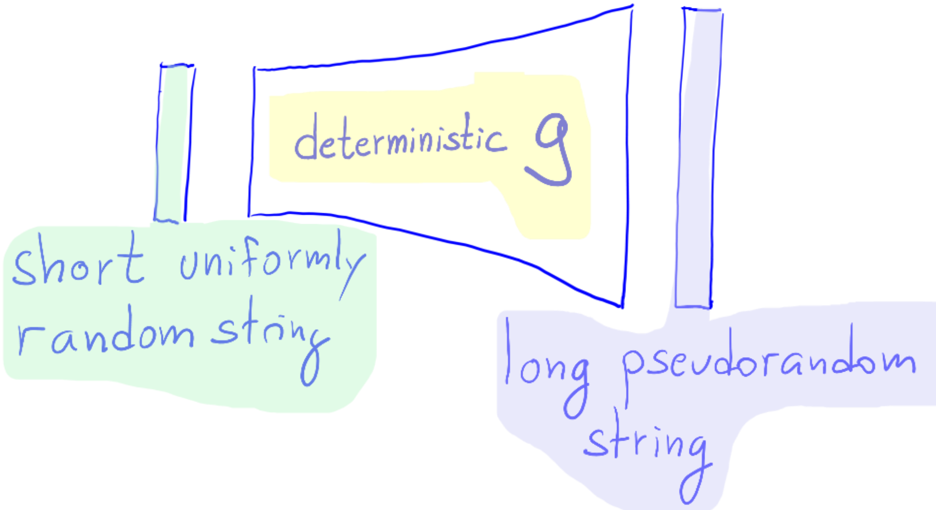
Indistinguishability Obfuscation (iO) – Definition

iO is powerful!

How can iO be useful to you? Lower bounds.

Fun open questions. Not recommendations.

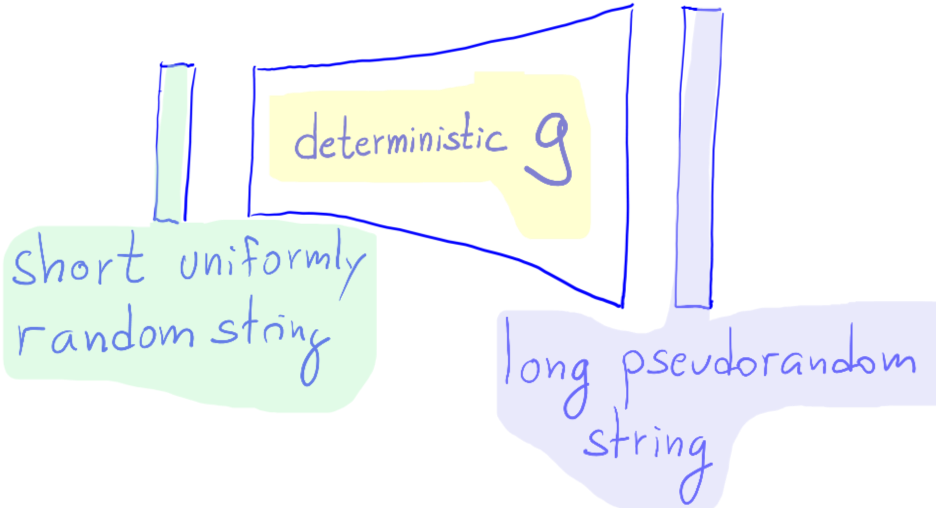
Cryptographic Pseudorandom Generators



For all
polynomial-time "$\mathcal{A}$"

$$\text{“}\mathcal{A}\text{”}(g(x)) \approx \text{“}\mathcal{A}\text{”}(y)$$

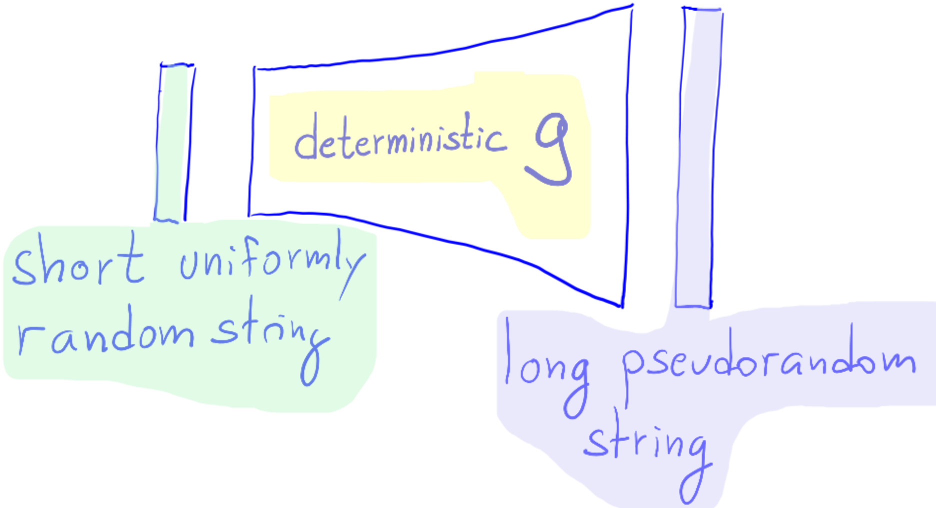
Cryptographic Pseudorandom Generators



For all polynomial-time "$\mathcal{A}$"

$$\Pr\left[1 = \text{"}\mathcal{A}\text{"}(g(x))\right] \approx \text{"}\mathcal{A}\text{"}(y)$$

