Low-Complexity Cryptography and Simple Hard-to-Learn Functions

Yuval Ishai

Technion

European
Research
Council

Average-Case Complexity:
From Cryptography to Statistical Learning
Simons Institute Workshop, 2021

This talk

- Cryptography and (hardness of) learning
- Low-complexity cryptography
- Low-complexity pseudorandom functions

What is Cryptography?

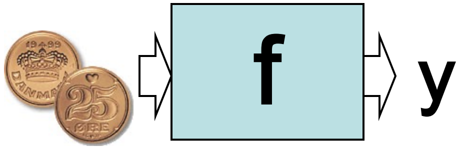
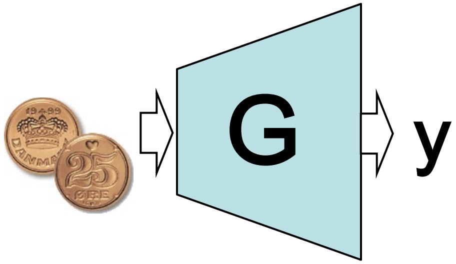
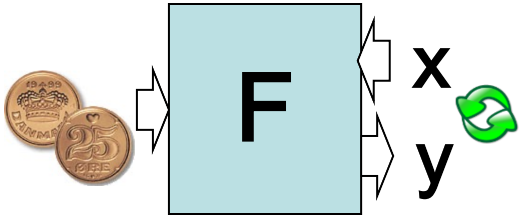
- Traditional definition:

  "THE PRACTICE AND STUDY OF TECHNIQUES FOR SECURE COMMUNICATION IN THE PRESENCE OF THIRD PARTIES."

- Broader definition:

  Allowing "good guys" to do G while preventing "bad guys" from achieving B.

Low-Level Primitives

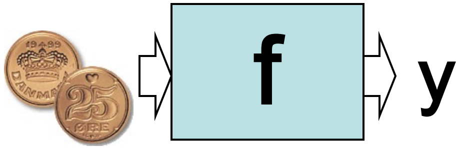
	G	B
OWF		find $x \in f^{-1}(y)$
PRG		distinguish y from a random string
PRF		distinguish F_k from a random function

Low-Level Primitives

G

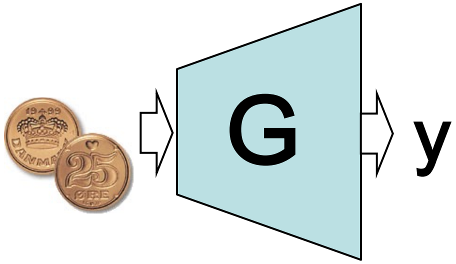
B

OWF



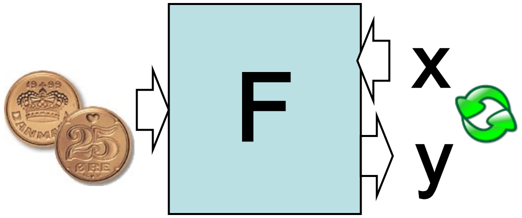
find $x \in f^{-1}(y)$

PRG



distinguish y from a random string

WPRF



distinguish F_k from a random function

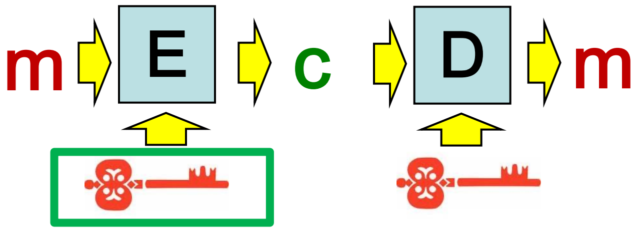
with uniformly random x_i

Higher-Level Primitives

G

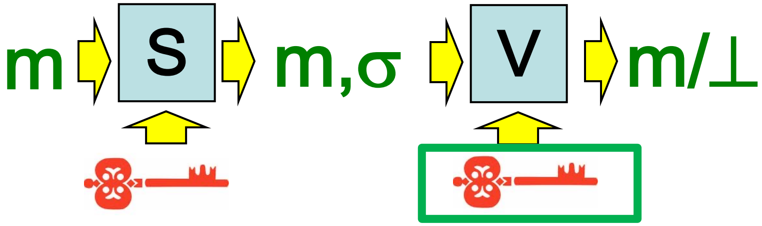
B

Encryption



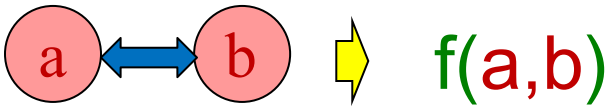
learn m from c

MAC / Signature



modify m

Secure Computation



learn input of other party from messages

Back to the 20th Century

Valiant ‘84:
A Theory of the Learnable

Introducing the PAC learning model

- Improper learning
- Distribution-free
- Approximate correctness

“Whether the classes of learnable Boolean concepts can be extended significantly … is an interesting question. There is circumstantial evidence from cryptography, however, that the whole class of functions computable by polynomial size circuits is not learnable.”

Goldreich-Goldwasser-Micali ‘87:
How to Construct Random Functions

Introducing Pseudo-Random Functions

- PRF construction from any one-way function
- Hard to learn!

“…one may c easy to
evaluate given porary
access to an o sumption
that one-way f

Even with:

- membership queries
- any high-entropy input distribution
- weak approximation guarantee

Goldreich-Goldwasser-Micali ‘87:
How to Construct Random Functions

Introducing Pseudo-Random Functions

- PRF construction from any one-way function
- Hard to learn!

Weak PRF:

Hard to learn under the uniform distribution

“…one may c … o
evaluate giver … y
access to an c … tion
that one-way functions exist.

Kearns-Valiant ‘89:
Cryptographic Limitations on Learning Boolean Formulae and Finite Automata

Hardness of learning simple functions based on standard cryptographic assumptions

- Decryption function is hard to learn
- Implement decryption in NC1, TC0

> “Our approach in this paper is based on refining the functions provided by cryptography in an attempt to find the simplest functions that are difficult to learn. … A technical open problem is to improve the constructions given here to … even simpler classes of formulae and circuits. ”

Blum-Furst-Kearns-Lipton ‘93:
Cryptographic Primitives Based on Hard Learning Problems

Apply hardness-of-learning conjectures towards simple cryptography

- Search-to-decision reduction for Learning Parity with Noise (LPN)
- WPRF candidate computable by poly-size DNF

$$f_{A,B}(x) = \mathrm{Parity}(x_A) \oplus \mathrm{Majority}(x_B) \quad |A| = |B| = \log n$$

“… as “simple” function classes … continue to elude efficient learning, our belief in the intractability of learning such classes increases, and we can exploit this intractability to obtain simpler cryptographic primitives.”

– WPRF candidate computable by poly-size DNF

$f_{A,B}(x) = \mathrm{Parity}(x_A) \oplus \mathrm{Majority}(x_B) \quad |A| = |B| = \log n$

Isn’t this cheating? Where’s the math?

“… [this is]a distribution on DNF formulas that seems to defy all known methods of attack, and we believe that any method that could even weakly predict such functions over a uniform D would require profoundly new ideas.”

– WPRF candidate computable by poly-size DNF

$f_{A,B}(x) = \mathrm{Parity}(x_A) \oplus \mathrm{Majority}(x_B) \quad |A| = |B| = \log n$

Isn’t this cheating? Where’s the math?

Well, suppose they are right.
Aren’t we done?

Only weak PRF
Only quasi-polynomial hardness

- WPRF candidate computable by poly-size DNF

  $f_{A,B}(x) = \mathrm{Parity}(x_A) \oplus \mathrm{Majority}(x_B) \quad |A| = |B| = \log n$

Isn’t this cheating? Where’s the math?

