Bounded Indistinguishability for Simple Sources


Andrej Bogdanov
CUHK


K. Dinesh
CUHK


Yuval Filmus
Technion


Yuval Ishai
Technion


Avi Kaplan
Technion


Akshay Srinivasan
TIFR

Cast of Characters

$X = (X_1, \ldots, X_n), Y = (Y_1, \ldots, Y_n)$ distributions on $\{0,1\}^n$

Cast of Characters

$X = (X_1, \ldots, X_n), Y = (Y_1, \ldots, Y_n)$ distributions on $\{0,1\}^n$

X is k-wise independent if every k coordinates look uniform

Cast of Characters

$X = (X_1, \ldots, X_n), Y = (Y_1, \ldots, Y_n)$ distributions on $\{0,1\}^n$

X is k-wise independent if every k coordinates look uniform

X, Y are k-wise indistinguishable if every k coordinates look the same

Cast of Characters

$X = (X_1, \ldots, X_n), Y = (Y_1, \ldots, Y_n)$ distributions on $\{0,1\}^n$

X is k-wise independent if every k coordinates look uniform

X, Y are k-wise indistinguishable if every k coordinates look the same

X is k-wise independent if X, U are k-wise indistinguishable

$\Uparrow$

uniform
distribution

Examples

Examples

Uniform distribution on even parity vectors: $(n-1)$-wise independent

Examples

Uniform distribution on even parity vectors: $(n-1)$-wise independent

Uniform distribution on subspace is $(k-1)$-wise independent,
where k is dual distance (shortest linear relation)

Examples

Uniform distribution on even parity vectors: $(n-1)$-wise independent

Uniform distribution on subspace is $(k-1)$-wise independent, where k is dual distance (shortest linear relation)

$X = (a_1, b_1, a_1 + b_1, \ldots, a_n, b_n, a_n + b_n)$ is 2-wise independent

Examples

Uniform distribution on even parity vectors: $(n-1)$-wise independent

Uniform distribution on subspace is $(k-1)$-wise independent, where k is dual distance (shortest linear relation)

$X = (a_1, b_1, a_1 + b_1, \ldots, a_n, b_n, a_n + b_n)$ is 2-wise independent

$X|_{a_1 + \cdots + a_n = 0}$ and $X|_{a_1 + \cdots + a_n = 1}$ are $(n-1)$-wise indistinguishable

Motivation

Motivation

k-wise independence: **derandomization**

Motivation

k-wise independence: **derandomization**

k-wise indistinguishability: **secret sharing schemes**

Motivation

k-wise independence: **derandomization**

k-wise indistinguishability: **secret sharing schemes**

any r parties can recover secret

no k keys leak *any* information

Motivation

k-wise independence: **derandomization**

k-wise indistinguishability: **secret sharing schemes**

any r parties can recover secret

no k keys leak *any* information

k-wise independent secret sharing schemes use linear reconstruction

AC^0 reconstruction requires k-wise indistinguishability

Motivation

k-wise independence: **derandomization**

k-wise indistinguishability: **secret sharing schemes**

any r parties can recover secret

no k keys leak *any* information

k-wise independent secret sharing schemes use linear reconstruction
AC^0 reconstruction requires k-wise indistinguishability

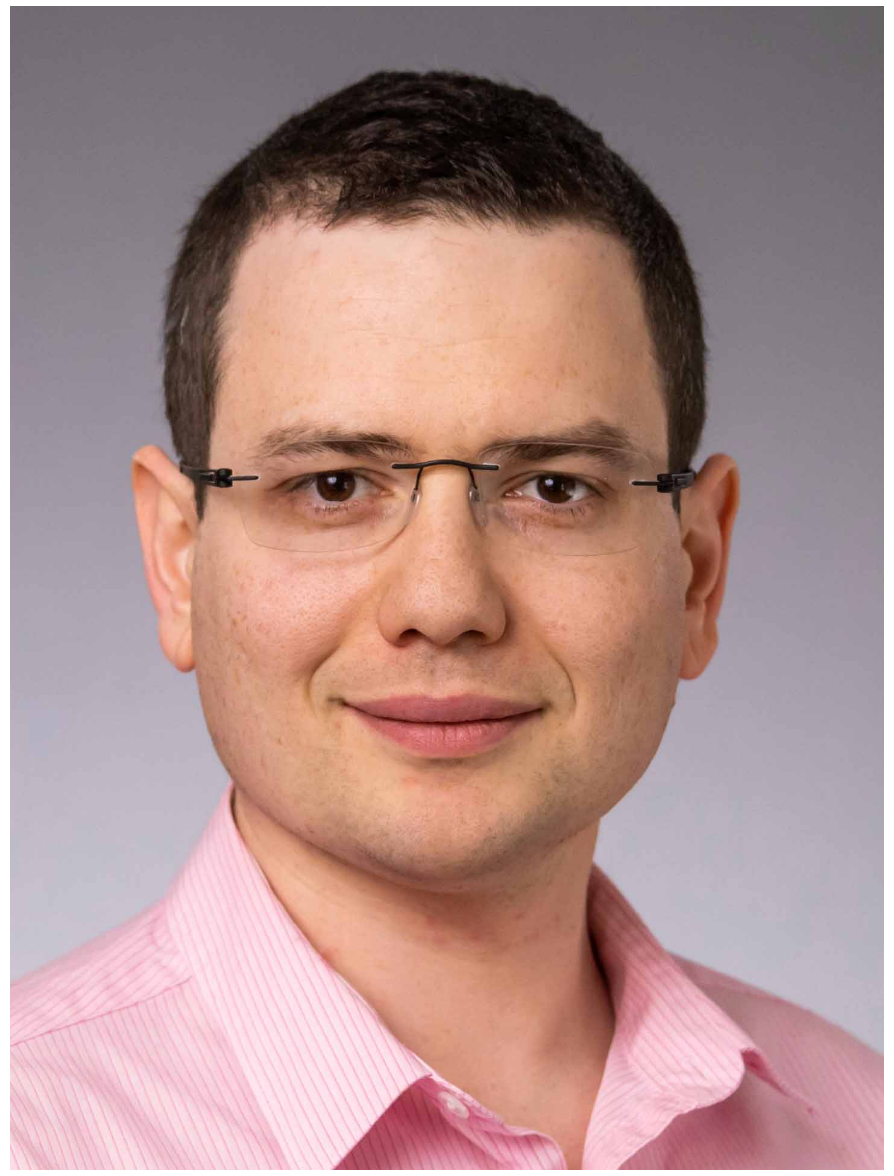
secure multiparty computation and **leakage-resilience** require share manipulation breaks k-wise independence but not k-wise indistinguishability

Braverman for indistinguishability?

[Bogdanov–Ishai–Viola–Williamson 2016]

Braverman for indistinguishability?

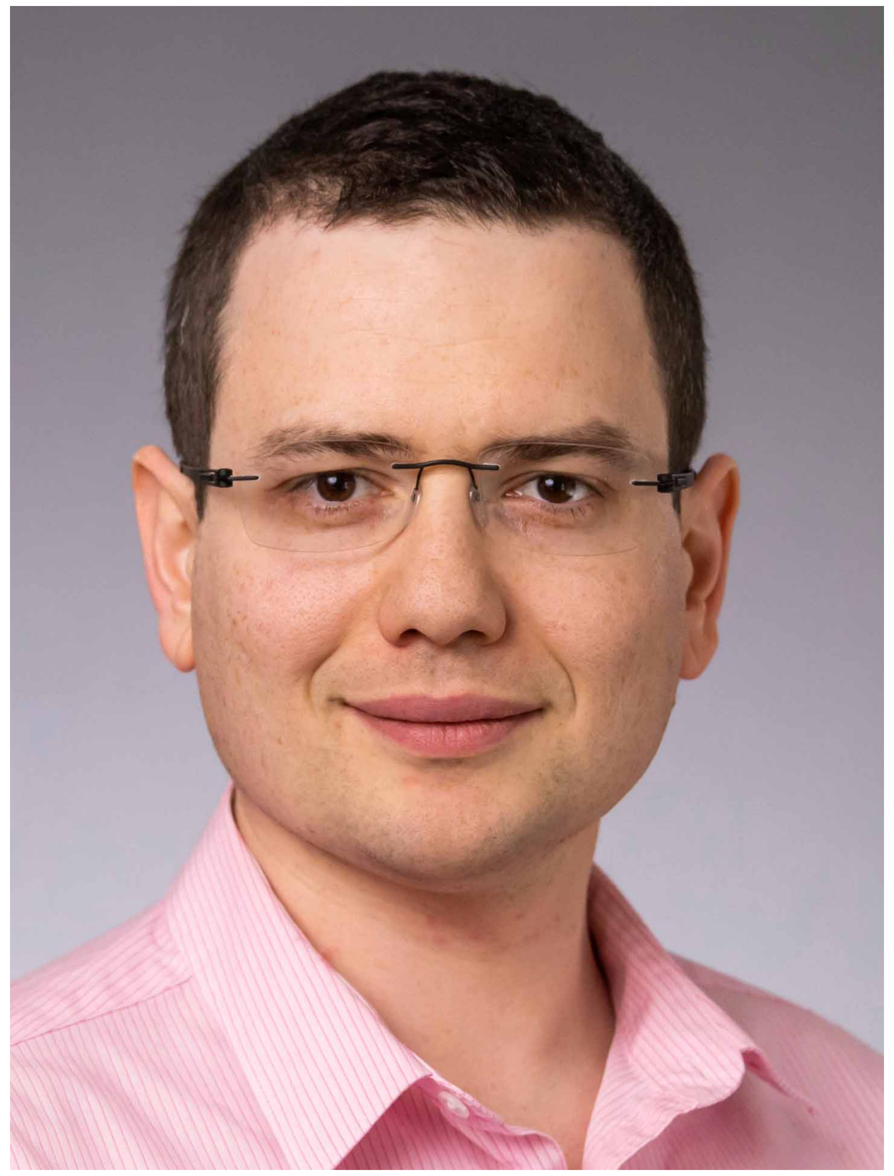
[Bogdanov–Ishai–Viola–Williamson 2016]



Braverman's theorem:
polylog independence
fools AC^0

Braverman for indistinguishability?

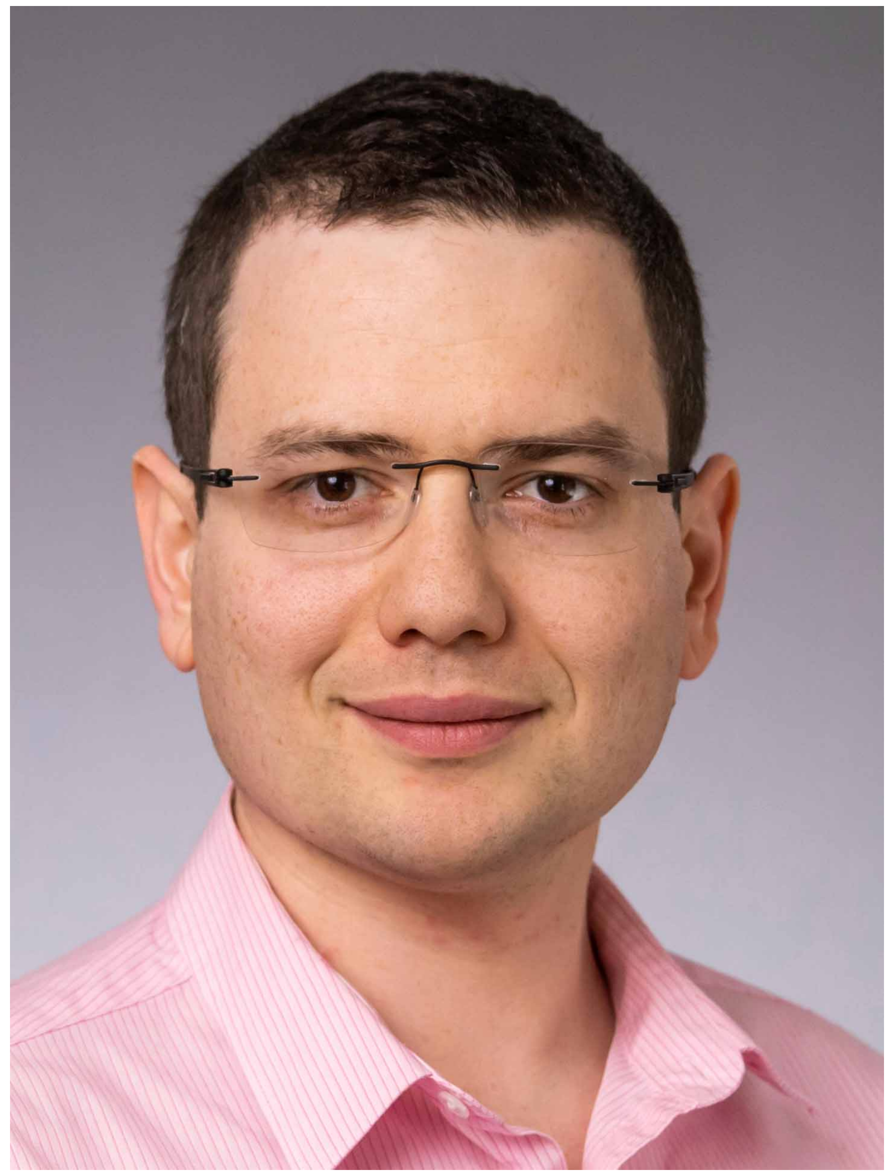
[Bogdanov–Ishai–Viola–Williamson 2016]