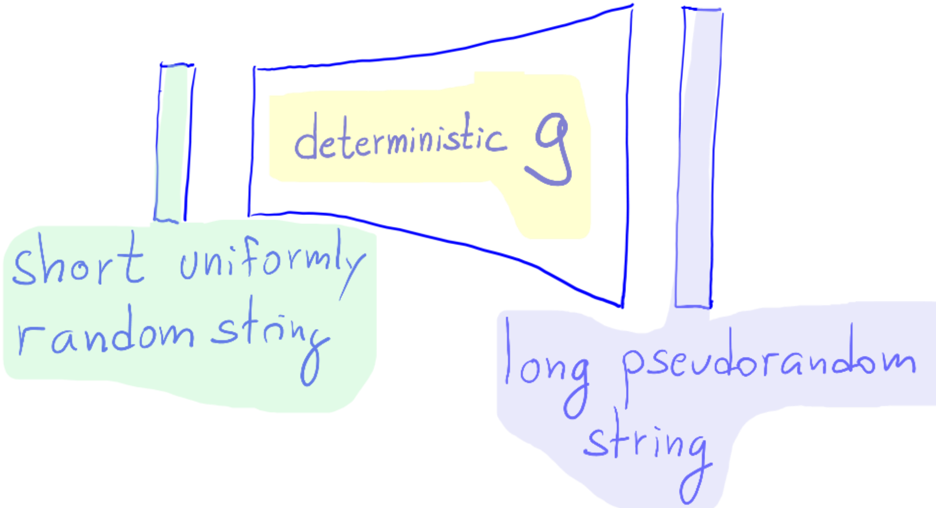
Cryptographic Pseudorandom Generators



For all polynomial-time "$\mathcal{A}$"

$$\Pr[1 = \text{“}\mathcal{A}\text{”}(g(x))] \approx \Pr[1 = \text{“}\mathcal{A}\text{”}(y)]$$

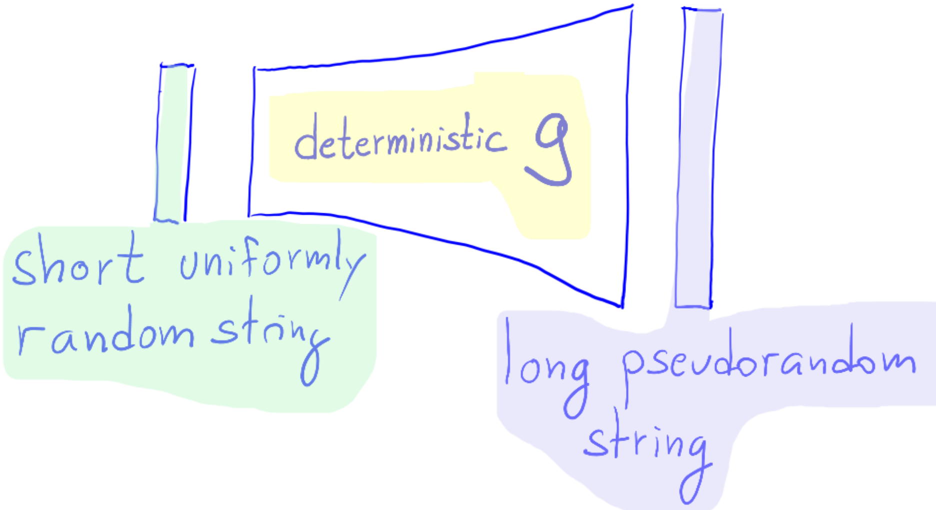
Cryptographic Pseudorandom Generators



For all polynomial-time "$\mathcal{A}$"

$$\Pr_{x \xleftarrow{\$} \{0,1\}^n}\left[1 = \text{"}\mathcal{A}\text{"}(g(x))\right] \approx \Pr_{y \xleftarrow{\$} \{0,1\}^{2n}}\left[1 = \text{"}\mathcal{A}\text{"}(y)\right]$$