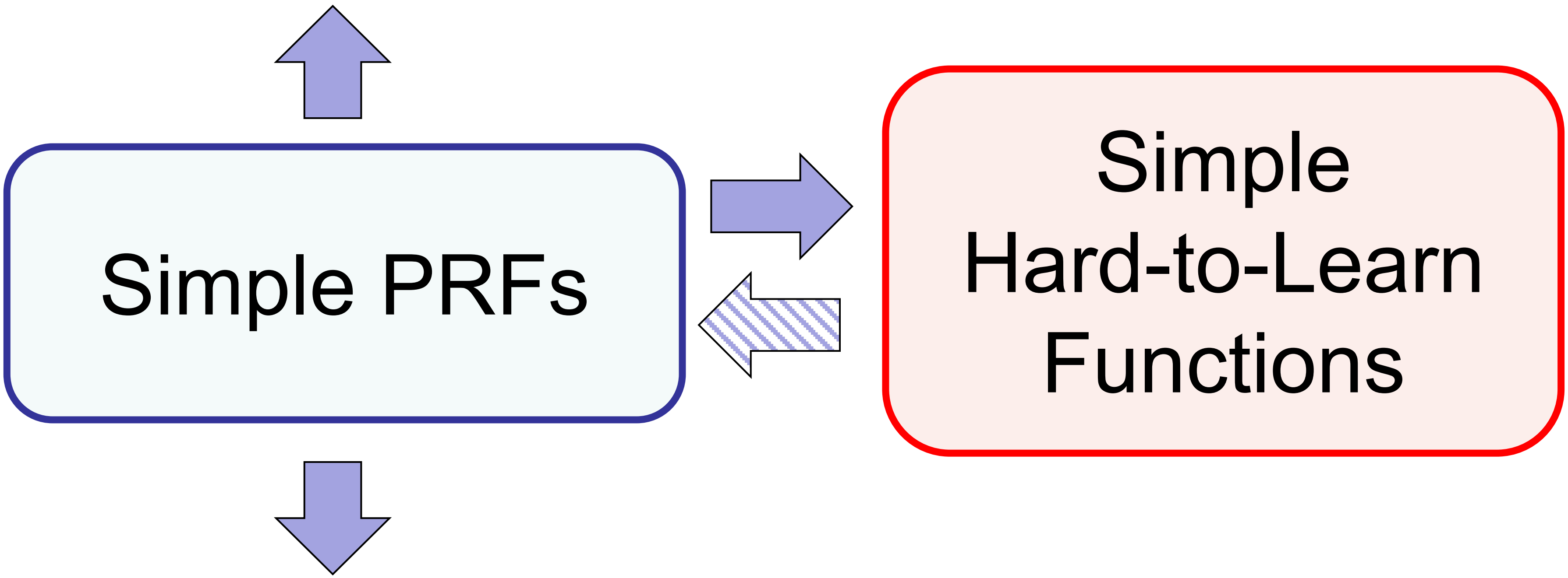
Well, suppose they are right.
done?

Both limitations inherent to AC0
[Linial-Mansour-Nisan 89]

Only weak PRF
Only quasi-polynomial hardness

Natural Proof Barriers

Different applications motivate different notions of simplicity



Simple PRFs

Simple Hard-to-Learn Functions

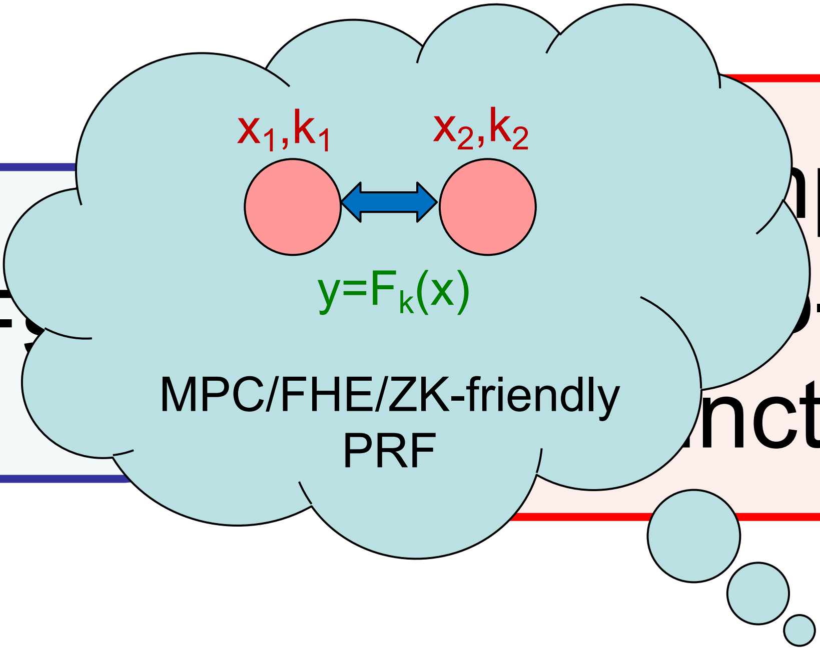
Cryptographic Applications

market for simple hard-to-learn functions

Natural Proof Barriers

Different applications motivate different notions of simplicity

Simple PRFs



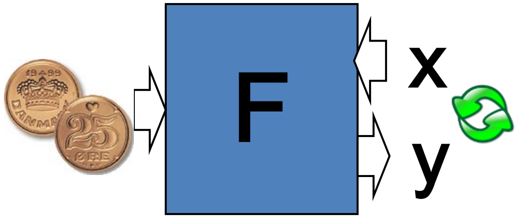
nple
-Learn
nctions

Cryptographic Applications

market for simple hard-to-learn functions

Motivating challenge: Asymptotically Optimal PRF

PRF



Hard to distinguish
from a random function

$F_k:\{0,1\}^n \rightarrow \{0,1\}^n$

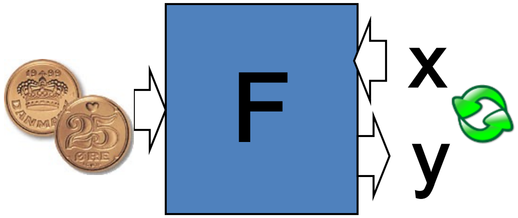
Efficiency: O(n)-size circuit

Security: $2^{\Omega(n)}$-size distinguishers

Any “provable” construction?

Motivating challenge: Asymptotically Optimal PRF

PRF



Hard to distinguish from a random function

$F_k:\{0,1\}^n \rightarrow \{0,1\}^n$

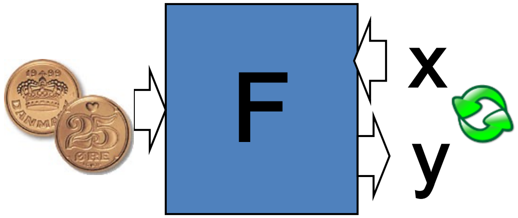
Efficiency: O(n)-size circuit

Security: $2^{\Omega(n)}$-size distinguishers

… or even heuristic?

Motivating challenge: Asymptotically Optimal PRF

PRF



Hard to distinguish from a random function

$F_k:\{0,1\}^n \rightarrow \{0,1\}^n$

Efficiency: O(n)-size circuit

Security: $2^{\Omega(n)}$-size distinguishers

Implies linear-time encodable codes…

Low-Complexity Cryptography

A very broad research agenda…

- Pick a crypto primitive
  - OWF, PRG, PRF, CRH, PKE, ZK, SNARG, MPC, FHE, HSS, ABE, IO,…
- Pick a target security level
  - Standard / sub-exponential / exponential? Post-quantum?
- Pick a complexity measure
  - Computation
    - Model: circuit, branching program, RAM, …
    - Metric: size, depth, …
  - Locality, algebraic degree
  - Communication, rounds
- Go as low as you can

What about assumptions?

- Typical methodology: build X under “acceptable” assumption Y
  - Notion of “acceptable” somewhat arbitrary

- No assumption? Certainly acceptable.

Information-Theoretic Cryptography

[BenOr-Kilian-Goldwasser-Wigderson 88]

IT-ZK => … PCP … => Practical ZK

What about assumptions?

- Typical methodology: build X under “acceptable” assumption Y
  - Notion of “acceptable” somewhat arbitrary

Drawing the line:

- Naor 03
  - Gentry-Wichs 11, Pass 11, …
- Goldwasser-Kalai 16
- …

What about assumptions?