Braverman's theorem: polylog independence fools AC^0

"Fooling escalation"

Braverman for indistinguishability?
[Bogdanov–Ishai–Viola–Williamson 2016]



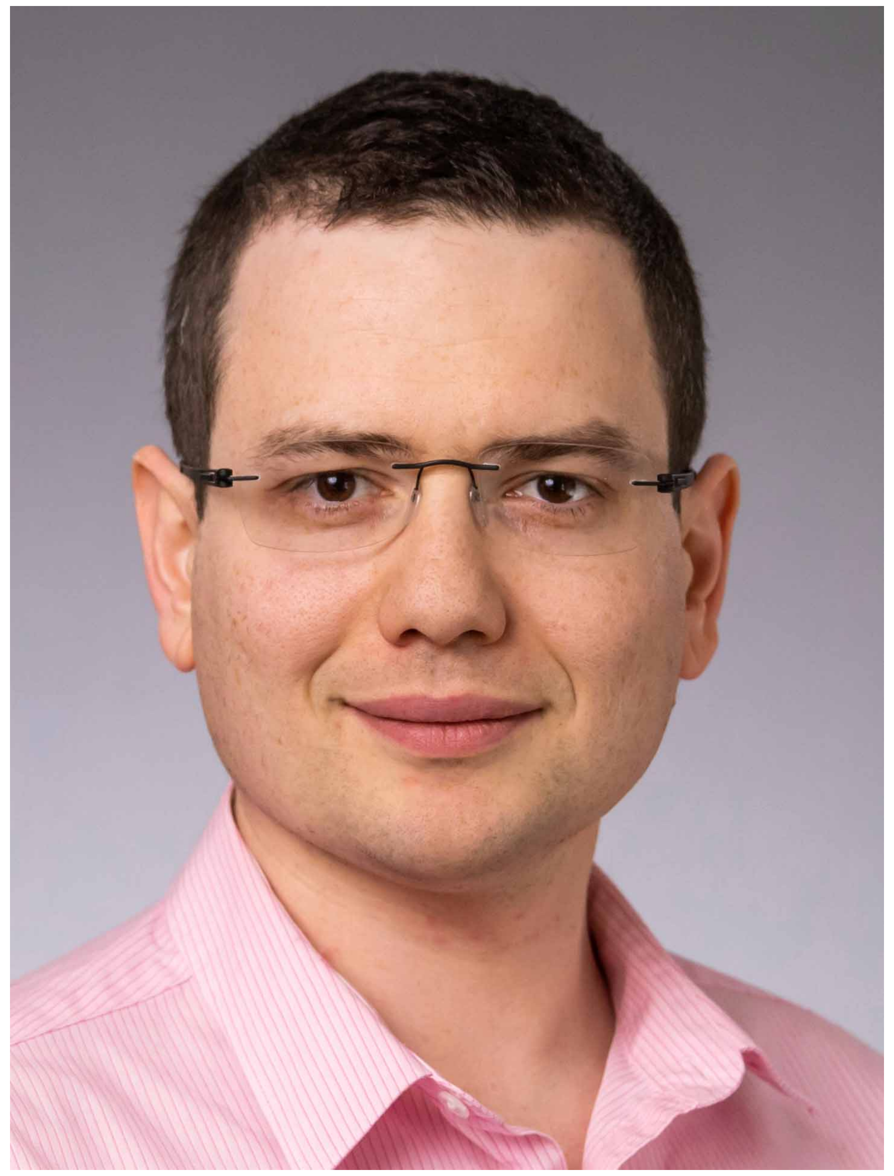
Braverman's theorem:
polylog independence
fools AC^0

"Fooling escalation"

Nisan–Szegedy:
approximate degree of OR is $\sqrt{n}$
so $\sqrt{n}$-wise indistinguishability
doesn't even fool OR!

Braverman for indistinguishability?

[Bogdanov–Ishai–Viola–Williamson 2016]



Braverman's theorem:
polylog independence
fools AC^0

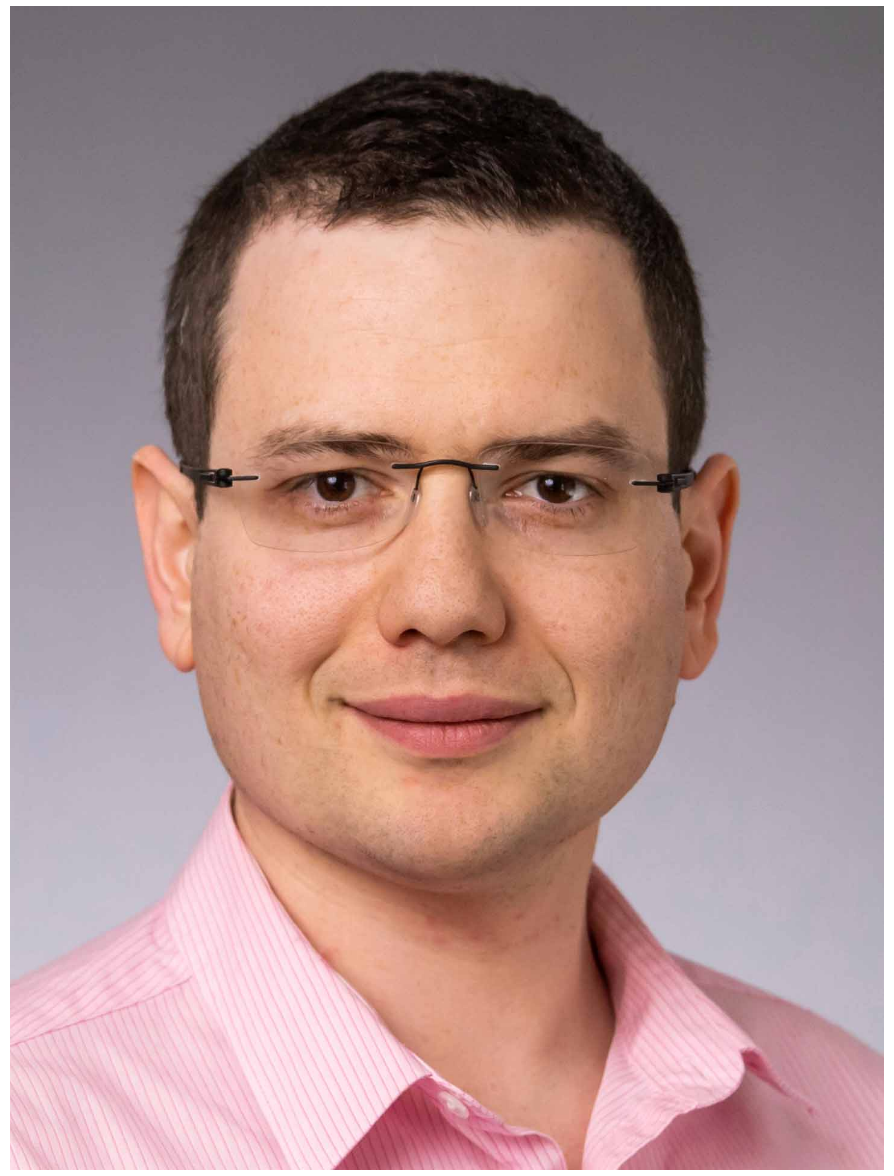
"Fooling escalation"

LP duality $\Rightarrow$

Nisan–Szegedy:
approximate degree of OR is $\sqrt{n}$
so $\sqrt{n}$-wise indistinguishability
doesn't even fool OR!

Braverman for indistinguishability?

[Bogdanov–Ishai–Viola–Williamson 2016]



Does Braverman hold for polylog indistinguishable *simple* sources?

Braverman's theorem: polylog independence fools AC^0

"Fooling escalation"

LP duality $\Rightarrow$

Nisan–Szegedy: approximate degree of OR is $\sqrt{n}$ so $\sqrt{n}$-wise indistinguishability doesn't even fool OR!

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Yes

Leakage-resilience of
secure multiparty computation
(also secure hardware etc.)

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Yes

Leakage-resilience of secure multiparty computation (also secure hardware etc.)

“Resilience escalation”

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Yes

↓

Leakage-resilience of secure multiparty computation (also secure hardware etc.)

"Resilience escalation"

AC^0 **models realistic leakage**

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Yes

Leakage-resilience of secure multiparty computation (also secure hardware etc.)

"Resilience escalation"

AC^0 **models realistic leakage**

No

Low-complexity secret sharing

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Yes

Leakage-resilience of secure multiparty computation (also secure hardware etc.)

"Resilience escalation"

AC^0 **models realistic leakage**

No

Low-complexity secret sharing

Generating shares is simple

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Yes

Leakage-resilience of secure multiparty computation (also secure hardware etc.)

"Resilience escalation"

AC^0 **models realistic leakage**

No

Low-complexity secret sharing

Generating shares is simple

Secret recovery in AC^0

Motivation

Does Braverman hold for polylog-wise indistinguishable simple sources?

Yes

Win–Win!

No

Leakage-resilience of secure multiparty computation (also secure hardware etc.)

“Resilience escalation”

AC^0 **models realistic leakage**

Low-complexity secret sharing

Generating shares is simple

Secret recovery in AC^0

Simple sources

Sources that are easy to sample given iid uniform random bits $r_1, r_2, r_3, \ldots$

Simple sources

Sources that are easy to sample given iid uniform random bits $r_1, r_2, r_3, \ldots$

- local sources

Simple sources

Sources that are easy to sample given iid uniform random bits $r_1, r_2, r_3, \ldots$

- local sources
- linear sources: **linear secret sharing**

Simple sources

Sources that are easy to sample given iid uniform random bits $r_1, r_2, r_3, \ldots$

- local sources
- linear sources: **linear secret sharing**
- affine sources: **“refreshing” secret sharing**

Simple sources

Sources that are easy to sample given iid uniform random bits $r_1, r_2, r_3, \ldots$

- local sources
- linear sources: **linear secret sharing**
- affine sources: **“refreshing” secret sharing**
- quadratic sources: **secure multiparty computation**

Simple sources

Sources that are easy to sample given iid uniform random bits $r_1, r_2, r_3, \ldots$

- local sources
- linear sources: **linear secret sharing**
- affine sources: **“refreshing” secret sharing**
- quadratic sources: **secure multiparty computation**

] Arise in natural crypto protocols when combining different shares

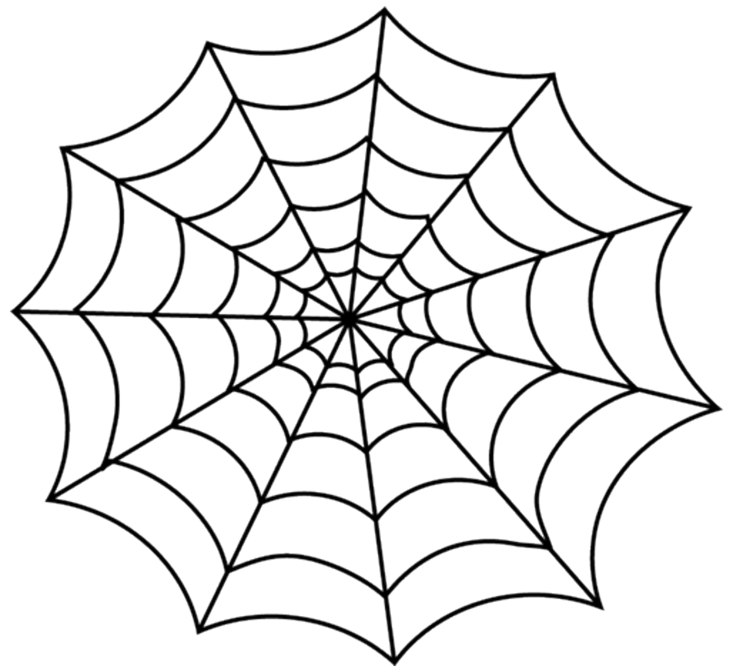
Simple sources

Sources that are easy to sample given iid uniform random bits $r_1, r_2, r_3, \ldots$

- local sources
- linear sources: **linear secret sharing**
- affine sources: **“refreshing” secret sharing**
- quadratic sources: **secure multiparty computation**