Cryptographic Pseudorandom Generators

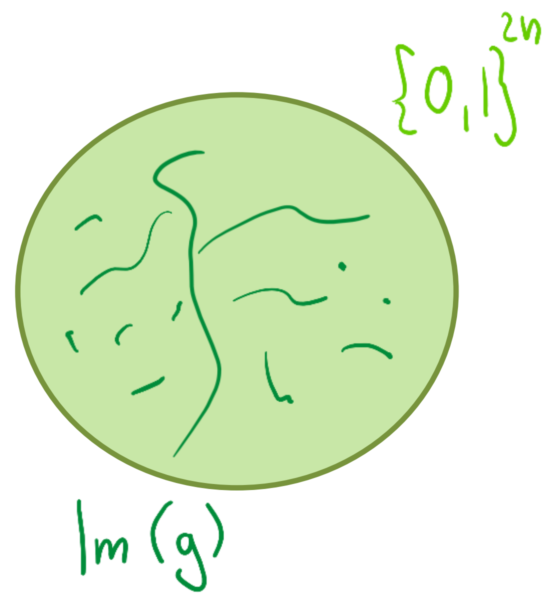


For all polynomial-time "A"

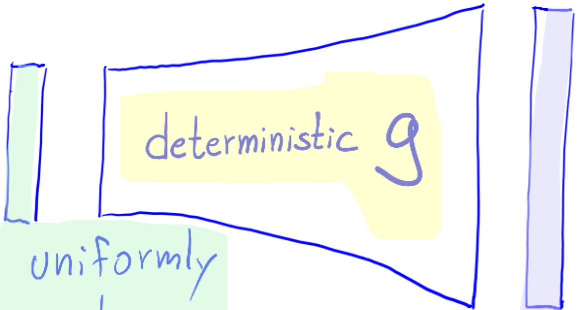
$$\Pr\left[1 = \text{"}A\text{"}(g(x))\right] \approx \Pr\left[1 = \text{"}A\text{"}(y)\right]$$

$x \xleftarrow{\$} \{0,1\}^n$ $\qquad$ $y \xleftarrow{\$} \{0,1\}^{2n}$

Image is a sparse set.



Cryptographic Pseudorandom Generators



short uniformly random string

long pseudorandom string

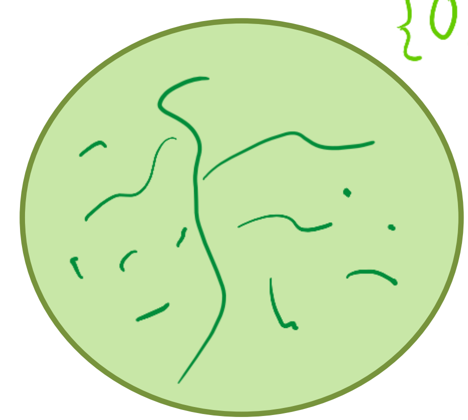
For all probabilistic polynomial-time "A"

$n \longrightarrow \infty$

$$\Pr_{x \xleftarrow{\$} \{0,1\}^n}\left[1 = \text{“}A\text{”}(g(x))\right] \approx \Pr_{y \xleftarrow{\$} \{0,1\}^{2n}}\left[1 = \text{“}A\text{”}(y)\right]$$

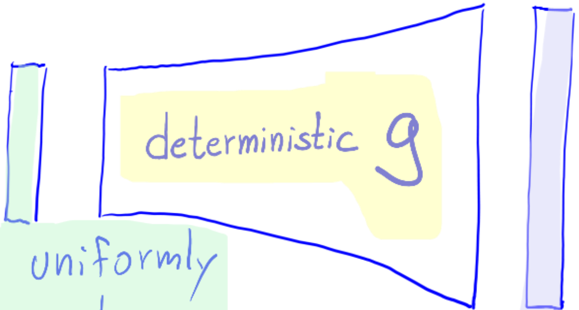
Image is a sparse set.

$\{0,1\}^{2n}$



$\mathrm{Im}(g)$

Cryptographic Pseudorandom Generators



short uniformly random string

long pseudorandom string

Computational indistinguishability

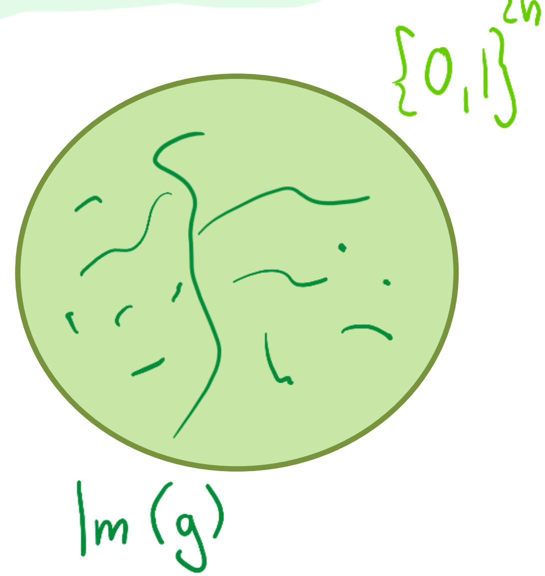
$g(x) \approx y$

For all probabilistic polynomial-time "A"

$n \longrightarrow \infty$

$$\Pr_{x \xleftarrow{\$} \{0,1\}^n}\left[1 = \text{“}A\text{”}(g(x))\right] \approx \Pr_{y \xleftarrow{\$} \{0,1\}^{2n}}\left[1 = \text{“}A\text{”}(y)\right]$$

Image is a sparse set.