- Typical methodology: build X under “acceptable” assumption Y
  - Notion of “acceptable” somewhat arbitrary

Typical “acceptable” assumptions:

- Clean and succinct
- Efficiently falsifiable
- Broadly applicable
- Win-win flavor
- Withstood test of time…

What about assumptions?

- Typical methodology: build X under “acceptable” assumption Y
  - Notion of “acceptable” somewhat arbitrary
  - In reality: “acceptable” aka “standard” = used by those we trust
  - Heavily influenced by historical coincidences

- What if this methodology fails?
  - When is it ok to make new assumptions?
  - Someone needs to be the first...

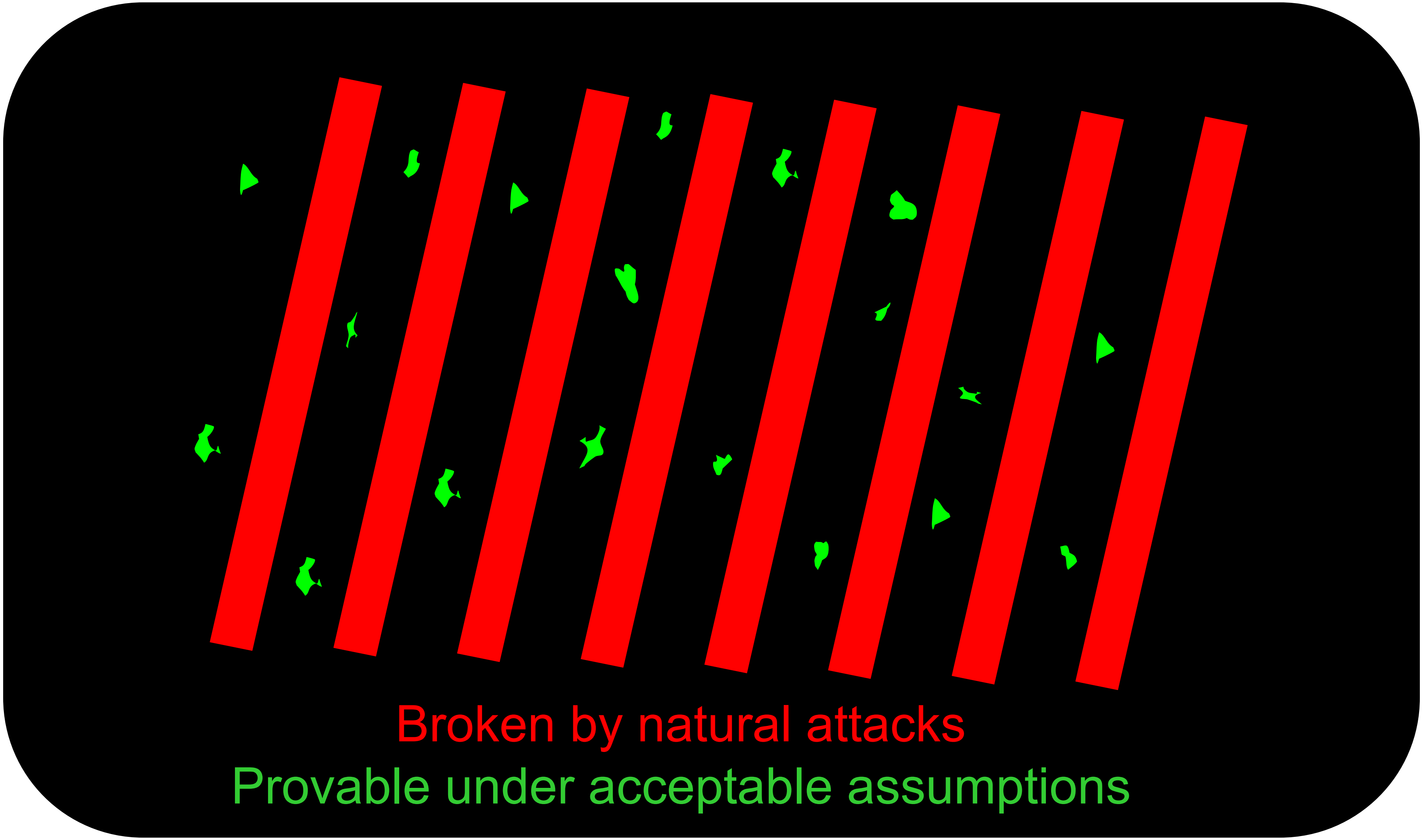
- Theory community tends to be conservative
  - Speculative new assumptions are often broken
  - Minimizing assumptions gave rise to a rich and deep theory

Alternative Methodology

1. Identify a class C of natural constructions
2. Identify a class A of natural attacks
3. Find efficient constructions from C resisting A
   - Often a combinatorial problem, with no inherent barriers
   - Systematic way for navigating "crypto dark matter"
   - May lead to new acceptable assumptions

- Common in applied crypto
  - Typically heuristic, not systematic, restricted to maximum security
- Less common in theory-oriented crypto
  - OWF, PRG [Goldreich00 … Applebaum-Lovett16 …]
  - PRF [Miles-Viola12 … Akavia-Bogdanov-Guo-Kamath-Rosen14 …]