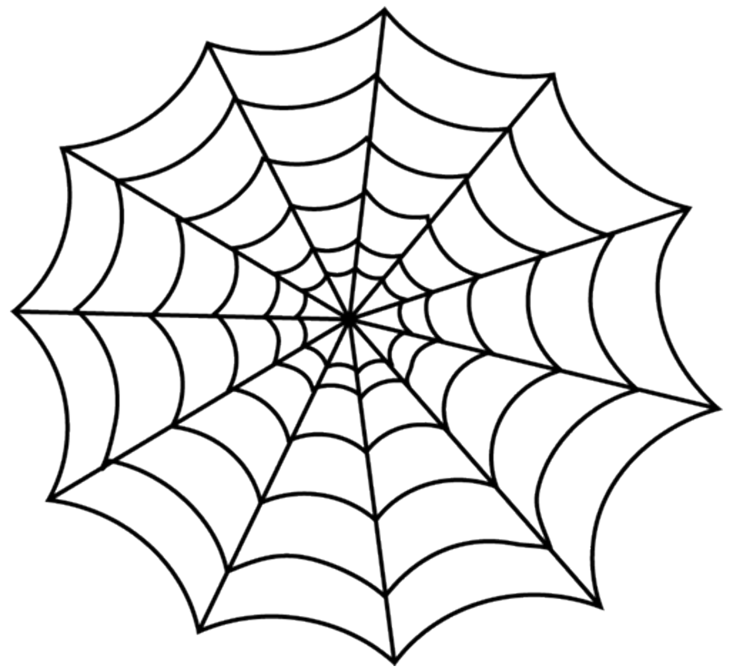
] Arise in natural crypto protocols when combining different shares

Some instances reducible to Braverman; others (e.g. LDPC codes) not



Web of Conjectures

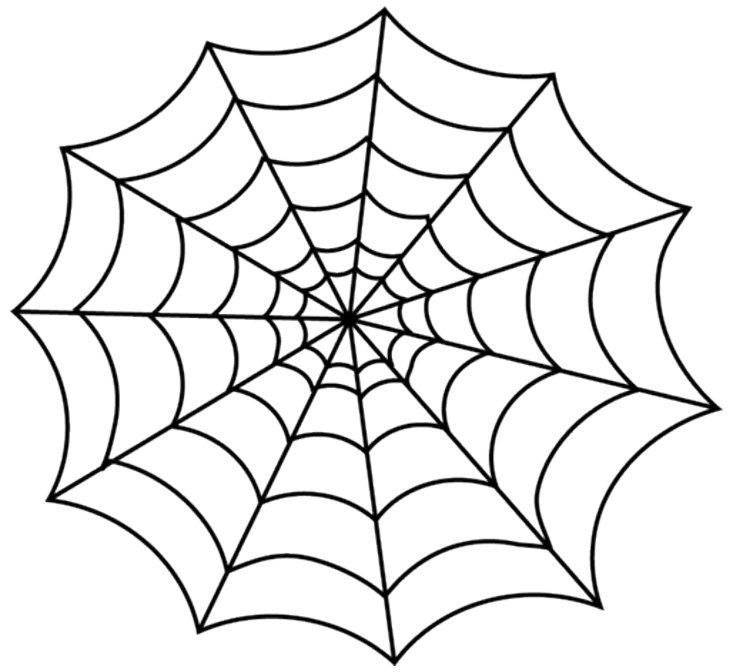
Given: class of sources (e.g. affine), class of circuits (e.g. AC^0)



Web of Conjectures

Given: class of sources (e.g. affine), class of circuits (e.g. AC^0)

Circuits cannot distinguish k-wise indistinguishable sources

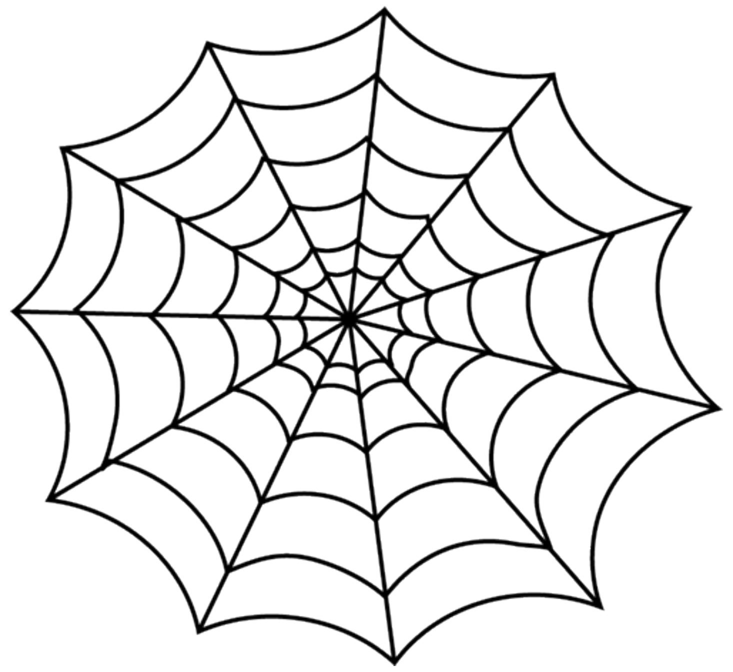


Web of Conjectures

Given: class of sources (e.g. affine), class of circuits (e.g. AC^0)

Circuits cannot distinguish k-wise indistinguishable sources

Circuits cannot distinguish k-wise indistinguishable sources of the form $X|_{r_1=0}$ and $X|_{r_1=1}$ ("cosets")



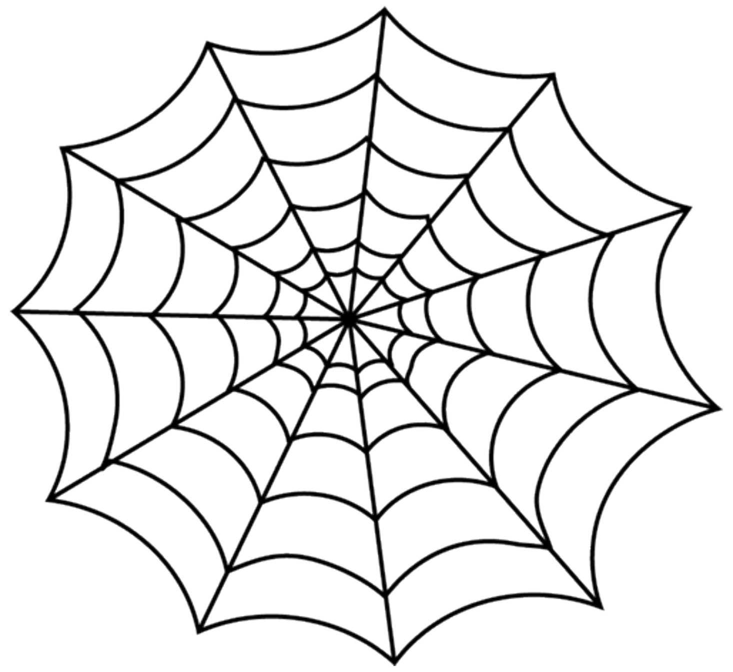
Web of Conjectures

Given: class of sources (e.g. affine), class of circuits (e.g. AC^0)

Circuits cannot distinguish k-wise indistinguishable sources

Circuits cannot distinguish k-wise indistinguishable sources of the form $X|_{r_1=0}$ and $X|_{r_1=1}$ ("cosets")

No k source bits contain any information on r_1 $\Rightarrow$ Circuits cannot predict r_1



Web of Conjectures

Given: class of sources (e.g. affine), class of circuits (e.g. AC^0)

Circuits cannot distinguish k-wise indistinguishable sources

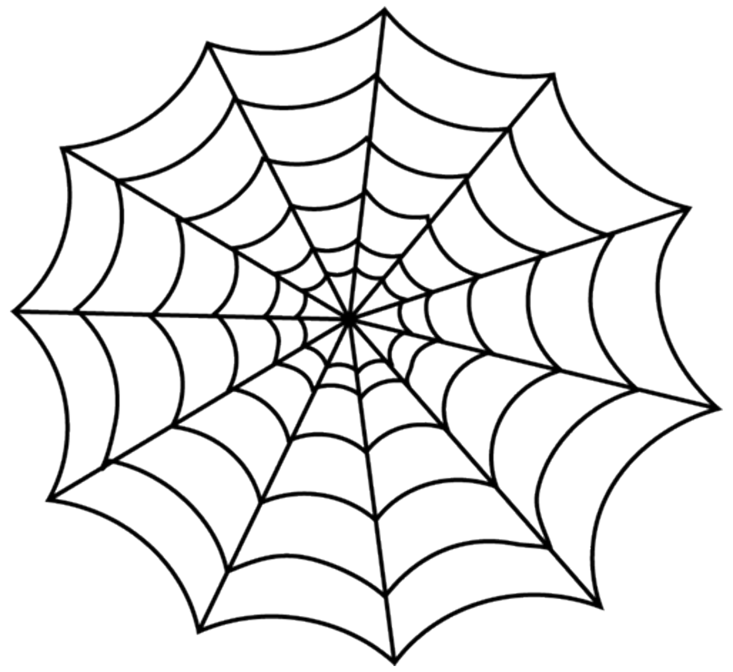
$\Downarrow$

Circuits cannot distinguish k-wise indistinguishable sources of the form $X|_{r_1=0}$ and $X|_{r_1=1}$ ("cosets")

Affine sources

$\Updownarrow$

No k source bits contain any information on r_1 $\Rightarrow$ Circuits cannot predict r_1



Web of Conjectures

Given: class of sources (e.g. affine), class of circuits (e.g. AC^0)

Affine sources

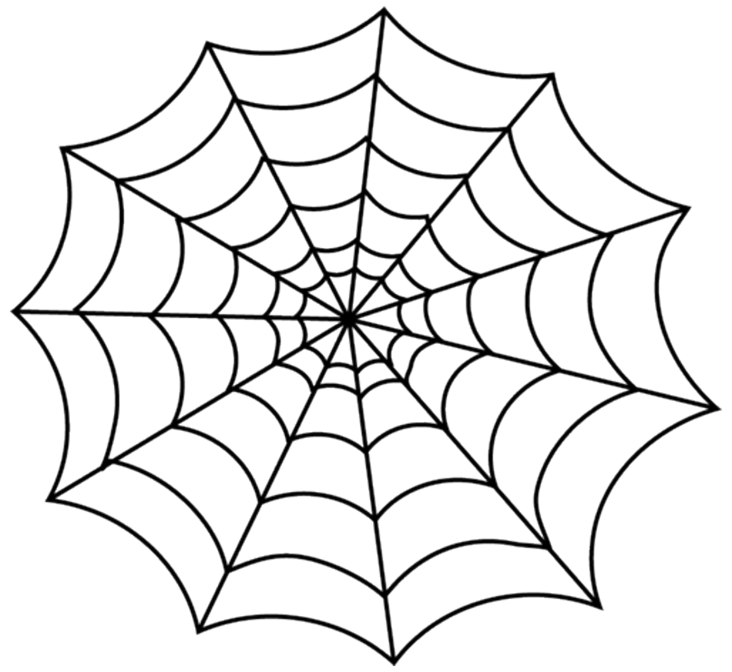
Circuits cannot distinguish k-wise indistinguishable sources

Circuits cannot distinguish k-wise indistinguishable sources of the form $X|_{r_1=0}$ and $X|_{r_1=1}$ ("cosets")

No k source bits contain any information on r_1 $\Rightarrow$ Circuits cannot predict r_1

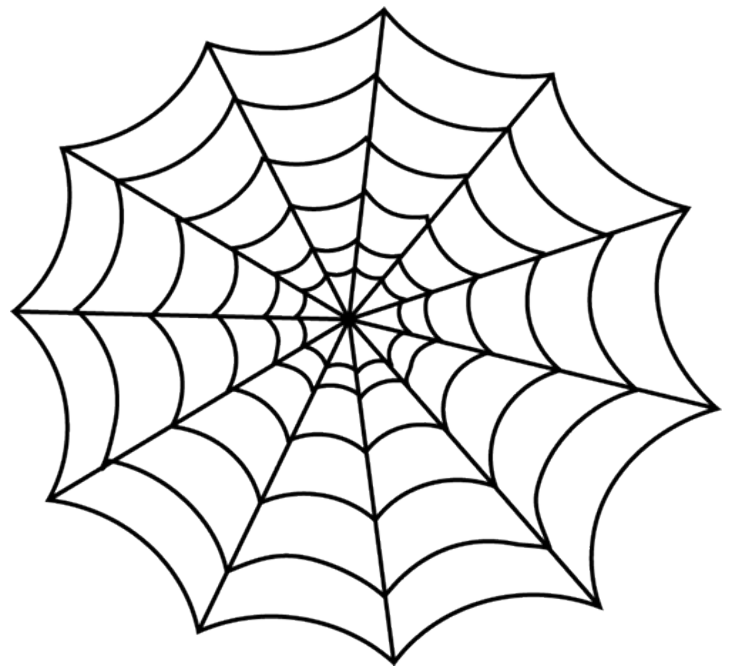
Special case: compute parity of codewords belonging to LDPC code