Obfuscation

same functionality
hides structure

Program P(.) $\xrightarrow{\text{Obf}}$ P‘(.)

Think of P(.) as a circuit with OR and NAND gates.

Or… …think of P as a C-program and P‘ as an unreadable version of it…

even worse
than before.

…and we want to prove that the obfuscation is “secure“.

Indistinguishability Obfuscation iO

polytime

$\mathrm{Obf}(P; r) = P'$ randomized $P \overset{\text{func}}{\equiv} P'$

Correctness: $\forall P, r, x$: $P(x) = P'(x)$, where $P' = \mathrm{Obf}(P; r)$

Security: $P_0 \overset{\text{func}}{\equiv} P_1 \implies \mathrm{Obf}(P_0) \approx \mathrm{Obf}(P_1)$

shorthand
for distribution
$r \xleftarrow{\$} \{0,1\}^\lambda$
ret. $\mathrm{Obf}(P_0; r)$

Indistinguishability Obfuscation iO

polytime

$\mathrm{Obf}(P; r) = P'$ randomized $P \overset{\text{func}}{\equiv} P'$

Correctness: $\forall P, r, x$: $P(x) = P'(x)$, where $P' = \mathrm{Obf}(P; r)$

Security: $P_0 \overset{\text{func}}{\equiv} P_1 \implies \mathrm{Obf}(P_0) \approx \mathrm{Obf}(P_1)$

$\mathcal{A}(P_0, P_1, \mathrm{Obf}(P_0; r))$

$\mathcal{A}(P_0, P_1, \mathrm{Obf}(P_1; r))$

Indistinguishability Obfuscation iO

polytime

$\mathrm{Obf}(P; r) = P'$ randomized $P \overset{\text{func}}{\equiv} P'$

Correctness: $\forall P, r, x$: $P(x) = P'(x)$, where $P' = \mathrm{Obf}(P; r)$

Security: $P_0 \overset{\text{func}}{\equiv} P_1 \implies \mathrm{Obf}(P_0) \approx \mathrm{Obf}(P_1)$

$$\Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_0; r)\right)\right]$$

$$\approx \Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_1; r)\right)\right]$$

"I must admit that I was very sceptic of the applicability of indistinguishability obfuscation."

Oded Goldreich

$P=NP \Rightarrow iO$ (for program = circuit)

Construction: Obf(C):=
lexicographically first circuit that computes the same function as C.

Security: For all C_0, C_1 that compute the same function: Obf(C_0(.)) and Obf(C_1(.)) are indistinguishable.

Equal!

Efficiency:

Indistinguishability Obfuscation iO

polytime

$\mathrm{Obf}(P; r) = P'$ randomized $P \stackrel{\text{func}}{\equiv} P'$

Correctness: $\forall P, r, x$: $P(x) = P'(x)$, where $P' = \mathrm{Obf}(P; r)$

Security: $P_0 \stackrel{\text{func}}{\equiv} P_1 \implies \mathrm{Obf}(P_0) \approx \mathrm{Obf}(P_1)$