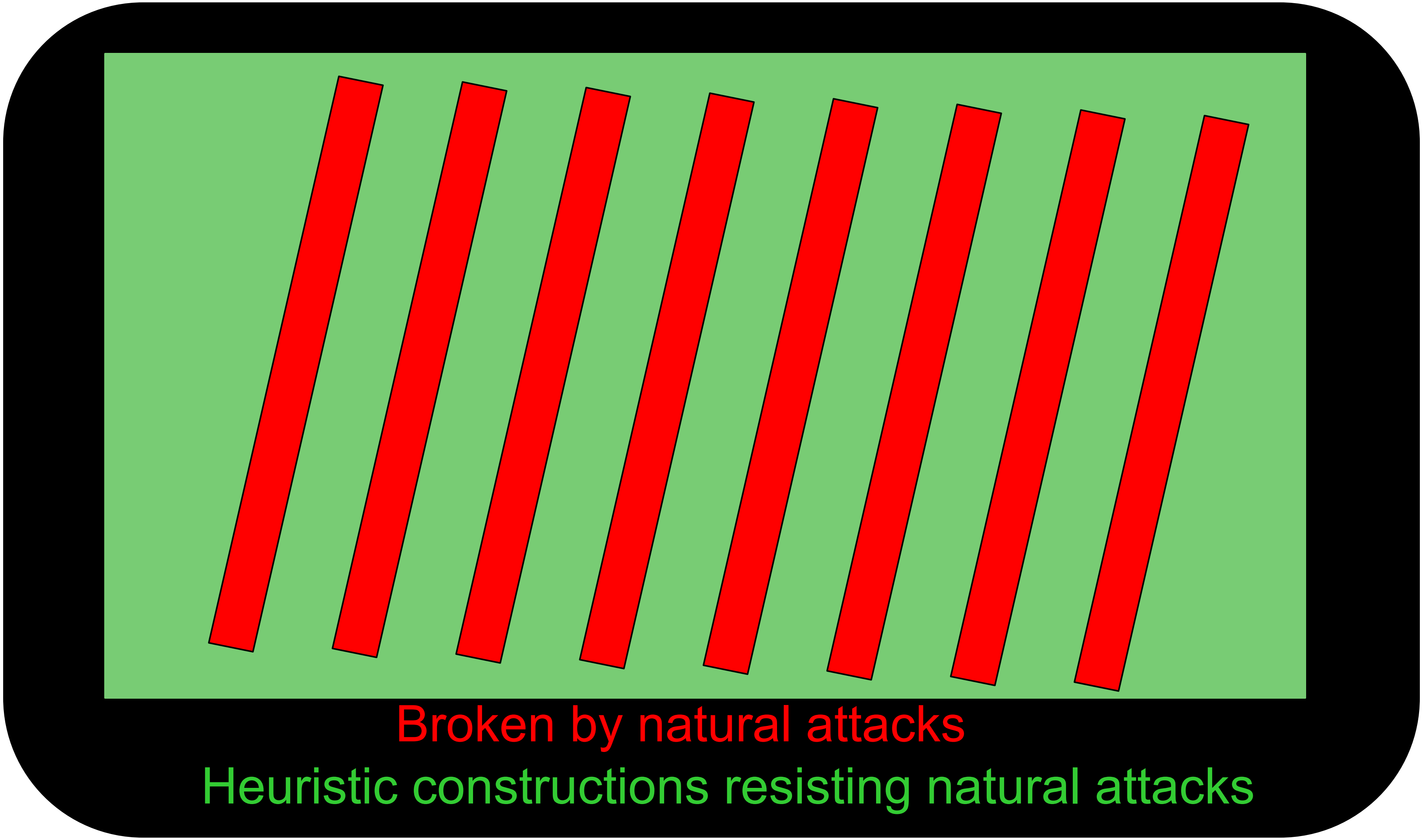
Crypto Universe



Broken by natural attacks

Provable under acceptable assumptions

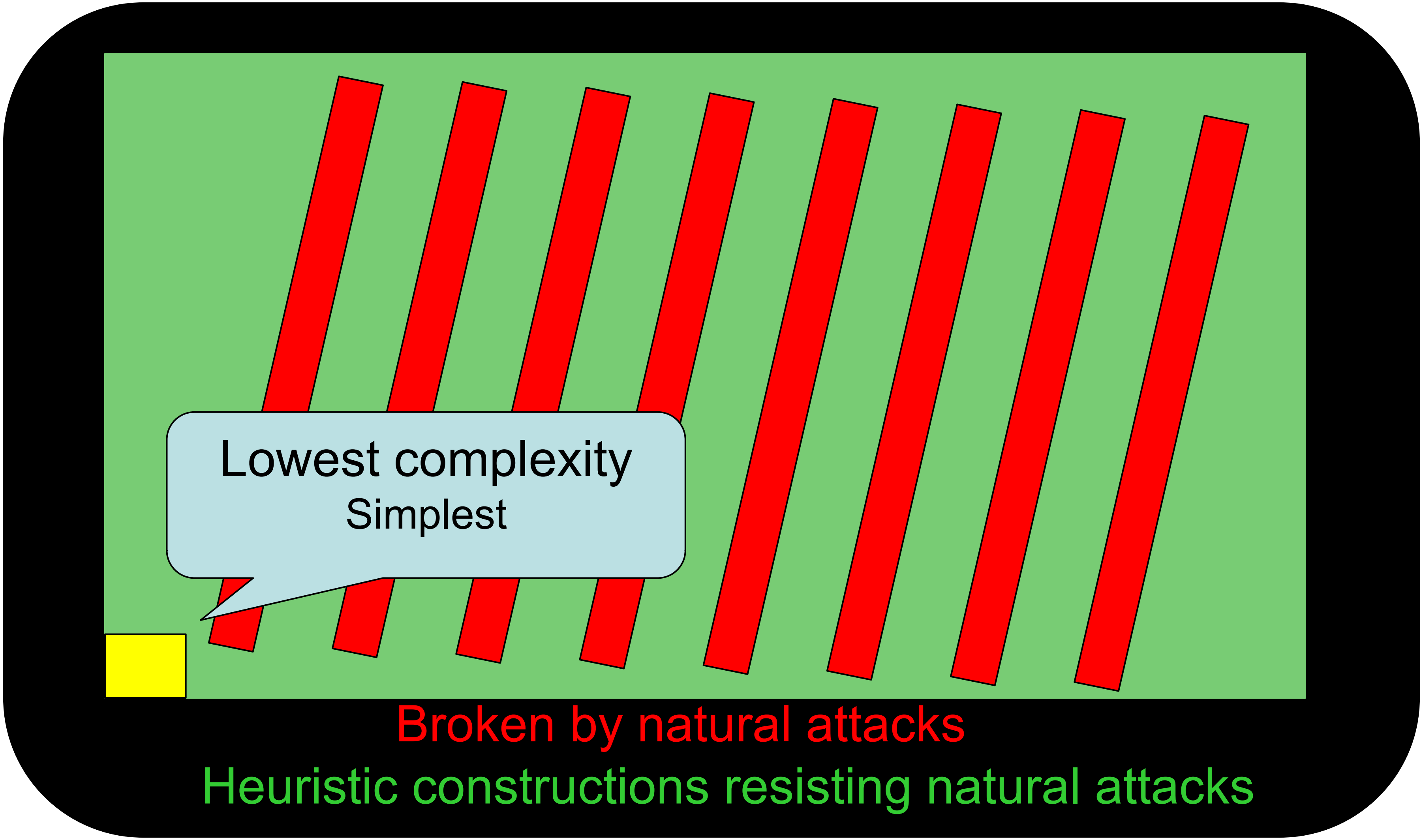
Crypto Universe



Broken by natural attacks

Heuristic constructions resisting natural attacks

Crypto Universe



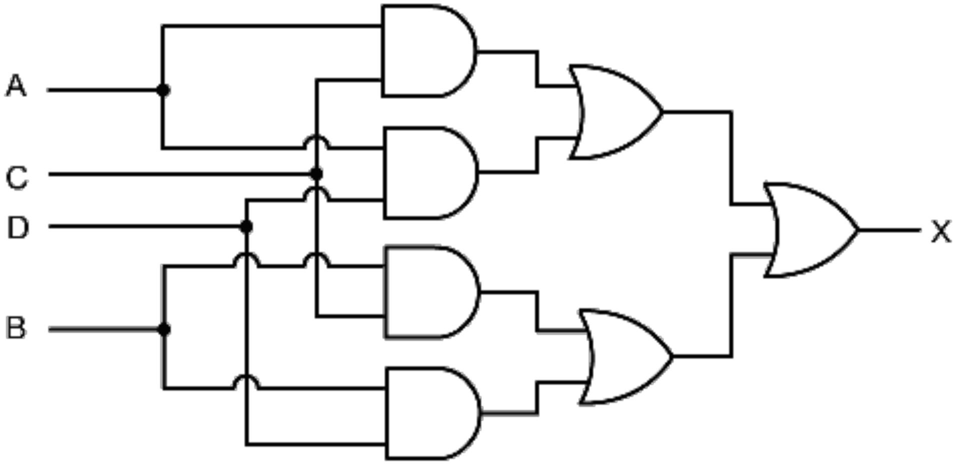
Lowest complexity

Simplest

Broken by natural attacks

Heuristic constructions resisting natural attacks

Computational Complexity of Cryptography



Default model:

boolean circuits with bounded fan-in

Minimizing Circuit Size

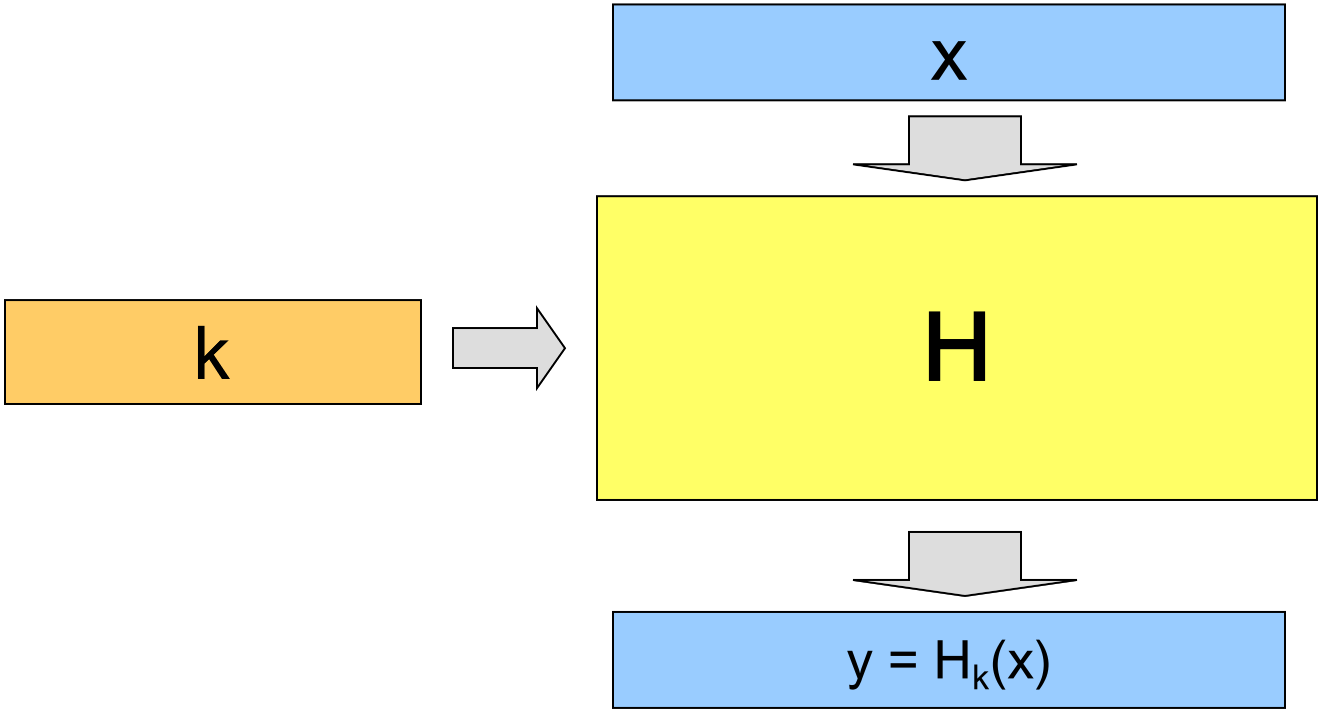
- λ = security parameter

	Insecure →	**Secure**
Typical:	s	s*poly(λ)
Dream goal….	s	O(s) i.e. O(s)+poly(λ)

Crypto with “constant overhead” ?

Universal Hashing

[Carter-Wegman77]

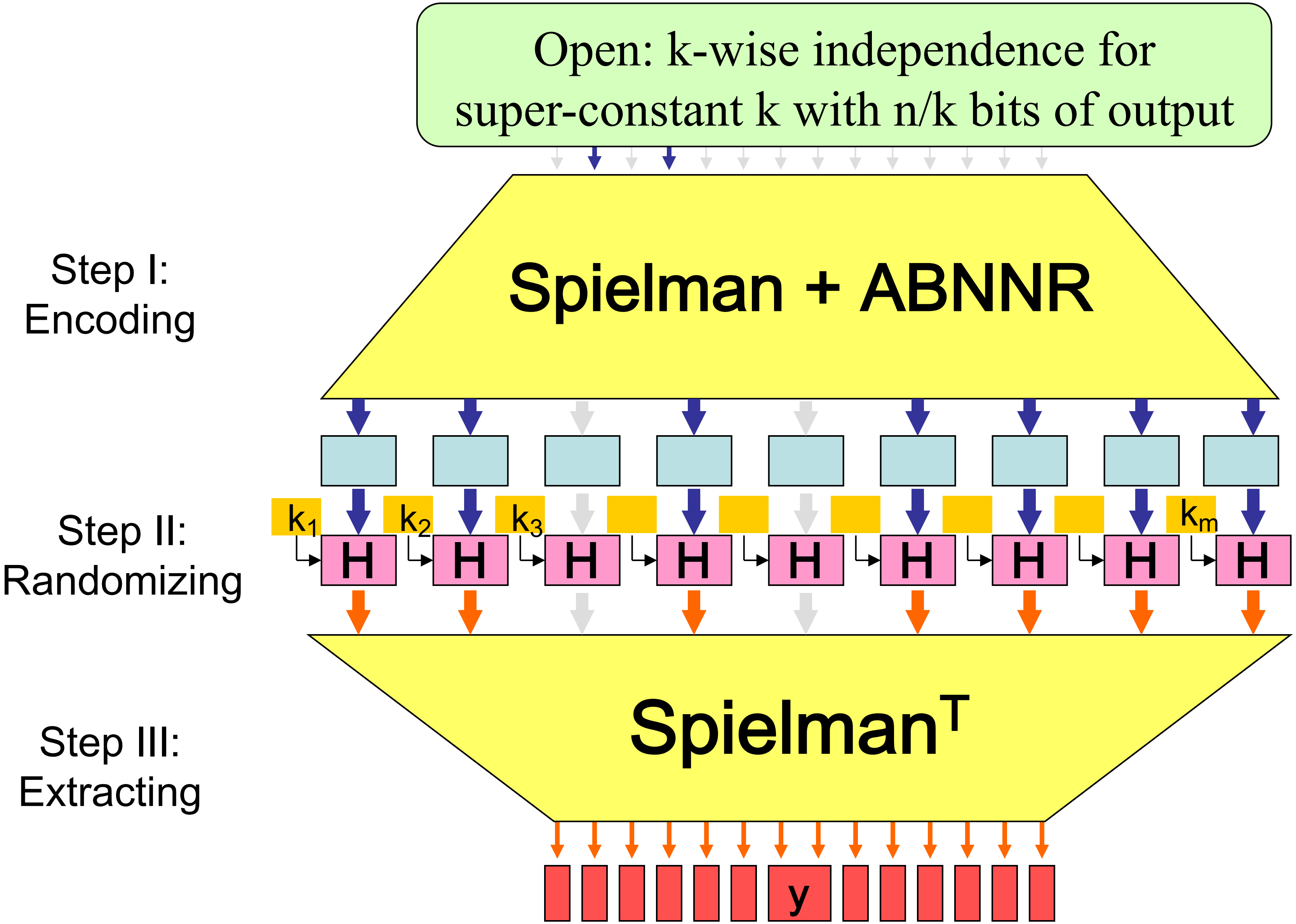


- Pairwise independence:
  - $x \neq x'$ ➔ $H_k(x), H_k(x')$ are uniform and independent

Complexity of Universal Hashing

- Standard constructions
  - $H_{a,b}(x)=ax+b$, $a,b \in GF(2^n)$
  - $H_{a,b}(x)=(a \circ x)+b$ $a \in Z_2^{2n-1}$, $b \in Z_2^n$
  - Both conjectured to require $\Omega(n \cdot \log n)$ circuit size
- [Mansour-Nisan-Tiwari 90]
  - Time-space tradeoff for universal hashing
  - Conjecture: Any universal hash function $H_k:\{0,1\}^n \rightarrow \{0,1\}^n$ requires circuits of size $\Omega(n \cdot \log n)$.
- [I-Kushilevitz-Ostrovsky-Sahai 08]
  - Can be done by linear-size circuits

Linear-Size Circuit for Hashing



Open: k-wise independence for super-constant k with n/k bits of output

Step I: Encoding

Step II: Randomizing

Step III: Extracting

Back to Coding Theory

[Druk-I 14]

- Family of linear-time encodable linear codes meeting the Gilbert-Varshamov bound
  - Efficient decoding?
  - Most likely not…
- … so back again to crypto
  - Linear-time substitute for random linear codes

Constant-Overhead Cryptography

Assumption	Primitive
none	Universal hashing One-time MAC
OWF	MAC “Shrinking” PRF
Lin-stretch local PRG	PRF, PKE Signatures
Poly-stretch local PRG	Secure Computation with semi-honest parties

Constant-Overhead Cryptography

Assumpti	Primitive
none	Universal hashing One-time MAC
OWF	MAC “Shrinking” PRF
Lin-stretch local PRG	PRF, PKE Signatures
Poly-stretch local PRG	Secure Computation with semi-honest parties

[Fan-Li-Yang 21]:

Circuit size 2n (over full basis) is sufficient and necessary!

Constant Overhead for Other Primitives

Assumption	Primitive
Binary-SVP [Applebaum-Haramaty-I-Kushilevitz-Vaikuntanathan17]	Collision-Resistant Hashing?
Exp-secure Local OWF [Baron-I-Ostrovsky16]	Exp-secure TDF? PRG?
New Candidate [Boneh-I-Passelègue-Sahai-Wu18]	Exp-secure PRF?
No candidate	Zero-knowledge proofs? Succinct arguments?
No candidate	Secure computation with malicious parties?