Inner Product w/ Preprocessing



Inner Product w/ Preprocessing

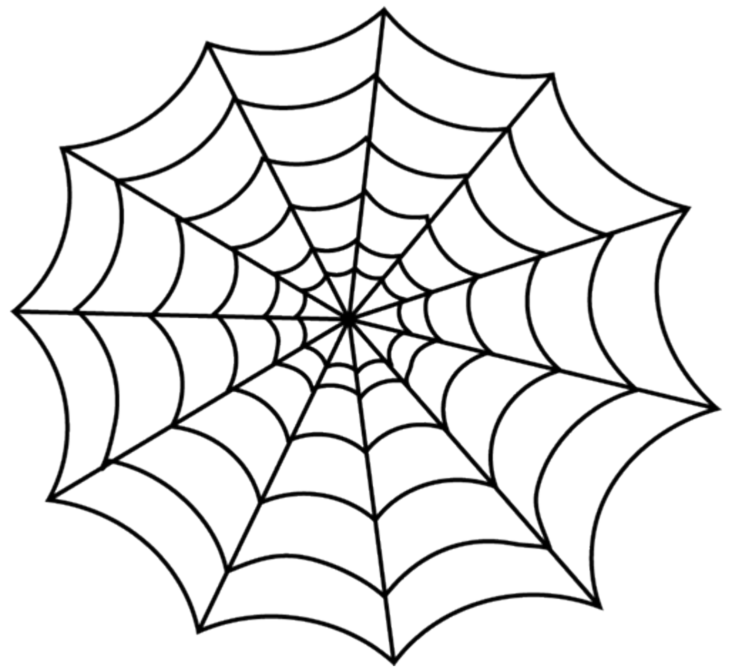
IPPP: Compute $\langle x, y\rangle$ in AC^0 given $f_i(x), g_j(y)$



Inner Product w/ Preprocessing

IPPP: Compute $\langle x, y \rangle$ in AC^0 given $f_i(x), g_j(y)$

Compute IP in PH^{cc}

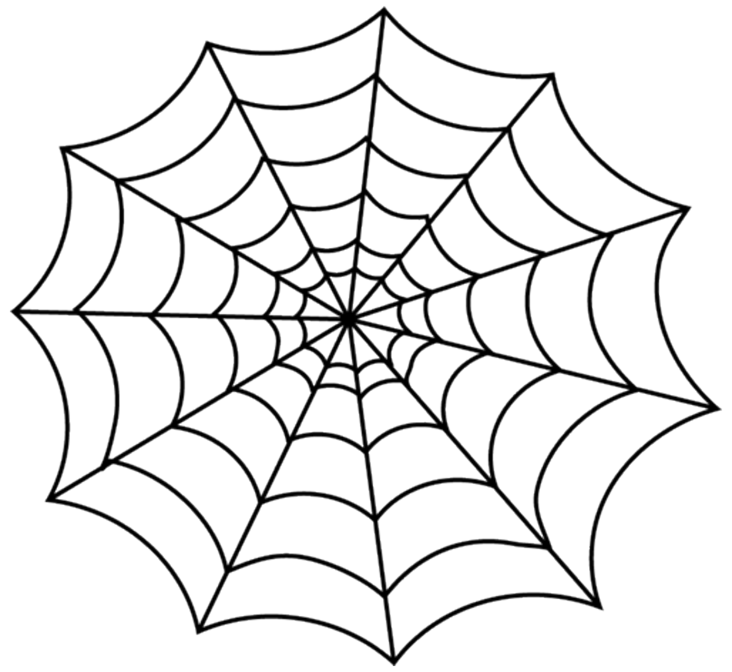


Inner Product w/ Preprocessing

IPPP: Compute $\langle x, y \rangle$ in AC^0 given $f_i(x), g_j(y)$

Compute IP in PH^{cc}

Linear IPPP: Compute $\langle x, y \rangle$ in $\mathsf{AC}^0 \circ \oplus$ (equivalently, $f_i(x), g_j(y)$ linear)



Inner Product w/ Preprocessing

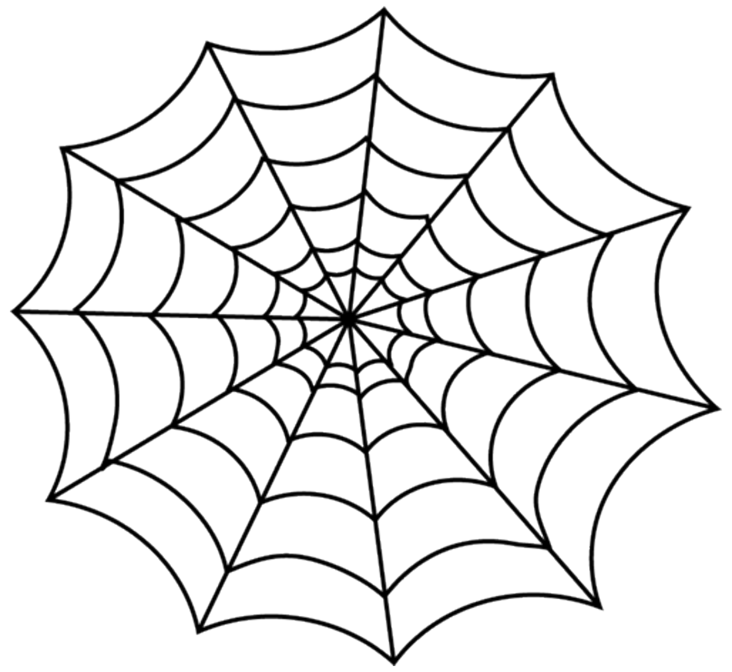
IPPP: Compute $\langle x, y \rangle$ in AC^0 given $f_i(x), g_j(y)$

Compute IP in PH^{cc}

Linear IPPP: Compute $\langle x, y \rangle$ in $AC^0 \circ \oplus$ (equivalently, $f_i(x), g_j(y)$ linear)

Linear sources, AC^0 circuits

No k source bits contain any information on r_1 $\Rightarrow$ Circuits cannot predict r_1



Inner Product w/ Preprocessing

IPPP: Compute $\langle x, y \rangle$ in AC^0 given $f_i(x), g_j(y)$

Compute IP in PH^{cc}

Linear IPPP: Compute $\langle x, y \rangle$ in $\mathrm{AC}^0 \circ \oplus$ (equivalently, $f_i(x), g_j(y)$ linear)

Linear sources, AC^0 circuits

No k source bits contain any information on r_1 $\Rightarrow$ Circuits cannot predict r_1

$\Downarrow$

Cannot compute $\langle x, y \rangle$ for all x in AC^0 given linear $g_j(y)$

$\Downarrow$

Cannot compute $\langle x, y \rangle$ in AC^0 given linear $f_i(x), g_j(y)$

Inner Product w/ Preprocessing

IPPP: Compute $\langle x, y \rangle$ in AC^0 given $f_i(x), g_j(y)$

Compute IP in PH^{cc}

Linear IPPP: Compute $\langle x, y \rangle$ in $AC^0 \circ \oplus$ (equivalently, $f_i(x), g_j(y)$ linear)

Linear sources, AC^0 circuits

No k source bits contain any information on r_1 $\Rightarrow$ Circuits cannot predict r_1

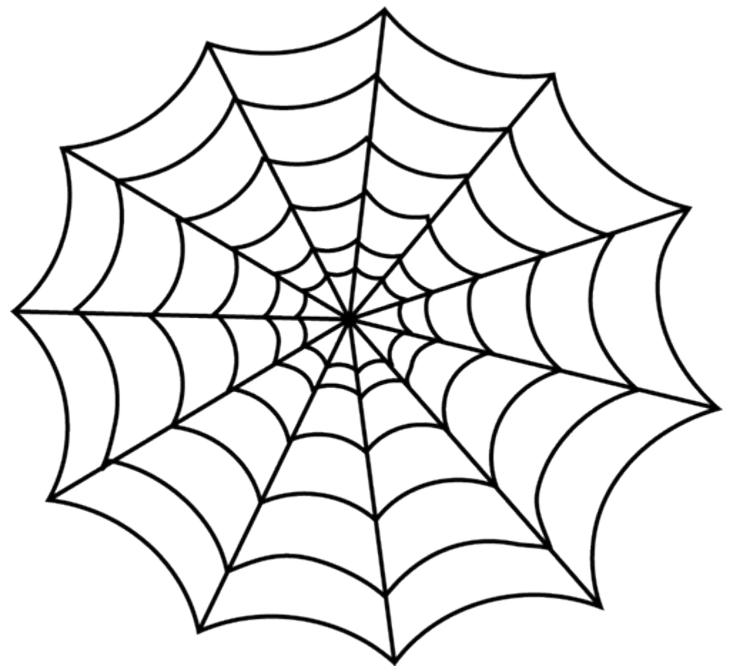
$\Downarrow$

Cannot compute $\langle x, y \rangle$ for all x in AC^0 given linear $g_j(y)$

$\Downarrow$

Cannot compute $\langle x, y \rangle$ in AC^0 given linear $f_i(x), g_j(y)$

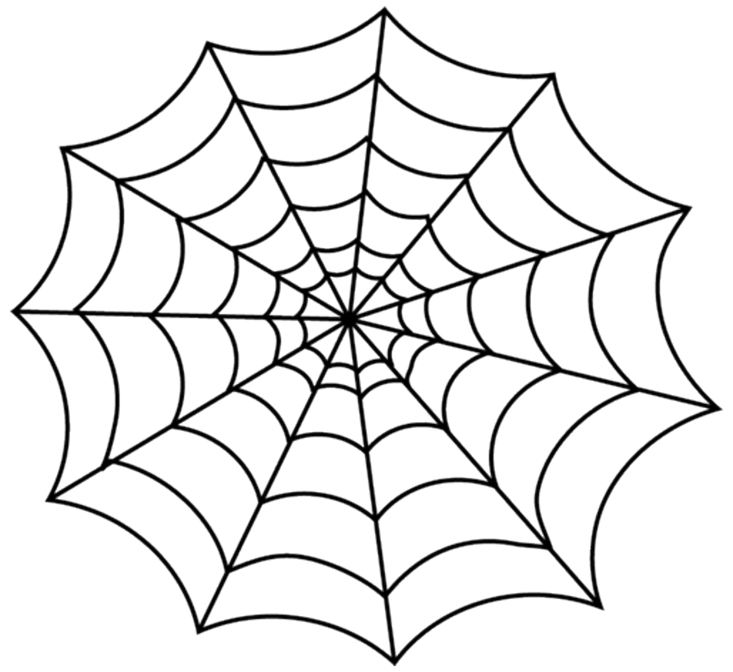
Arbitrary preprocessing



Barrier

AC^0 **circuits cannot distinguish k-wise indistinguishable linear sources**

Cannot compute $\langle x, y \rangle$ in $\mathsf{AC}^0 \circ \oplus$



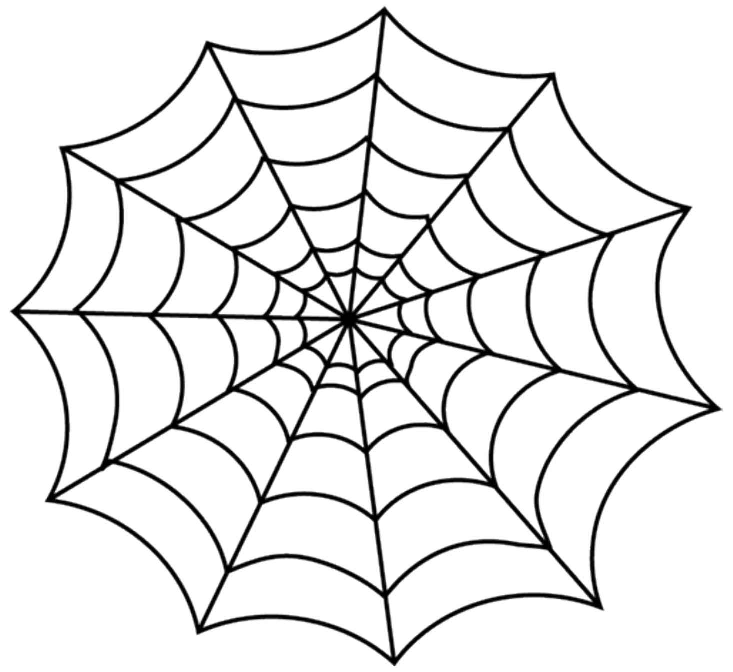
Barrier

AC^0 **circuits cannot distinguish k-wise indistinguishable linear sources**

$\Downarrow$

Cannot compute $\langle x, y \rangle$ in $\mathrm{AC}^0 \circ \oplus$

Hard!