$$\Pr_r\left[1 = \mathcal{A}(P_0, P_1, \mathrm{Obf}(P_0; r))\right]$$

$$\approx \Pr_r\left[1 = \mathcal{A}(P_0, P_1, \mathrm{Obf}(P_1; r))\right]$$

Statistically secure

Indistinguishability Obfuscation iO

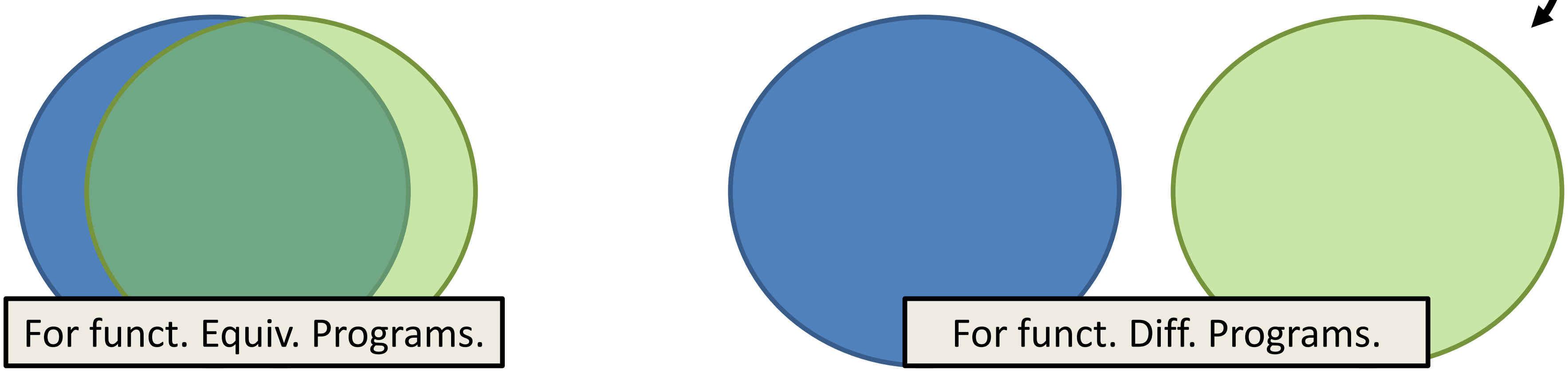
polytime

$Obf(P; r) = P'$ randomized $P \overset{func}{\equiv} P'$

Correctness: $\forall P, r, x:$ $P(x) = P'(x)$, where $P' = Obf(P; r)$

Security: $P_0 \overset{func}{\equiv} P_1 \Longrightarrow Obf(P_0) \approx Obf(P_1)$

These two distributions are close/equal (not just hard to distinguish).



OWF $\Leftarrow$ NP $\not\subseteq$ BPP + (stat. secure) iO

Construction:

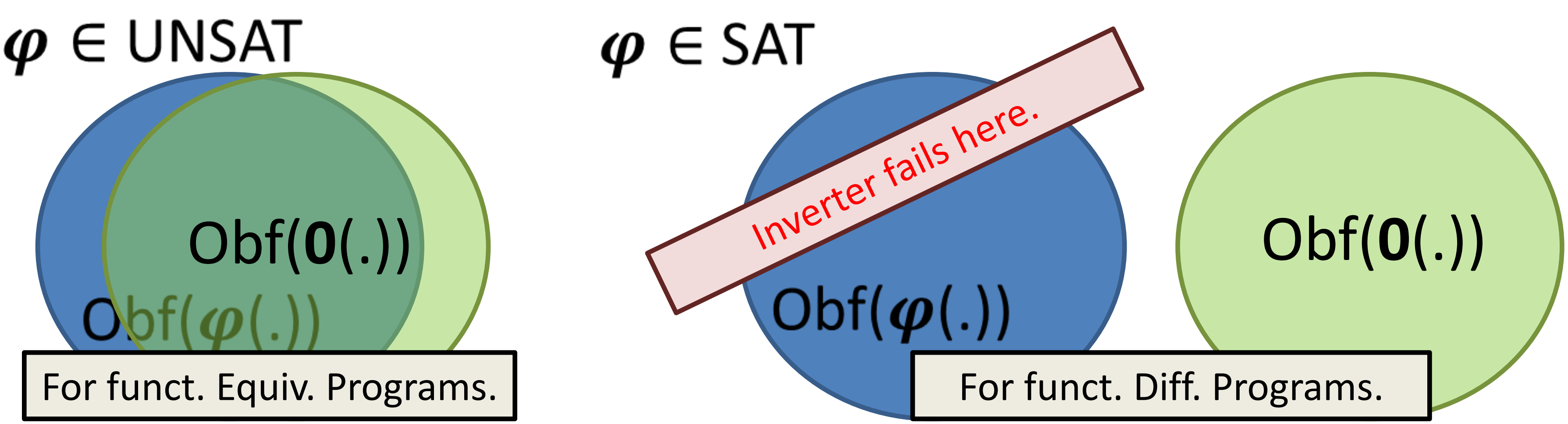
$r \to \text{Obf}(\mathbf{0}(.);r)$

Constant zero function that maps all values to 0.

Randomness of the obfuscator

Why is this an OWF?

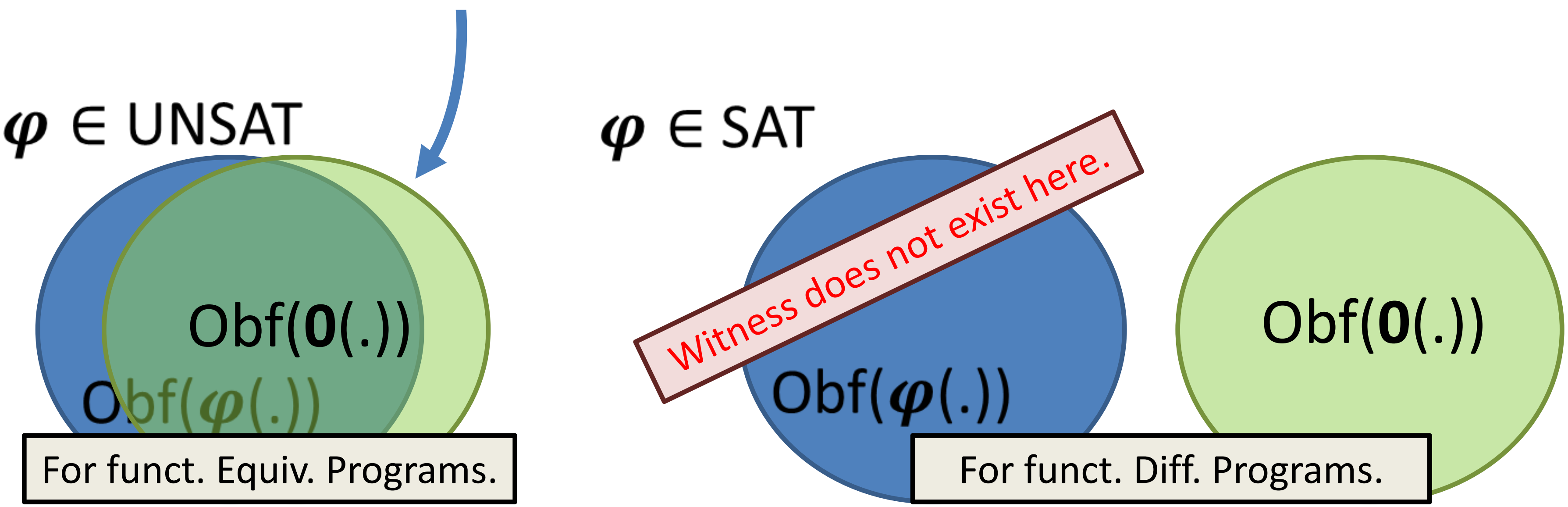
Assume towards contradiction that there exists a polytime inverter…
Goal: distinguish satisfiable from unsatisfiable formulae (& reach contradiction)