Constant ves

Applebaum 17:

Implies gap-ETH

[Dinur 16; Manurangsi-Raghavendra 16]

Assumpt

Binary-SVP

[Applebaum-Haramat

Kushilevitz-Vaikunta athan17]

Collision-Resistant Hashing?

Exp-secure Local OWF

[Baron-I-Ostrovsky16]

Exp-secure TDF? PRG?

New Candidate

[Boneh-I-Passelègu Wu18]

Exp-secure PRF?

Natural proof barrier for linear-size circuits

(Previously: quasi-linear size candidate [Miles-Viola12])

Later in the talk…

No

oofs?

ts?

No

on

with malicious parties?

Constant Overhead for Other Primitives

Assumption	Primitive
Binary-SVP [Applebaum-Haramaty-I-Kushilevitz-Vaikuntanathan17]	Collision-Resistant Hashing?
Exp-secure Local OWF [Baron-I-Ostrovsky16]	Exp-secure TDF? PRG?
New Candidate	Exp-secure PRF?
	Zero-knowledge proofs? Succinct arguments?
	Secure computation with malicious parties?

- Yes for arithmetic circuits

[Bootle-Cerulli-Ghadafi-Groth-Hajiabadi-Jakobsen17]
[Applebaum-Damgård-I-Nielsen-Zichron17]
[Boyle-Couteau-Gilboa-I18, Chase-Dodis-I-Kraschewski-Liu-Ostrovsky-Vaikuntanathan19]

- Best overhead for Boolean: polylog(λ)

[Damgård-I-Krøigaard10]

Low-Complexity Pseudorandom Functions

Taxonomy of Constructions

- Security type
  - Weak vs. Strong
- Security level
  - Polynomial, Quasipolynomial, Subexponential, Exponential
- Complexity class
  - Constant-depth poly-size circuits with unbounded fan-in
    - AC0: AND/OR/NOT
    - AC0[mod_p]: + parity / mod_p for prime p
    - ACC0: + mod_m for composite m
  - Linear-size circuits
- Assumptions
  - Standard, heuristic

Taxonomy of Constructions

- Security type
  - Weak vs. Strong
- Security level
  - Polyn l, Subexponential, Exponential
- Complexi y class
  - Constant-de unbounded fan-in
    - AC0: AND/
    - AC0[mod_p]: + parity / mod_p for prime p
    - ACC0: + mod_m for composite m
  - Linear-si e circuits
- Assum
  - Star

Viewing key k as fixed

No strong PRFs with better than qpoly security [RR94]

TC0:
Strong PRFs under standard cryptographic assumptions [Naor-Reingold 97, ...]

Taxonomy of Constructions

- Security type
  - Weak vs. Strong
- Security level
  - Polynomial, Quasipolynomial, Subexponential, Exponential
- Complexity class
  - Constant-depth poly-size circuits with unbounded fan-in
    - AC0: AND/OR/NOT
    - AC0[mod_p]: + parity / mod_p for prime p
    - ACC0: + mod_m for co
  - Linear-size circuits
- Assumptions
  - Standard, heuristic

Typically: Provable security against "relevant" attacks: linear, algebraic, ...

AC0

- Limitations [LMN89]
  - No strong PRF
  - Quasi-polynomial attack against WPRF

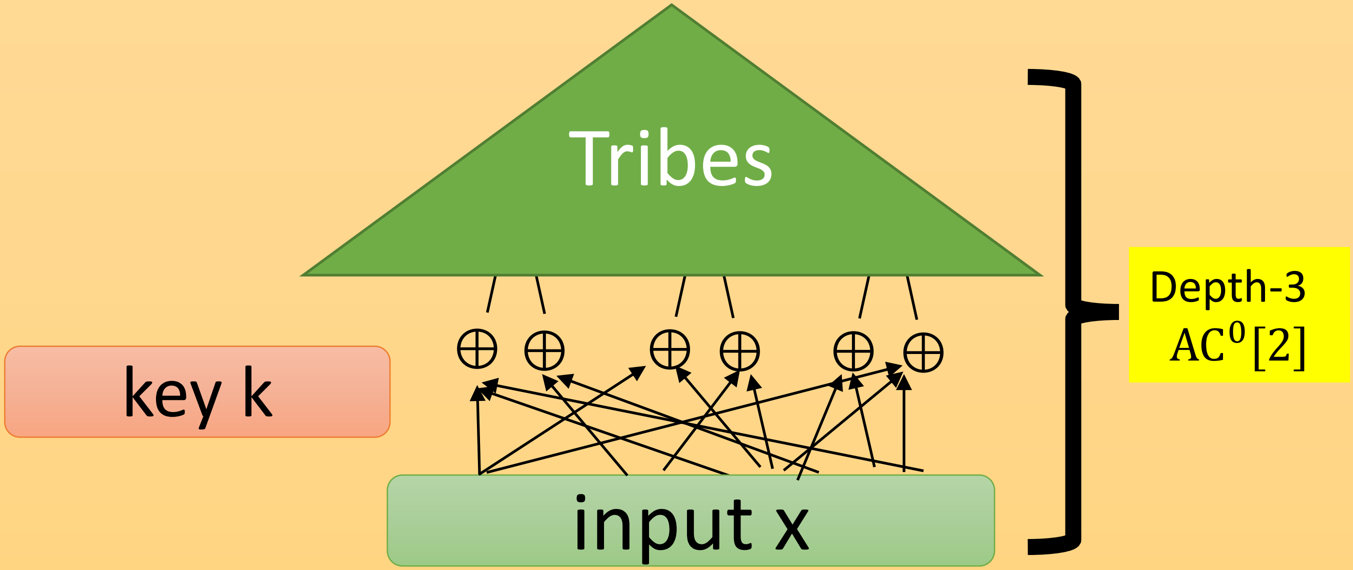
- Depth 2
  - WPRF candidate [BFKL93]
  - “Biased-input” WPRF from local PRG [Applebaum-Barak-Wigderson 10, Daniely-Vardi 21]

- Depth 3
  - WPRF from local PRG [Applebaum-Raykov 16, DV21]

AC0 on top of parities?

WPRF Candidate

[Akavia-Bogdanov-Guo-Kamath-Rosen14]

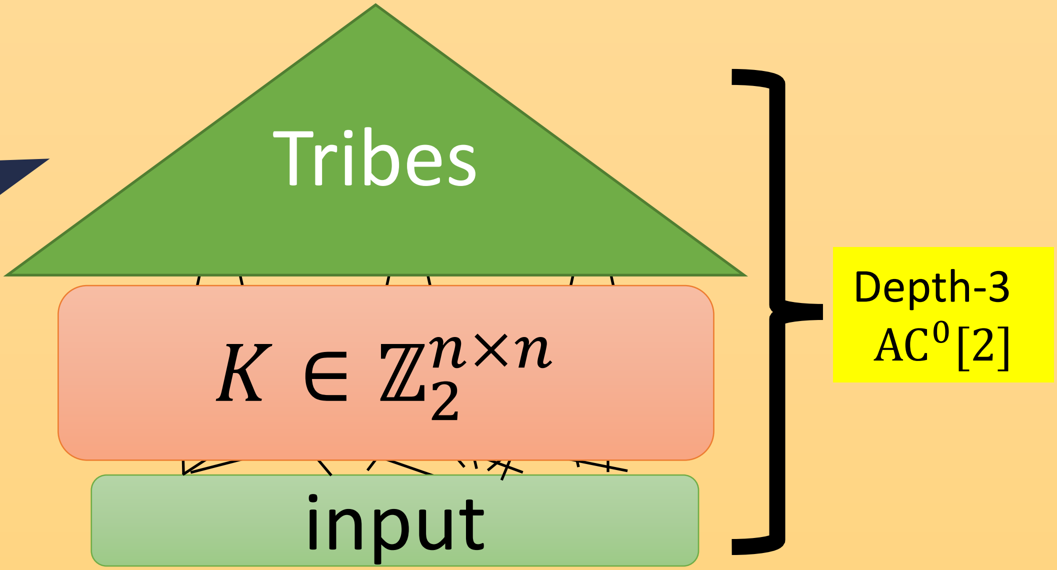


AC0 on top of parities?