Barrier

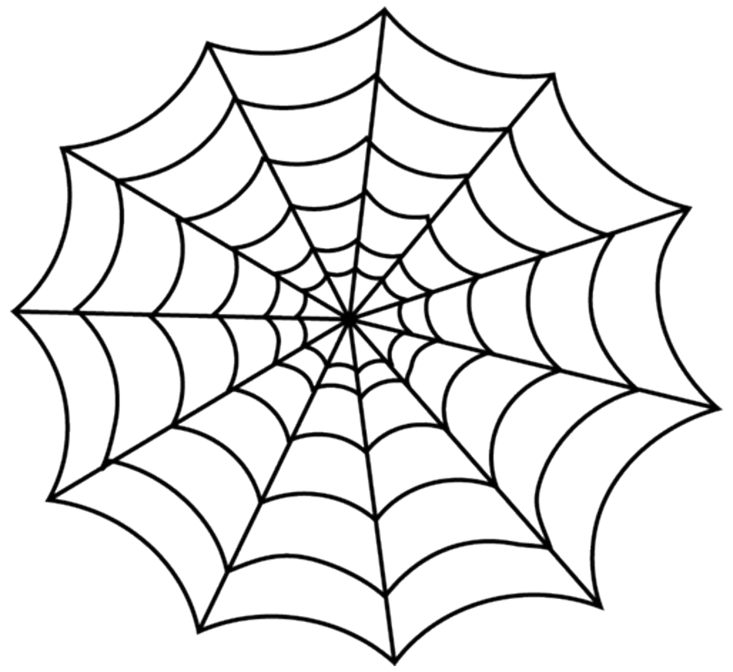
AC^0 **circuits cannot distinguish k-wise indistinguishable linear sources**

Cannot compute $\langle x, y \rangle$ in $\mathsf{AC}^0 \circ \oplus$

Hard!

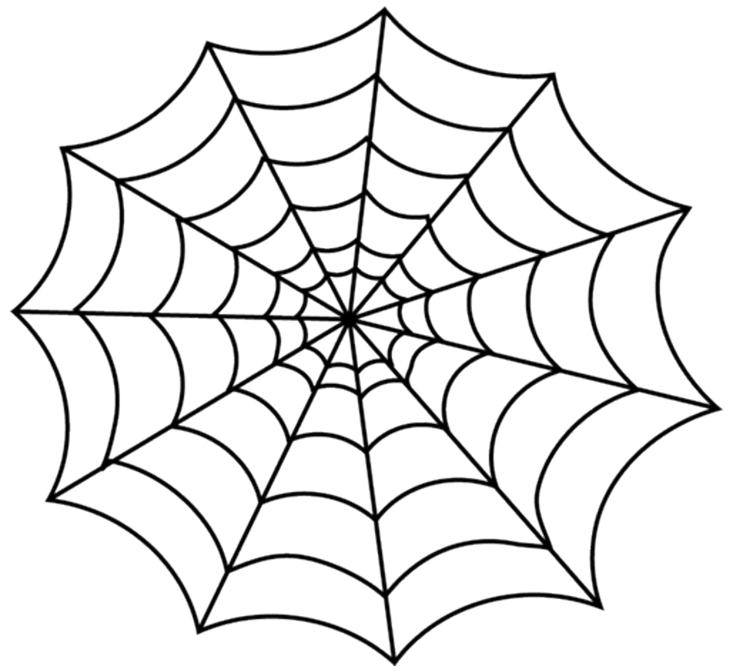
Concentrate on OR, decision trees, DNFs

OR is interesting!



OR is interesting!

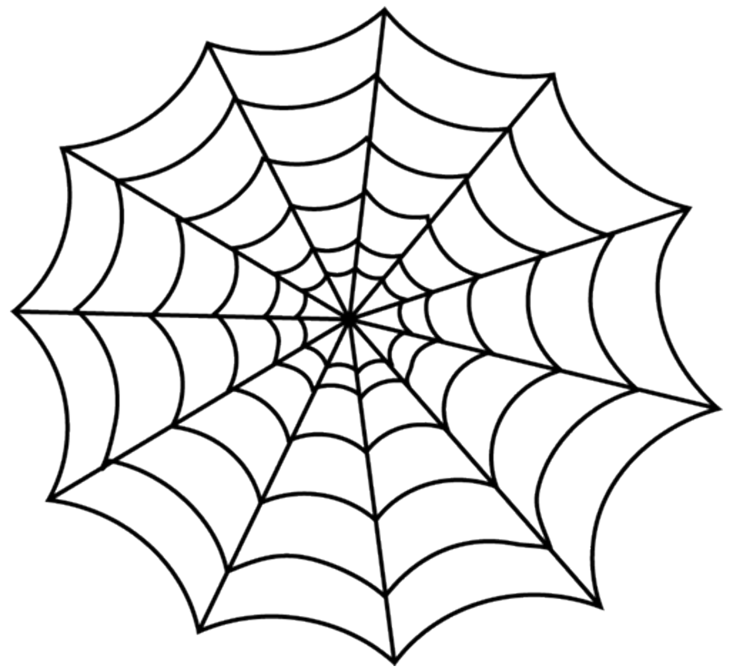
selective failure attacks



OR is interesting!

selective failure attacks

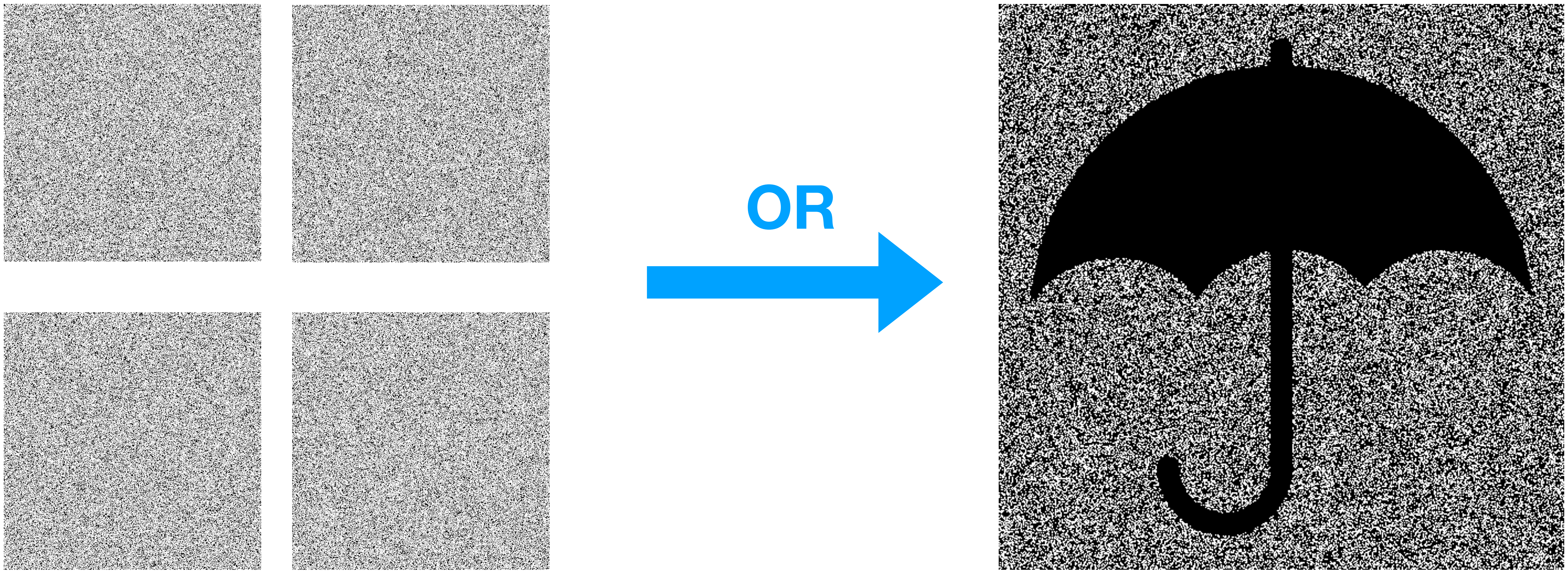
visual secret sharing
[Naor–Shamir 1994]

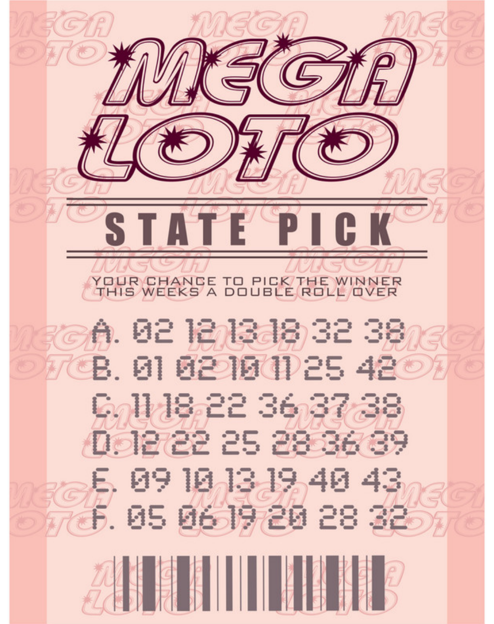


OR is interesting!

selective failure attacks

visual secret sharing
[Naor–Shamir 1994]