No iO with statistical security

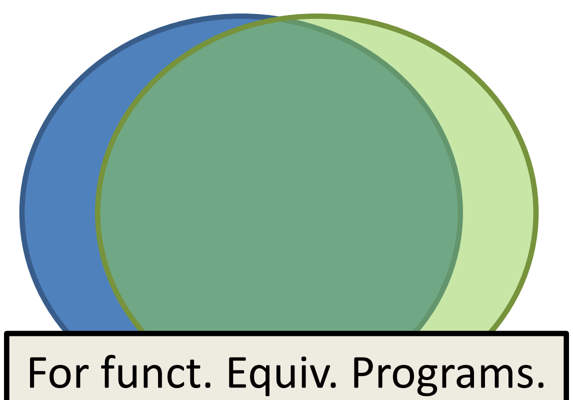
(under complexity assumptions)

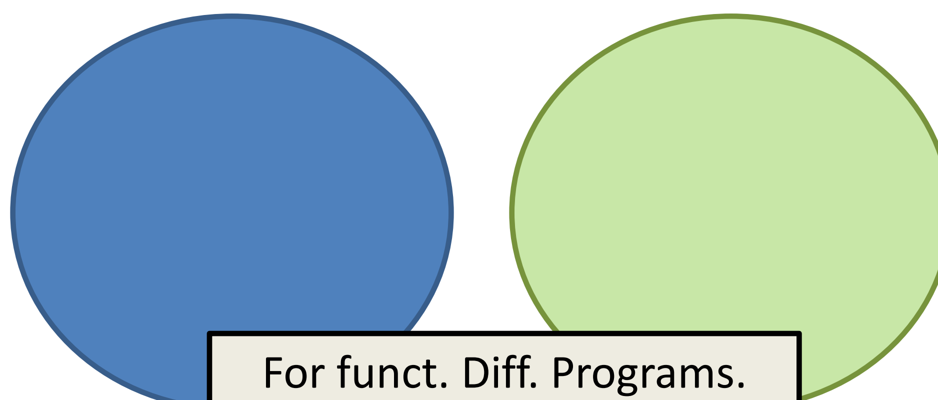
- $\exists$ siO $\Rightarrow$ coNP $\subseteq$ NP



0-functions are important.

- **OWF:** r ↦ Obf(**0**;r)
- **Impossibility:** Considers range of Obf(**0**;r)
- **Lower bounds** in Proof Complexity





Indistinguishability Obfuscation iO

polytime

$\mathrm{Obf}(P; r) = P'$ randomized $\quad P \overset{func}{\equiv} P'$

Correctness: $\forall P, r, x : P(x) = P'(x)$, where $P' = \mathrm{Obf}(P; r)$

Security: $P_0 \overset{func}{\equiv} P_1 \implies \mathrm{Obf}(P_0) \approx \mathrm{Obf}(P_1)$

$$\Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_0; r)\right)\right]$$

$$\approx \Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_1; r)\right)\right]$$

Indistinguishability Obfuscation iO

polytime

$\mathrm{Obf}(P; r) = P'$ randomized $P \overset{\text{func}}{\equiv} P'$

Correctness: $\forall P, r, x$: $P(x) = P'(x)$, where $P' = \mathrm{Obf}(P; r)$

Security: $P_0 \overset{\text{func}}{\equiv} P_1 \implies \mathrm{Obf}(P_0) \approx \mathrm{Obf}(P_1)$

$$\left| \Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_0; r)\right)\right] - \Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_1; r)\right)\right] \right| \leq \delta$$

AVOID

$^{*}m > n$

Input: Expanding* circuit $C: \{0,1\}^n \rightarrow \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

AVOID

$^* m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP $\subseteq$ NP.