WPRF Candidate

[Akavia-Bogdanov-Guo-Kamath-Rosen14]

[Bogdanov-Rosen 17]: quasi-polynomial time algebraic attack via low rational degree

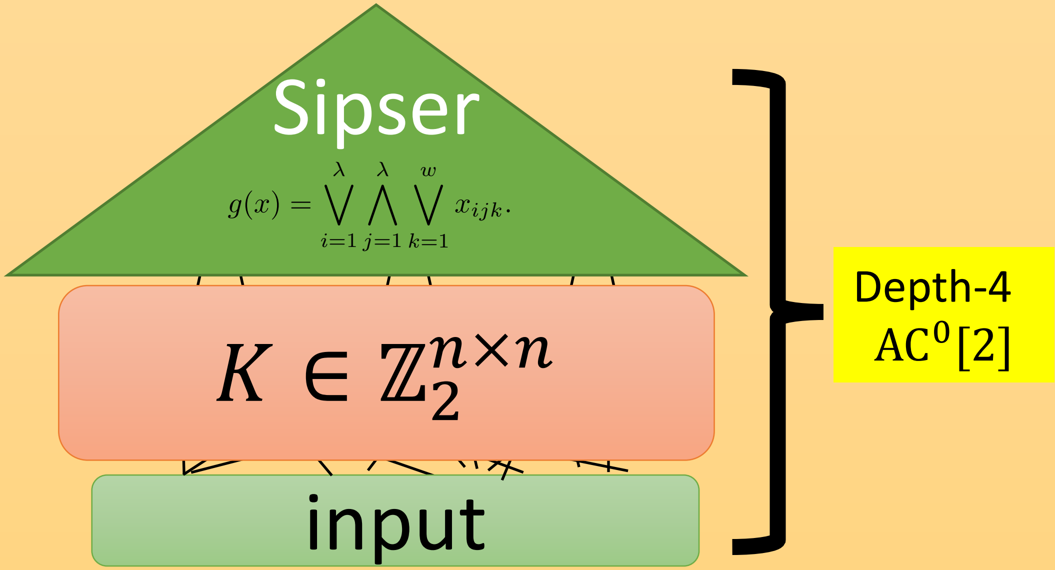


Take 2

WPRF Candidate

[Boyle-Couteau-Gilboa-I-Kohl-Scholl 21]

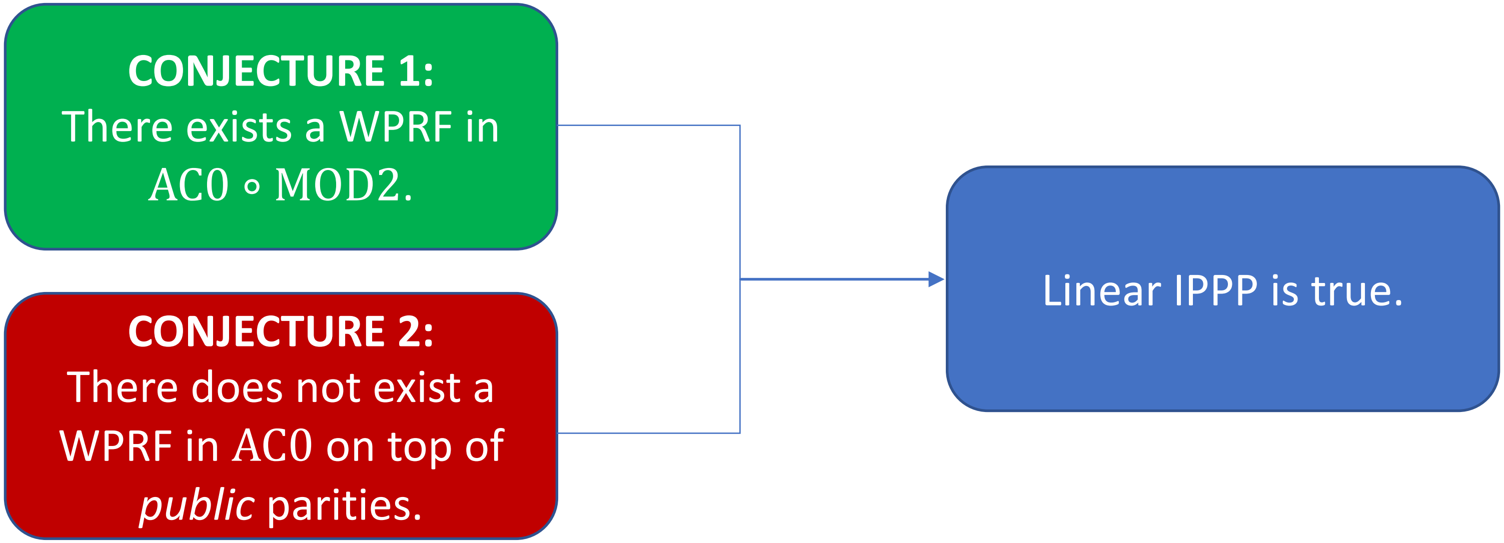
Provably high rational degree



AC0 on top of public parities?

[BCGIKS21]:
WPRF ruled out by a variant of a conjecture from [ABGKR14].

Linear IPPP conjecture [Servedio-Viola 12]:
Inner-product mod 2 cannot be computed in $\mathrm{AC0} \circ \mathrm{MOD2}$.



CONJECTURE 1:
There exists a WPRF in $\mathrm{AC0} \circ \mathrm{MOD2}$.

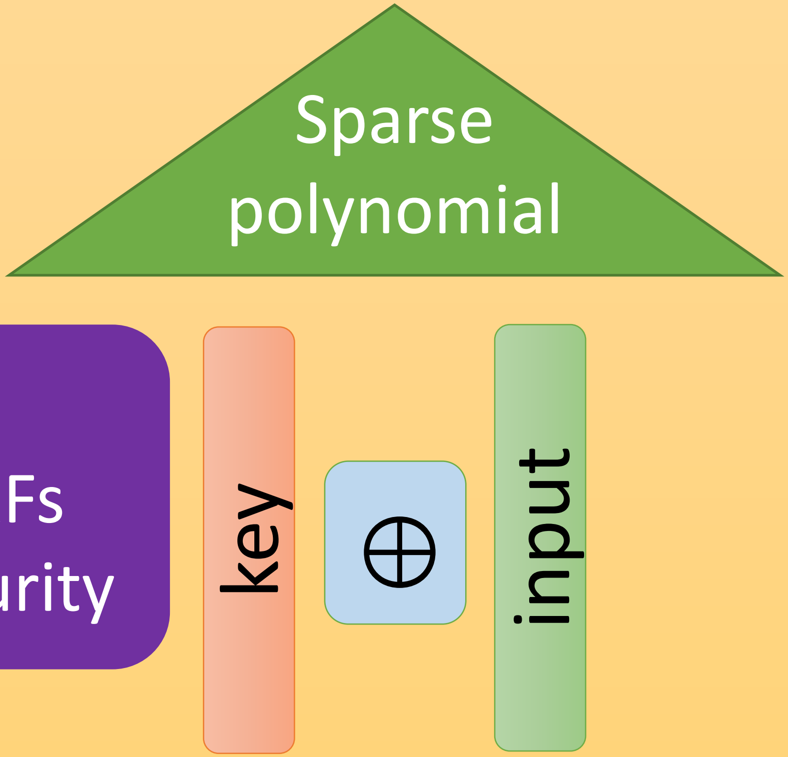
CONJECTURE 2:
There does not exist a WPRF in $\mathrm{AC0}$ on top of *public* parities.

Linear IPPP is true.

Depth-2 WPRF?

Candidate WPRF by XNF formulas

[Boyle-Couteau-Gilboa-I-Kohl-Scholl 20]



Applications:
- Correlated PRFs
- XOR-RKA security

Depth-2 WPRF?