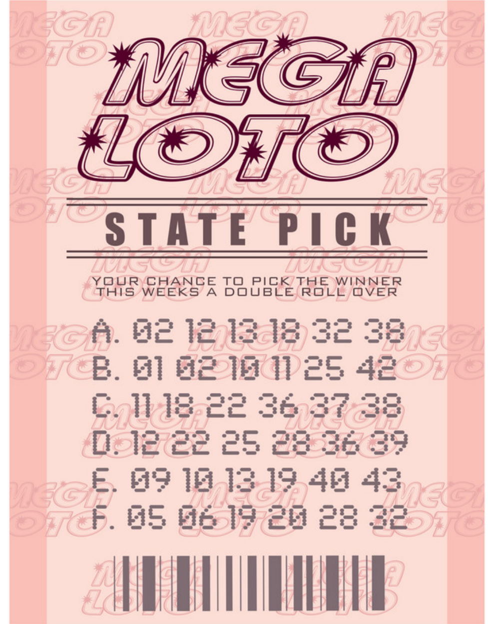




Our Results

k-wise indistinguishable source

ε = distinguishing advantage



Our Results

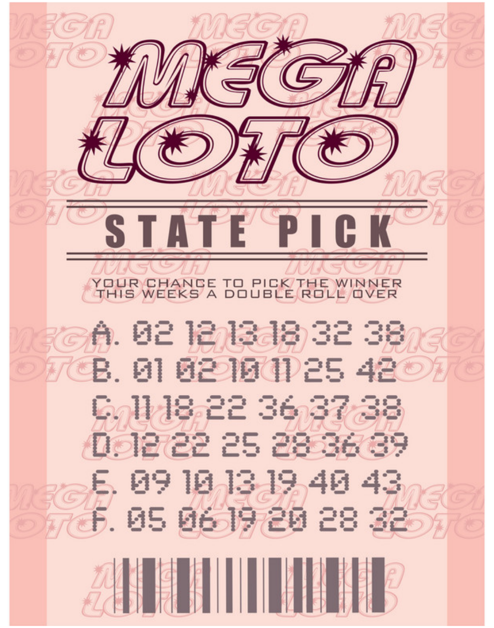
k-wise indistinguishable source

ε = distinguishing advantage

Affine sources

Constant k fools OR, decision trees, narrow DNFs

OR: $k = \log(1/\epsilon)$



Our Results

k-wise indistinguishable source

ε = distinguishing advantage

Affine sources

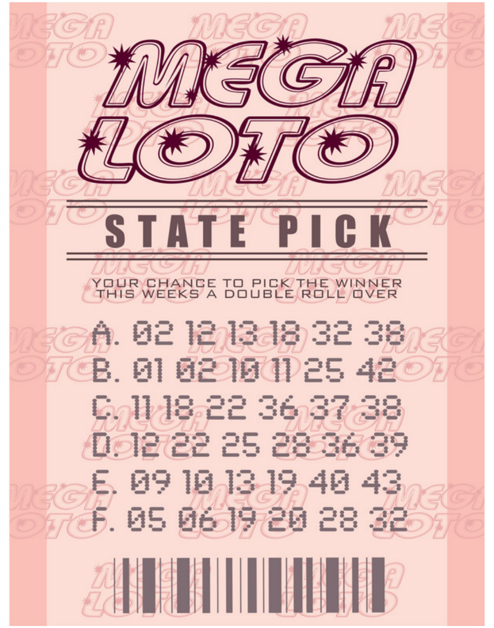
Constant degree
or
Constant locality

Constant k fools OR, decision trees, narrow DNFs

Constant k fools OR
Quadratic: decision trees

OR: $k = \log(1/\epsilon)$

Quadratic: $k = \text{polylog}(1/\epsilon)$



Our Results

k-wise indistinguishable source

ε = distinguishing advantage

Affine sources

Constant degree
or
Constant locality

Degree $\log n$

Constant k fools OR, decision trees, narrow DNFs

Constant k fools OR
Quadratic: decision trees

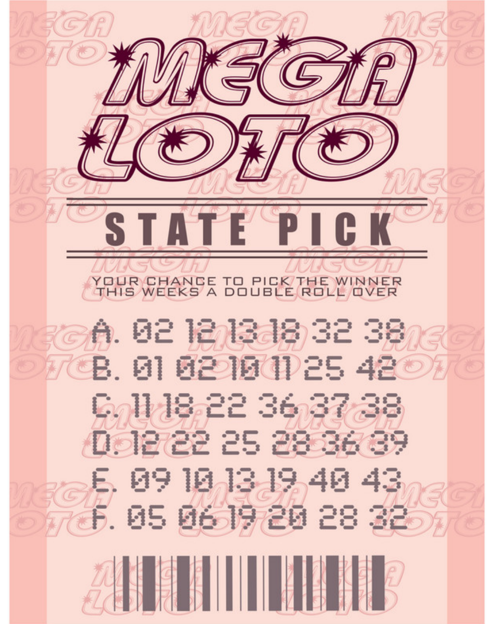
OR distinguishes $k = \sqrt{n}$

OR: $k = \log(1/\epsilon)$

Quadratic: $k = \text{polylog}(1/\epsilon)$

Mixture of iid: application to visual secret sharing


Techniques



Our Results

k-wise indistinguishable source

ε = distinguishing advantage

Affine sources

Constant degree
or
Constant locality

Degree $\log n$

Constant k fools OR, decision trees, narrow DNFs

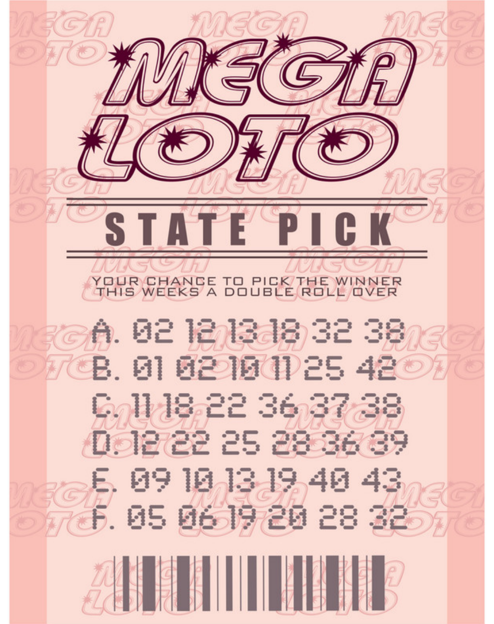
Constant k fools OR
Quadratic: decision trees

OR distinguishes $k = \sqrt{n}$

OR: $k = \log(1/\epsilon)$

Quadratic: $k = \operatorname{polylog}(1/\epsilon)$

Mixture of iid: application to visual secret sharing



Our Results

k-wise indistinguishable source

ε = distinguishing advantage

Affine sources

Constant k fools OR, decision trees, narrow DNFs

OR: $k = \log(1/\epsilon)$

Constant degree or Constant locality

Constant k fools OR
Quadratic: decision trees

Quadratic: $k = \mathrm{polylog}(1/\epsilon)$

Degree $\log n$

OR distinguishes $k = \sqrt{n}$

Mixture of iid: application to visual secret sharing

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable sources X, Y distinguished by OR

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable sources X, Y distinguished by OR

Natural idea: Consider symmetric distributions

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable sources X, Y distinguished by OR

Natural idea: Consider symmetric distributions

De Finetti's theorem: Symmetric distributions are mixtures of iid (in the limit)

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable sources X, Y distinguished by OR

Natural idea: Consider symmetric distributions

De Finetti's theorem: Symmetric distributions are mixtures of iid (in the limit)

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable sources X, Y distinguished by OR

Natural idea: Consider symmetric distributions

De Finetti's theorem: Symmetric distributions are mixtures of iid (in the limit)

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$
Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

k-wise indistinguishability $\iff \underset{i \sim p}{\mathbb{E}}[\alpha_i^\ell] = \underset{j \sim q}{\mathbb{E}}[\beta_j^\ell]$ for all $\ell \leq k$

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable sources X, Y distinguished by OR

Natural idea: Consider symmetric distributions

De Finetti's theorem: Symmetric distributions are mixtures of iid (in the limit)

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

k-wise indistinguishability $\iff \mathop{\mathbb{E}}_{i\sim p}[\alpha_i^\ell] = \mathop{\mathbb{E}}_{j\sim q}[\beta_j^\ell]$ for all $\ell \leq k$

OR can distinguish: $p_1 = \Omega(1)$, $\alpha_1 = 1$, $\beta_j \leq 1 - \Omega\left(\frac{1}{n}\right)$

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable sources X, Y distinguished by OR

Natural idea: Consider symmetric distributions

De Finetti's theorem: Symmetric distributions are mixtures of iid (in the limit)

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

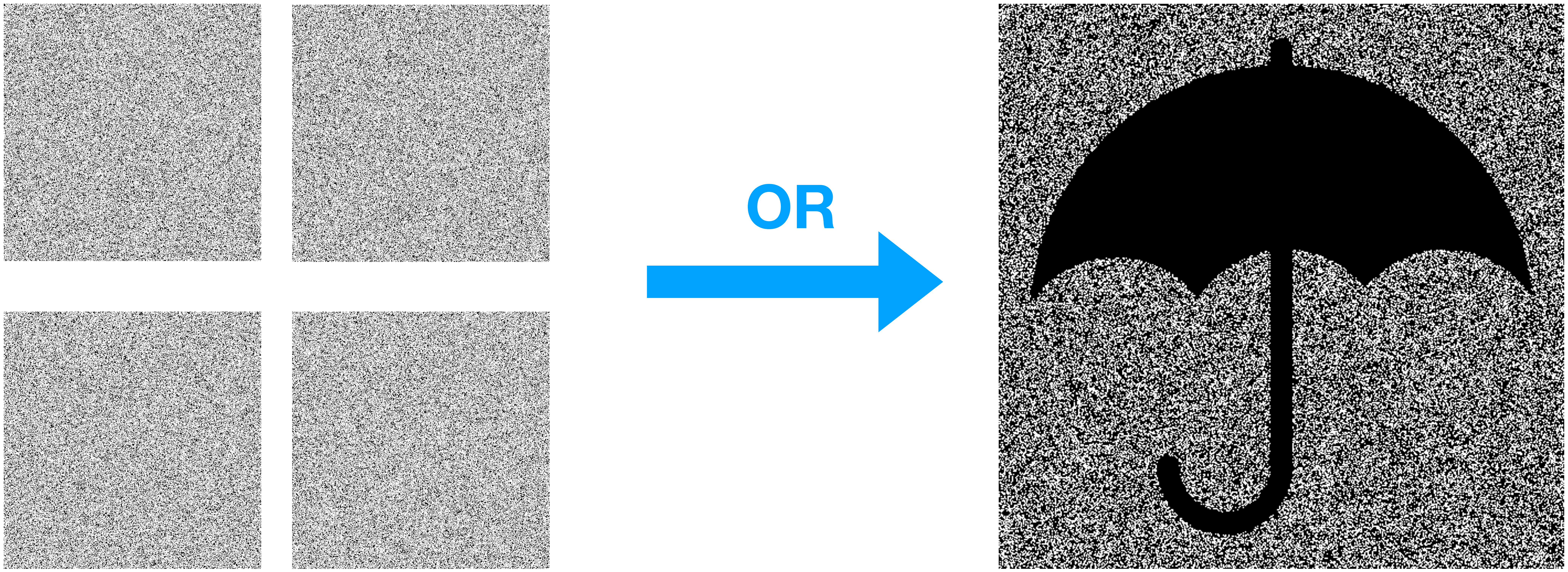
Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

k-wise indistinguishability $\iff \mathbb{E}_{i \sim p}[\alpha_i^\ell] = \mathbb{E}_{j \sim q}[\beta_j^\ell]$ for all $\ell \leq k$

OR can distinguish: $p_1 = \Omega(1)$, $\alpha_1 = 1$, $\beta_j \leq 1 - \Omega\left(\frac{1}{n}\right)$

Explicit construction — guess α_i, β_j, compute p, q

Application: Visual Secret Sharing



X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

Construct *polynomial size* decision trees for sampling "reduced precision" X, Y

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

Construct *polynomial size* decision trees for sampling “reduced precision” X, Y

Skipping some technicalities

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

X: sample i according to distribution p, then sample n iid $\text{Bernoulli}(\alpha_i)$
Y: sample j according to distribution q, then sample n iid $\text{Bernoulli}(\beta_j)$

Construct *polynomial size* decision trees for sampling "reduced precision" X, Y

Skipping some technicalities

Express each source as *disjoint* (*unambiguous*) DNF $T_1 \vee \cdots \vee T_m$

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$
Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

Construct *polynomial size* decision trees for sampling “reduced precision” X, Y

Skipping some technicalities

Express each source as *disjoint* (*unambiguous*) DNF $T_1 \vee \cdots \vee T_m$

Reduce degree to $O(\log \text{size}) = O(\log n)$ using Razborov–Smolensky encoding

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$

Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

Construct *polynomial size* decision trees for sampling “reduced precision” X, Y

Skipping some technicalities

Express each source as *disjoint* (*unambiguous*) DNF $T_1 \vee \cdots \vee T_m$

Reduce degree to $O(\log \text{size}) = O(\log n)$ using Razborov–Smolensky encoding

Encode $\ell_1 \wedge \cdots \wedge \ell_w$ as $\prod_k \left[1 + \sum_j (1 + \ell_j) r_{k,j}\right]$

OR distinguishes $\sqrt{n}$-wise indistinguishable *simple* sources

Goal: Construct two $\sqrt{n}$-wise indistinguishable *simple* sources Z, W distinguished by OR

X: sample i according to distribution p, then sample n iid $\mathrm{Bernoulli}(\alpha_i)$
Y: sample j according to distribution q, then sample n iid $\mathrm{Bernoulli}(\beta_j)$

Construct *polynomial size* decision trees for sampling “reduced precision” X, Y

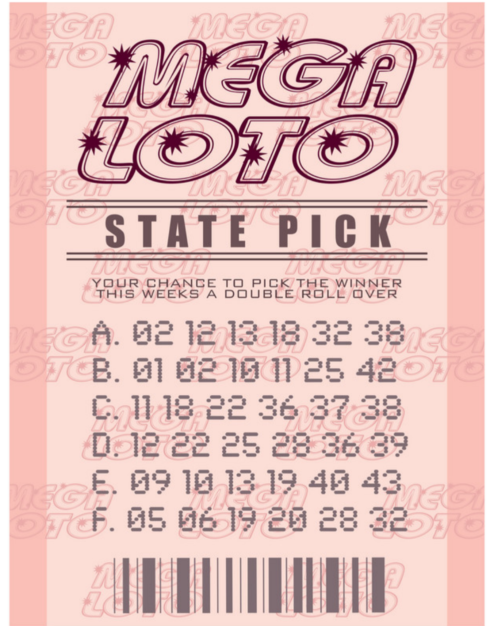
Skipping some technicalities

Express each source as *disjoint* (*unambiguous*) DNF $T_1 \vee \cdots \vee T_m$

Reduce degree to $O(\log \text{size}) = O(\log n)$ using Razborov–Smolensky encoding

Encode $\ell_1 \wedge \cdots \wedge \ell_w$ as $\prod_k \left[1 + \sum_j (1 + \ell_j) r_{k,j}\right]$

Sum over all terms (use disjointness)



Our Results

k-wise indistinguishable source

ε = distinguishing advantage

Affine sources

Constant degree
or
Constant locality

Degree $\log n$

Constant k fools OR, decision trees, narrow DNFs

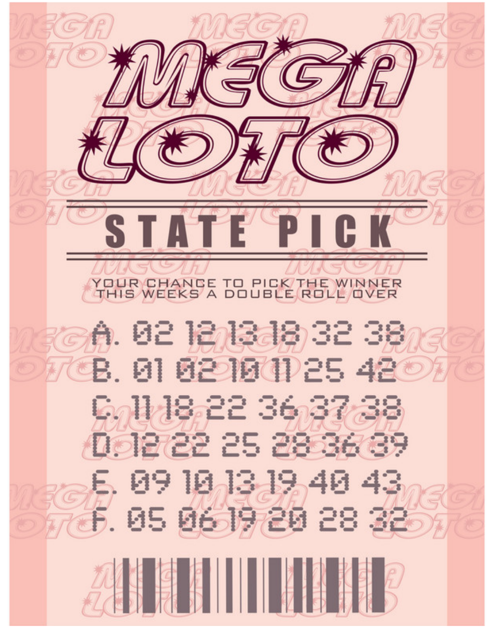
Constant k fools OR
Quadratic: decision trees