This is quite weak, because most $y \notin Im(C)$ still would not have a proof that they're outside $Im(C)$.

2. A **deterministic** poly-time algo. for AVOID would be proof that a certain $y \in \{0,1\}^m$ is not $\notin Im(C)$.

AVOID

$^{*}m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP ⊆ NP.

Proof.

$^*m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

AVOID

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP $\subseteq$ NP.

Proof.

$$\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}](\boldsymbol{x}) := \begin{cases} 0^m & \textbf{if } \psi(x) = 0 \\ y & \textbf{if } \psi(x) = 1 \end{cases}$$

SAT formula

$\in \{0,1\}^m$

obfuscator randomness

Witness for unsatisfiability of ψ:

(y, r) such that:

AVOID(C)=y, where C=Obf($\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}]$;r)

$^*m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

AVOID

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP $\subseteq$ NP.

Proof.

$$\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}](\boldsymbol{x}) := \begin{cases} 0^m & \textbf{if } \psi(x) = 0 \\ y & \textbf{if } \psi(x) = 1 \end{cases}$$

Witness for unsatisfiability of ψ:
(y, r) such that:
AVOID(C)=y, where C=Obf($\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}]$;r)

SAT formula

$\in \{0,1\}^m$

This branch is not used.

correct iO

correctness of AVOID algo

Soundness. Witness (y, r) exists. $\Rightarrow \boldsymbol{y} \notin \boldsymbol{Im}(C)$

$\Rightarrow y \notin Im(C[\psi, y])$

$\Rightarrow \psi$ unsatisfiable

$^{*}m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

AVOID

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP $\subseteq$ NP.

Proof.

$$\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}](\boldsymbol{x}) := \begin{cases} 0^m & \textbf{if } \psi(x) = 0 \\ y & \textbf{if } \psi(x) = 1 \end{cases}$$

SAT formula

$\in \{0,1\}^{\mathrm{m}}$

Witness for unsatisfiability of ψ:
(y, r) such that:
AVOID(C)=y, where C=Obf($\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}]$;r)

Completeness. $\boldsymbol{\psi}$ unsatisfiable. Find (y, r)**.**

Problem. It could be that for every $y \in \{0,1\}^m$, it holds that for every r, y $\neq$ AVOID(C) for C=Obf($C[\psi, y]; r$)

circular dependency

$^{*}m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$
Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

AVOID

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP $\subseteq$ NP.

Proof.

$$\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}](\boldsymbol{x}) := \begin{cases} 0^m & \textbf{if } \psi(x) = 0 \\ y & \textbf{if } \psi(x) = 1 \end{cases}$$

SAT formula

$\in \{0,1\}^m$

Witness for unsatisfiability of ψ:
(y, r) such that:
AVOID(C)=y, where C=Obf($\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}]$;r)

Completeness. $\boldsymbol{\psi}$ unsatisfiable. Find (y, r).

Problem. It could be that for every $y \in \{0,1\}^m$, it holds that for every $r, y \neq$ AVOID(C) for C=Obf($C[\psi, y]; r$)

Idea. Choose $y \in \{0,1\}^m$ such that

$$\Pr_r[\mathrm{y} = \text{AVOID(C)} \mid \text{C=Obf}(\mathbf{0}; r)] \geq 2^{-m}$$

no dependency ☺

$^*m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

AVOID

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP $\subseteq$ NP.

Proof.

$$\boldsymbol{C}[\psi, y](\boldsymbol{x}) := \begin{cases} 0^m & \textbf{if } \psi(x) = 0 \\ y & \textbf{if } \psi(x) = 1 \end{cases}$$

SAT formula

$\in \{0,1\}^m$

Witness for unsatisfiability of ψ: (y, r) such that: AVOID(C)=y, where C=Obf($\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}]$;r)

Completeness. $\boldsymbol{\psi}$ unsatisfiable. Find (y, r).

Idea. Choose $y \in \{0,1\}^m$ such that

$$\Pr_r[y = \text{AVOID(C)} \mid \text{C=Obf}(\mathbf{0}; r)] \geq 2^{-m}$$

Since AVOID needs to output strings in $\{0,1\}^m$, at least one of them needs to be chosen with prob. $\geq 2^{-m}$, so that probabilities add up to 1.

$^{*}m > n$

Input: Expanding* circuit $C: \{0,1\}^n \to \{0,1\}^m$

Goal: Output $y \in \{0,1\}^m$ such that $y \notin Im(C)$.

AVOID

Theorem [Ilango-Li-Williams]: det. polytime algo for AVOID + iO with small enough $\delta \Rightarrow$ coNP $\subseteq$ NP.

Proof.

$$\boldsymbol{C}[\underbrace{\boldsymbol{\psi}}_{\text{SAT formula}}, \underbrace{\boldsymbol{y}}_{\in \{0,1\}^m}](\boldsymbol{x}) := \begin{cases} 0^m & \textbf{if } \psi(x) = 0 \\ y & \textbf{if } \psi(x) = 1 \end{cases}$$

Witness for unsatisfiability of ψ:
(y, r) such that:
AVOID(C)=y, where C=Obf($\boldsymbol{C}[\boldsymbol{\psi}, \boldsymbol{y}]$;r)

Completeness. $\boldsymbol{\psi}$ unsatisfiable. Find (y, r).

Both equivalent.
Both all-zero circuit.

Idea. Choose $y \in \{0,1\}^m$ such that

$$\Pr_r[\mathrm{y} = \mathrm{AVOID(C)} \mid \mathrm{C{=}Obf}(\mathbf{0}; r)] \geq 2^{-m}$$

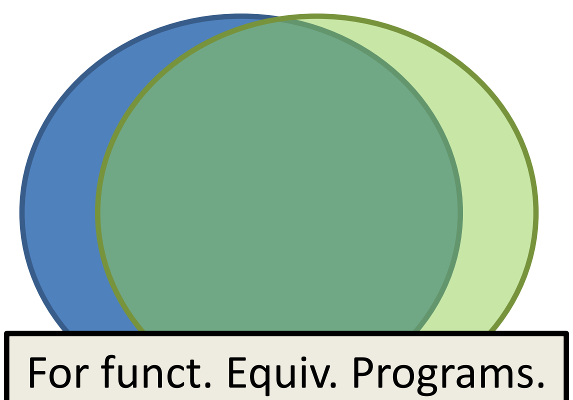
$$\Pr_r[\mathrm{y} = \mathrm{AVOID(C)} \mid \mathrm{C{=}Obf}(C[\psi, y]; r)] > 0$$

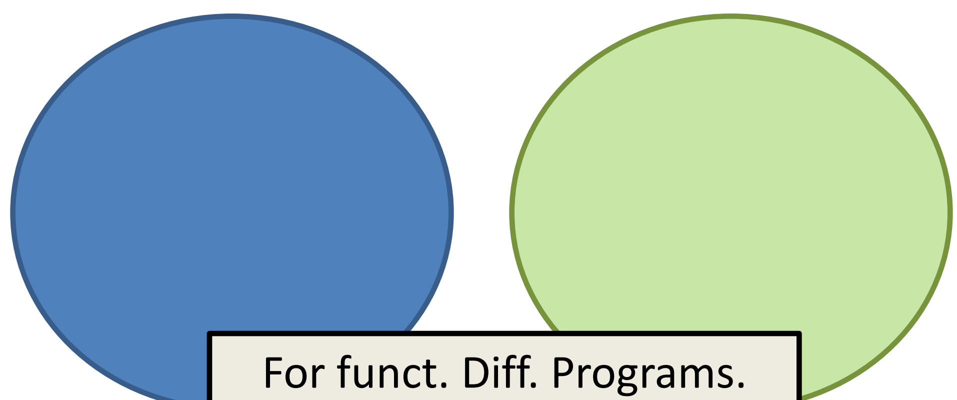
if $\delta < 2^{-m}$

Choose r such that this holds.

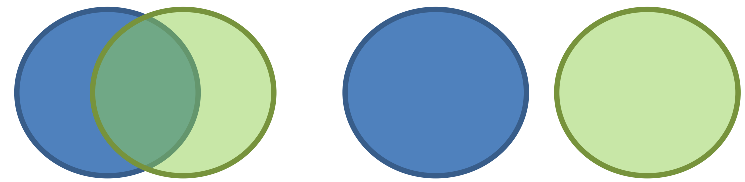
0-functions are important.

- **OWF:** r $\mapsto$ Obf(**0**;r)
- **Impossibility:** Consider range of Obf(**0**;r)
- **Lower bounds** in Proof Complexity





0-functions are important.



- **OWF:** r $\mapsto$ Obf(**0**;r)
- **Impossibility:** Consider range of Obf(**0**;r)
- **Lower bounds** in Proof Complexity

iO for P_0, P_1 with short proof $P_0 \overset{func}{\equiv} P_1$

Impossibility/OWF/Lower bounds: not anymore

Pure crypto applications: still seem to work

iO for Turing Machines: else only for circuits

Better Constructions? (or bounded-input Turing Machines)

Indistinguishability Obfuscation iO

polytime

$\mathrm{Obf}(P; r) = P'$ randomized $P \overset{\text{func}}{\equiv} P'$

Correctness: $\forall P, r, x$: $P(x) = P'(x)$, where $P' = \mathrm{Obf}(P; r)$

Security: $P_0 \overset{\text{func}}{\equiv} P_1 \Longrightarrow \mathrm{Obf}(P_0) \approx \mathrm{Obf}(P_1)$

$$\Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_0; r)\right)\right]$$

$$\approx \Pr_r\left[1 = \mathcal{A}\left(P_0, P_1, \mathrm{Obf}(P_1; r)\right)\right]$$

Thank you!