OR distinguishes $k = \sqrt{n}$

OR: $k = \log(1/\epsilon)$

Quadratic: $k = \mathrm{polylog}(1/\epsilon)$

Mixture of iid: application to visual secret sharing



Our Results

k-wise indistinguishable source

ε = distinguishing advantage

Affine sources

Constant degree
or
Constant locality

Degree $\log n$

Constant k fools OR, decision trees, narrow DNFs

Constant k fools OR
Quadratic: decision trees

OR distinguishes $k = \sqrt{n}$

OR: $k = \log(1/\epsilon)$

Quadratic: $k = \mathrm{polylog}(1/\epsilon)$

Mixture of iid: application to visual secret sharing

$O(1)$-wise indistinguishable *simple* sources fool OR

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source “predicted” by small set of coordinates

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source “predicted” by small set of coordinates

Formally: Probability that $X_i = 0$ for all $i \in S$ but $X \neq 0$ is at most ε

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source “predicted” by small set of coordinates

Formally: Probability that $X_i = 0$ for all $i \in S$ but $X \neq 0$ is at most ε

Reduces problem to understanding a *single* source

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source "predicted" by small set of coordinates

Formally: Probability that $X_i = 0$ for all $i \in S$ but $X \neq 0$ is at most ε

Reduces problem to understanding a *single* source

Suppose X, Y are two k-wise indistinguishable simple sources

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source "predicted" by small set of coordinates

Formally: Probability that $X_i = 0$ for all $i \in S$ but $X \neq 0$ is at most ε

Reduces problem to understanding a *single* source

Suppose X, Y are two k-wise indistinguishable simple sources

S predicts X and T predicts $Y \implies S \cup T$ predicts both

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source "predicted" by small set of coordinates

Formally: Probability that $X_i = 0$ for all $i \in S$ but $X \neq 0$ is at most ε

Reduces problem to understanding a *single* source

Suppose X, Y are two k-wise indistinguishable simple sources

S predicts X and T predicts $Y \implies S \cup T$ predicts both

S, T are small $\implies \Pr[X = 0] \approx \Pr[X|_{S \cup T} = 0] = \Pr[Y|_{S \cup T} = 0] \approx \Pr[Y = 0]$

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source "predicted" by small set of coordinates

Formally: Probability that $X_i = 0$ for all $i \in S$ but $X \neq 0$ is at most ε

Reduces problem to understanding a *single* source

Suppose X, Y are two k-wise indistinguishable simple sources

S predicts X and T predicts $Y \implies S \cup T$ predicts both

S, T are small $\implies \Pr[X = 0] \approx \Pr[X|_{S \cup T} = 0] = \Pr[Y|_{S \cup T} = 0] \approx \Pr[Y = 0]$

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Idea: Value of OR on any simple source “predicted” by small set of coordinates

Formally: Probability that $X_i = 0$ for all $i \in S$ but $X \neq 0$ is at most ε

Reduces problem to understanding a *single* source

Suppose X, Y are two k-wise indistinguishable simple sources

S predicts X and T predicts $Y \implies S \cup T$ predicts both

S, T are small $\implies \Pr[X = 0] \approx \Pr[X|_{S \cup T} = 0] = \Pr[Y|_{S \cup T} = 0] \approx \Pr[Y = 0]$

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$

Unlikely that $r_i = 0$ for many i

$O(1)$-wise indistinguishable *simple* sources fool OR

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$

Unlikely that $r_i = 0$ for many i

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$

Unlikely that $r_i = 0$ for many i

Warm-up: Linear sources

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$

Unlikely that $r_i = 0$ for many i

Warm-up: Linear sources

Case 1: Source has low rank

Choose basis

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$

Unlikely that $r_i = 0$ for many i

Warm-up: Linear sources

Case 1: Source has low rank

Choose basis

Case 2: Source has high rank

Choose many linearly independent X_i

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$

Unlikely that $r_i = 0$ for many i

Warm-up: Linear sources

Case 1: Source has low rank

Choose basis

Case 2: Source has high rank

Choose many linearly independent X_i

Sources of low degree or low locality

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Example 1: $X_i = r_0 r_i$

If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$

Unlikely that $r_i = 0$ for many i

Warm-up: Linear sources

Case 1: Source has low rank

Choose basis

Case 2: Source has high rank

Choose many linearly independent X_i

Sources of low degree or low locality

Case 1: Source has low "rank"

Use to *simplify* source

$O(1)$-wise indistinguishable *simple* sources fool OR

Simple sources: samplable from $r_1, r_2, r_3, \ldots$ in constant degree or constant locality

Goal: Find small S s.t. probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Example 1: $X_i = r_0 r_i$ | If $r_0 r_i = 0$ for many i then probably $r_0 = 0$ hence $X = 0$

Example 2: $X_i = r_i$ | Unlikely that $r_i = 0$ for many i

Warm-up: Linear sources

Case 1: Source has low rank | Choose basis

Case 2: Source has high rank | Choose many linearly independent X_i

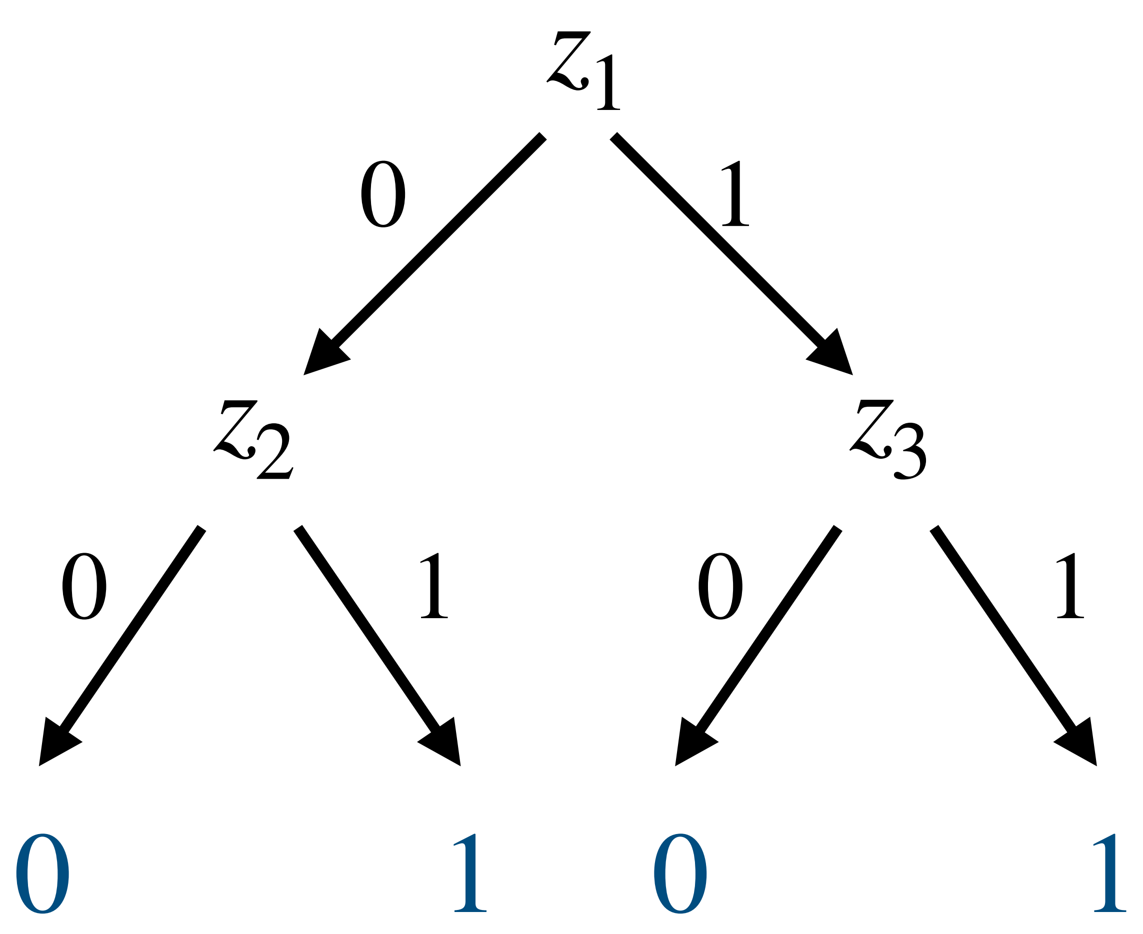
Sources of low degree or low locality

Case 1: Source has low “rank” | Use to *simplify* source

Case 2: Source has high “rank” | Choose many “independent” X_i

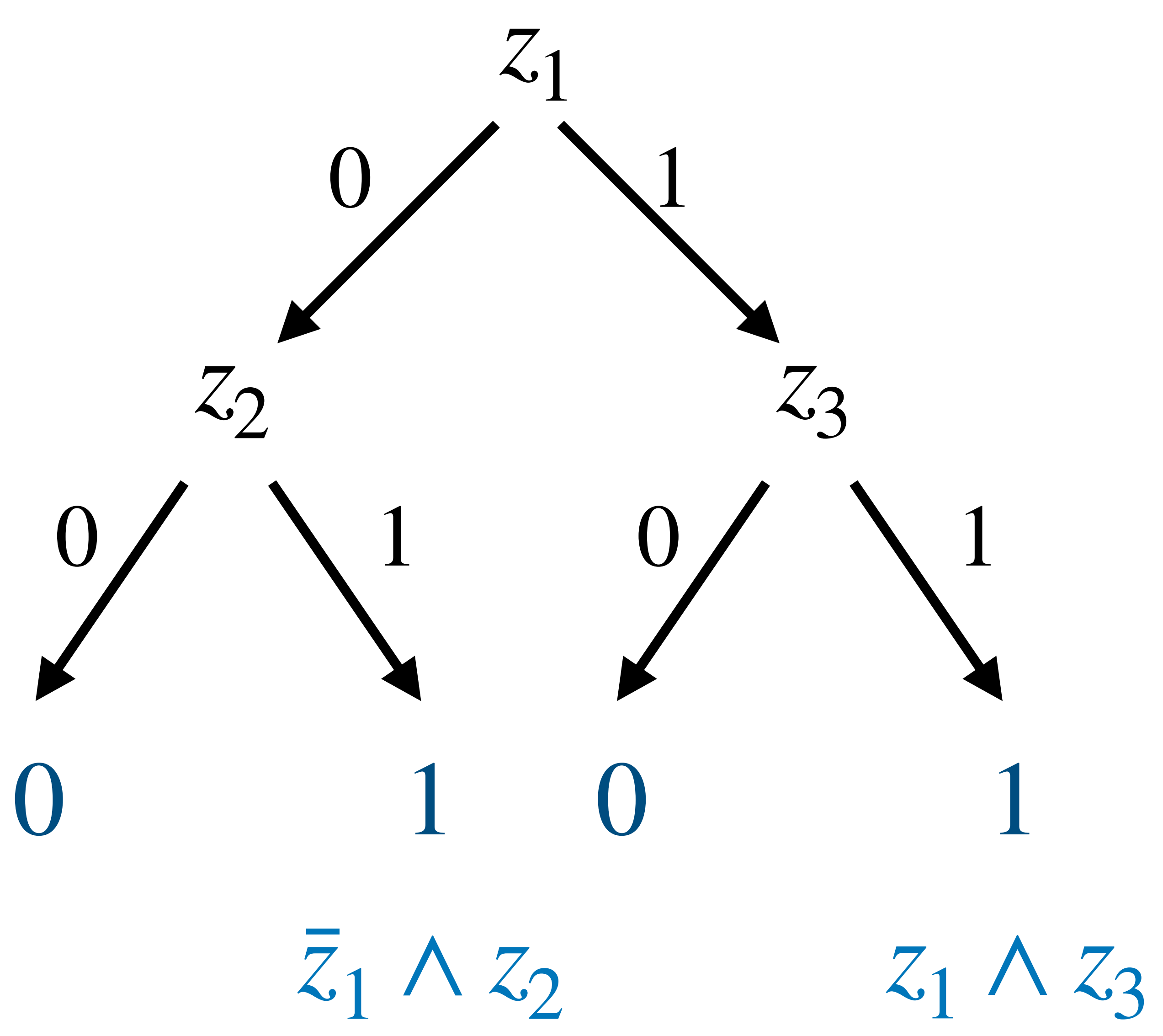
polylog-wise indistinguishable quadratic sources fool polynomial size decision trees

Decision tree with m leaves



polylog-wise indistinguishable quadratic sources fool polynomial size decision trees

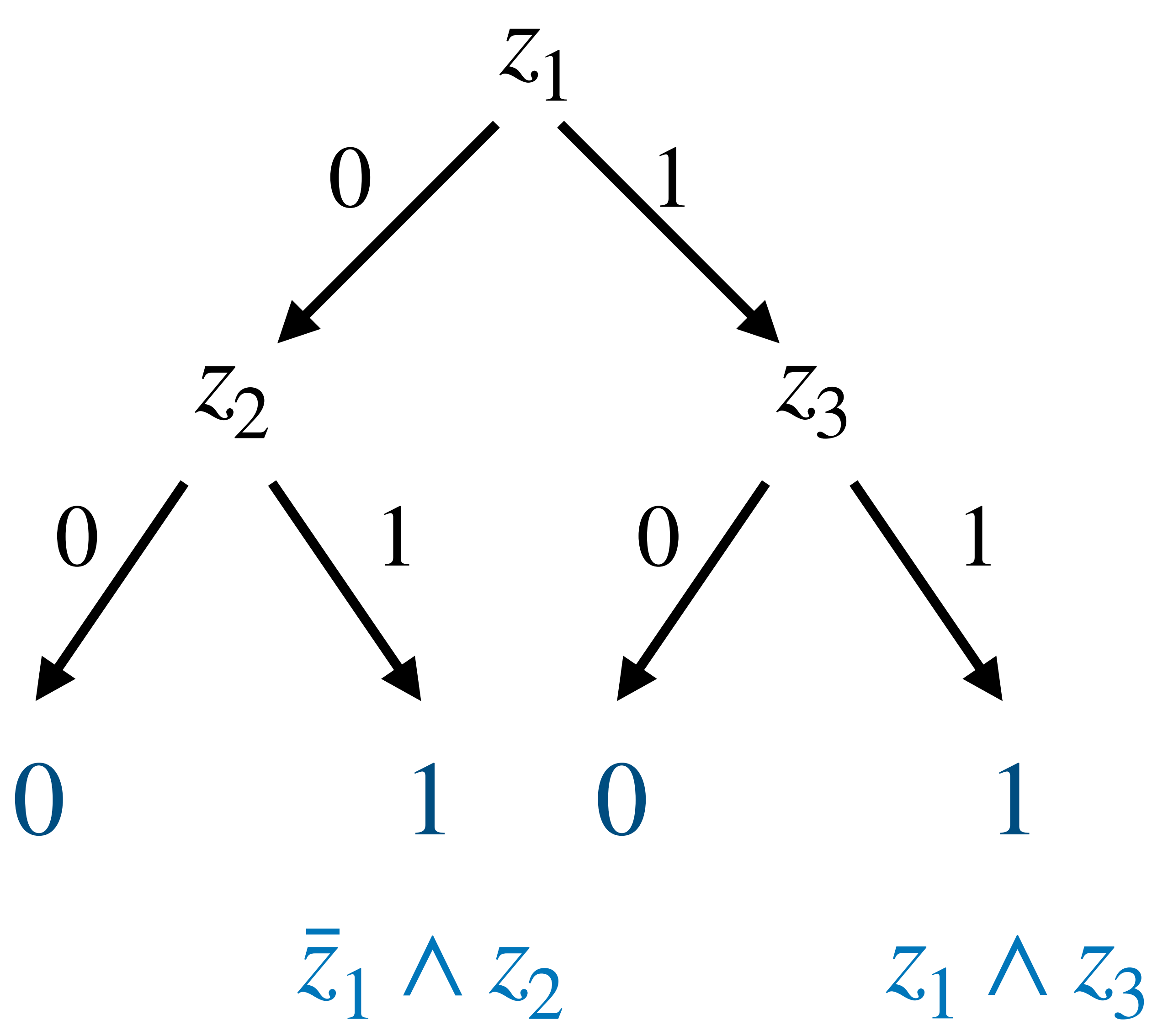
Decision tree with m leaves



polylog(m/ε)-wise indistinguishability ε/m-fools every 1-leaf

polylog-wise indistinguishable quadratic sources fool polynomial size decision trees

Decision tree with m leaves

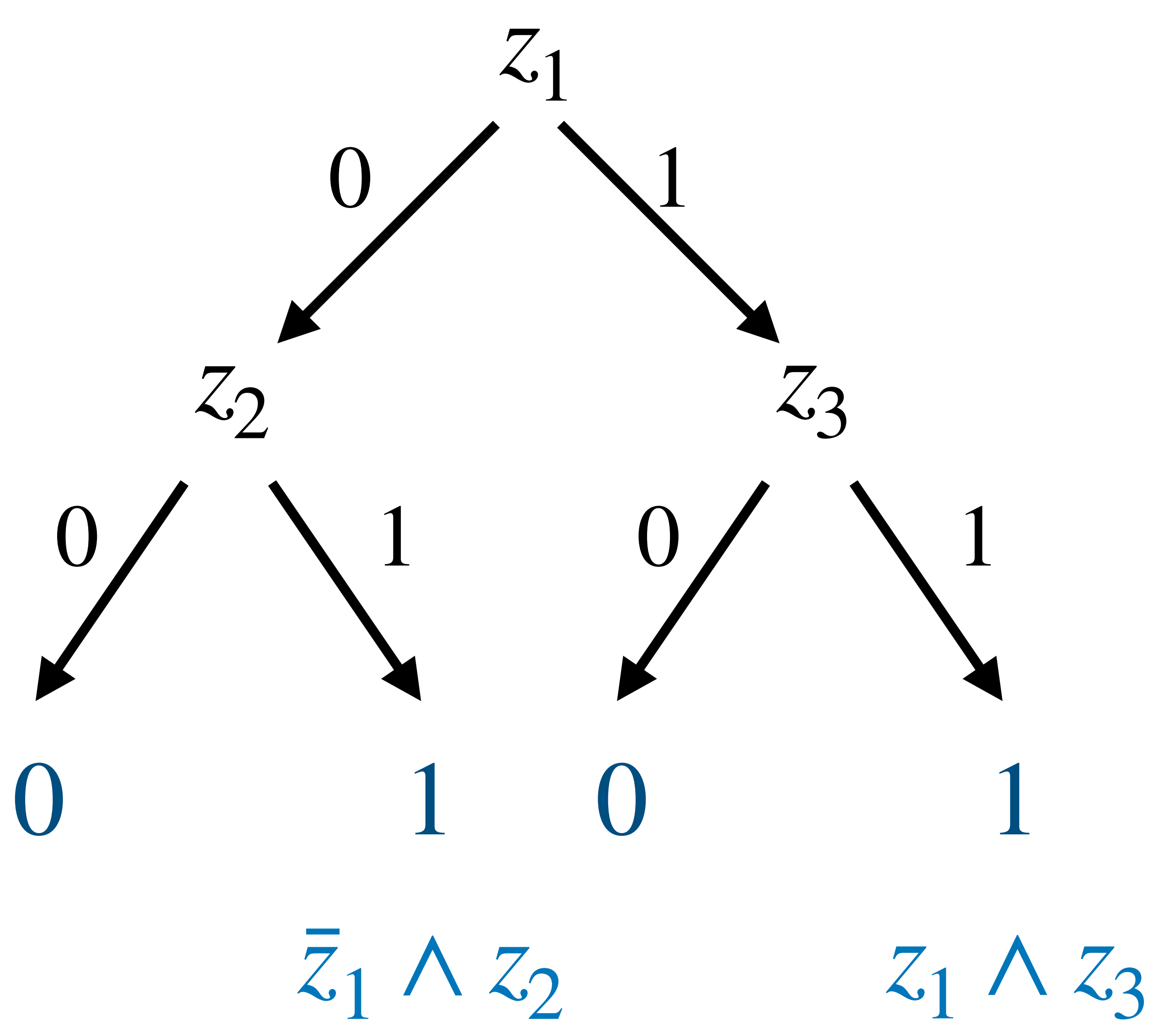


polylog(m/ε)-wise indistinguishability ε/m-fools every 1-leaf

$\varepsilon = \mathrm{poly}(1/n) \implies \mathrm{polylog}(n)$-wise indistinguishability ε-fools decision tree

polylog-wise indistinguishable quadratic sources fool polynomial size decision trees

Decision tree with m leaves



polylog(m/ε)-wise indistinguishability ε/m-fools every 1-leaf

$\varepsilon = \text{poly}(1/n) \implies \text{polylog}(n)$-wise indistinguishability ε-fools decision tree

Crucially relies on $k = \text{polylog}(1/\epsilon)$!

Open Questions

Open Questions

Beyond OR

Results on DNFs or AC^0? No barriers for local sources!

Open Questions

Beyond OR

Results on DNFs or AC^0? No barriers for local sources!

Application: secret-sharing with sharing in NC^0 and reconstruction in AC^0
(current best: sharing using decision trees and reconstruction using OR)

Open Questions

Beyond OR

Results on DNFs or AC^0? No barriers for local sources! **NC^0/AC^0 secret-sharing**

Web of conjectures

Given linear preprocessing $g_j(y)$, which parities of y are computable in AC^0?

Linear IPPP: not all — Our conjecture: short linear combinations — Equivalent?

Open Questions

Beyond OR

Results on DNFs or AC^0? No barriers for local sources! **NC^0/AC^0 secret-sharing**

Web of conjectures

Given linear preprocessing $g_j(y)$, which parities of y are computable in AC^0?

Linear IPPP: not all — Our conjecture: short linear combinations — Equivalent?

Conjectures about linear sources imply conjectures about quadratic sources?

Open Questions

Beyond OR

Results on DNFs or AC^0? No barriers for local sources! **NC^0/AC^0 secret-sharing**

Web of conjectures

Given linear preprocessing $g_j(y)$, which parities of y are computable in AC^0?

Linear IPPP: not all — Our conjecture: short linear combinations — Equivalent?

Conjectures about linear sources imply conjectures about quadratic sources?

More on OR

Best degree? (know: $O(\log n)$ and $\omega(1)$)

Best locality? (reduced precision implies locality 4; ruled out for mixture of iid)

Open Questions

Beyond OR

Results on DNFs or AC^0? No barriers for local sources! NC^0/AC^0 secret-sharing

Web of conjectures

Given linear preprocessing $g_j(y)$, which parities of y are computable in AC^0?

Linear IPPP: not all — Our conjecture: short linear combinations — Equivalent?

Conjectures about linear sources imply conjectures about quadratic sources?

More on OR

Best degree? (know: $O(\log n)$ and $\omega(1)$)

Best locality? (reduced precision implies locality 4; ruled out for mixture of iid)

Beyond Boolean

$(n-1)$-wise indistinguishable distributions over Σ^n distinguished by AC^0?

Connection to approximate degree breaks down

Open Questions

Beyond OR

Results on DNFs or AC^0? No barriers for local sources! **NC^0/AC^0 secret-sharing**

Web of conjectures

Given linear preprocessing $g_j(y)$, which parities of y are computable in AC^0?

Linear IPPP: not all — Our conjecture: short linear combinations — Equivalent?

Conjectures about linear sources imply conjectures about quadratic sources?

More on OR

Best degree? (know: $O(\log n)$ and $\omega(1)$)

Best locality? (reduced precision implies locality 4; ruled out for mixture of iid)

Beyond Boolean

$(n-1)$-wise indistinguishable distributions over $(\{0,1\}^n)^n$ distinguished by AC^0?

Application: secret sharing scheme in AC^0 with "sharp threshold"

$O(1)$-wise indistinguishable *simple* sources fool OR

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Sources samplable in locality s

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Sources samplable in locality s

If there are many X_i depending on disjoint random bits: done

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Sources samplable in locality s

If there are many X_i depending on disjoint random bits: done

Otherwise, we found small “hitting set” for entire source

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Sources samplable in locality s

If there are many X_i depending on disjoint random bits: done

Otherwise, we found small "hitting set" for entire source

Consider every possible setting of hitting set $\implies$ locality reduces to $s - 1$

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Sources samplable in locality s

If there are many X_i depending on disjoint random bits: done

Otherwise, we found small "hitting set" for entire source

Consider every possible setting of hitting set $\implies$ locality reduces to $s - 1$

Sources samplable in degree d

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Sources samplable in locality s

If there are many X_i depending on disjoint random bits: done

Otherwise, we found small "hitting set" for entire source

Consider every possible setting of hitting set $\Longrightarrow$ locality reduces to $s - 1$

Sources samplable in degree d

Use higher-order Fourier analysis to implement similar argument

$O(1)$-wise indistinguishable *simple* sources fool OR

Goal: Find S such that probability that $X|_S = 0$ but $X \neq 0$ is at most ε

Sources samplable in locality s

If there are many X_i depending on disjoint random bits: done

Otherwise, we found small "hitting set" for entire source

Consider every possible setting of hitting set $\Longrightarrow$ locality reduces to $s - 1$

Sources samplable in degree d

Use higher-order Fourier analysis to implement similar argument

Quadratic case ($d = 2$): dedicated argument gives better bounds