Candidate WPRF by XNF formulas

[Boyle-Couteau-Gilboa-I-Kob ll 20]

Sparse multivariate $\mathbb{F}_2$-polynomials in inputs *and their negation*

Secure under variable-density variant of LPN

Applications:
- Correlated PRFs
- XOR-RKA security

Best possible security: $2^{\sqrt{n}}$

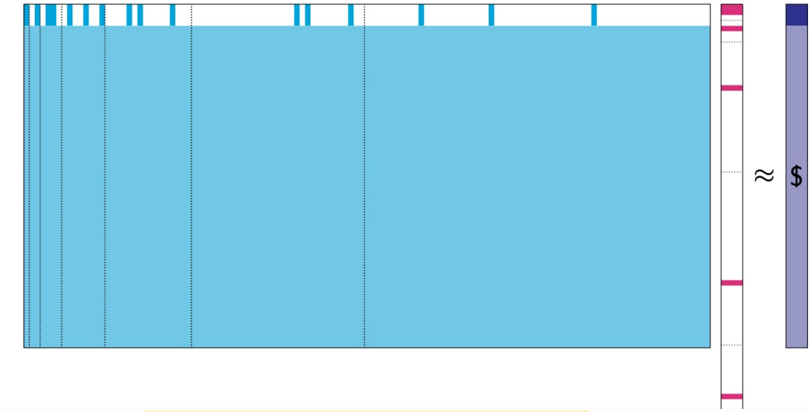
[Hellerstein-Servedio 07]

WPRF by XNF

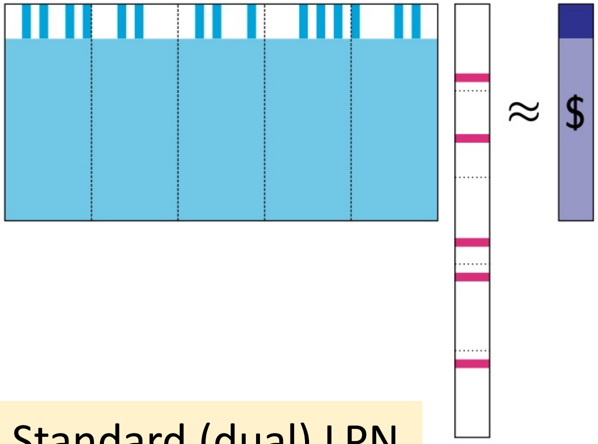
Bigger j ➔
more bias towards 0

$$f_K(x) = \bigoplus_{i=1}^{w} \bigoplus_{j=1}^{m} \overbrace{\bigwedge_{k=1}^{j} (x_{ijk} \oplus K_{ijk})}$$

Intuition: With more samples, more of these terms will "kick in"



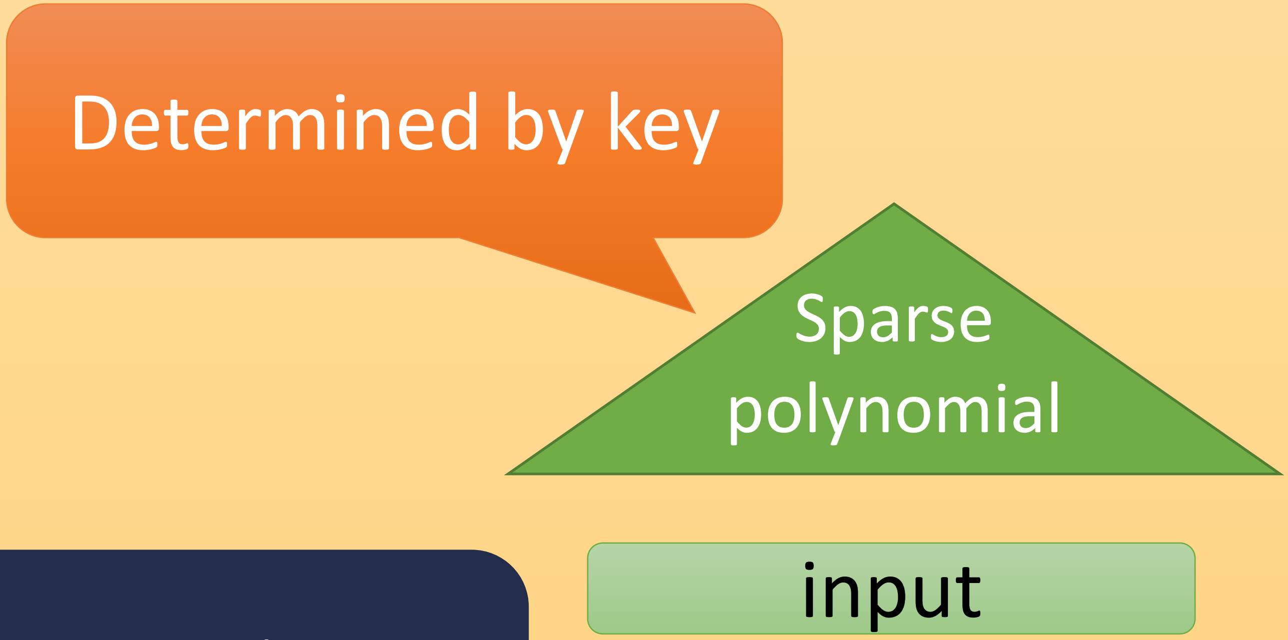
Variable-density LPN



Standard (dual) LPN

WPRF by sparse F_2-polynomials

[Boyle-Couteau-Gilboa-I-Kohl-Scholl 21]



Determined by key

Sparse polynomial

input

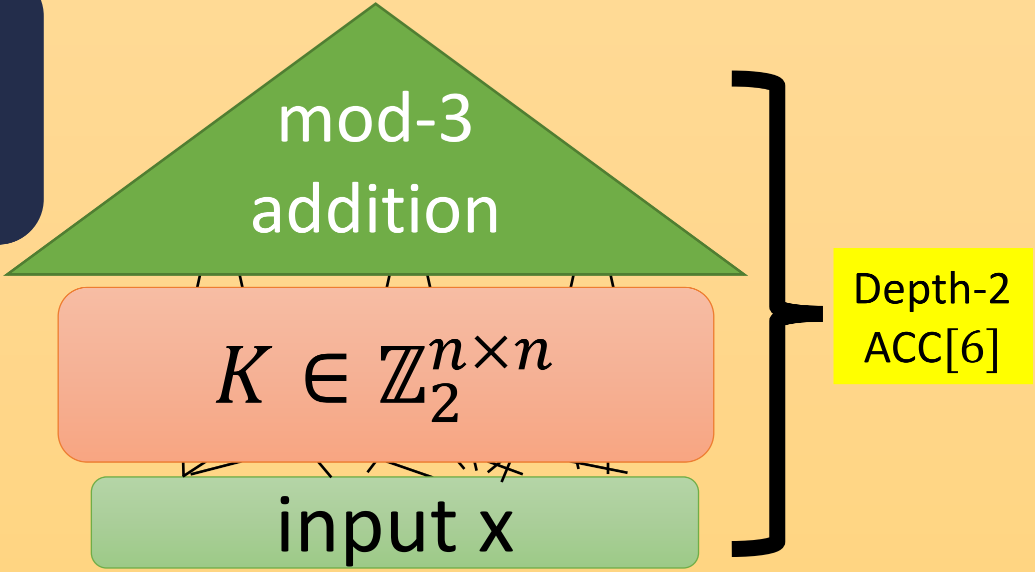
Subexponential security against linear and algebraic attacks

Mixing Moduli

[Boneh-I-Passelègue-Sahai-Wu 18]

WPRF candidate in ACC0

Conjecture:
Exponential security



Mixing Moduli

[Boneh-I-Passelègue-Sahai-Wu 18]

WP

So far withstood analysis
[Cheon-Cho-Kim-Kim 21]
[Dinur-Goldfeder-Halevi-I-Kelkar-Sharma-Zaverucha 21]

Conjecture:
Exponential security

mod-3

- Exponential hardness of learning $\text{mod}_3 \circ \text{XOR}$ circuits under uniform
- Same for $\text{FORMULA}[n^{2.8}] \circ \text{XOR}$
  [Kabanets-Koroth-Lu-Myrisiotis-Oliviera 20]

Mixing Moduli

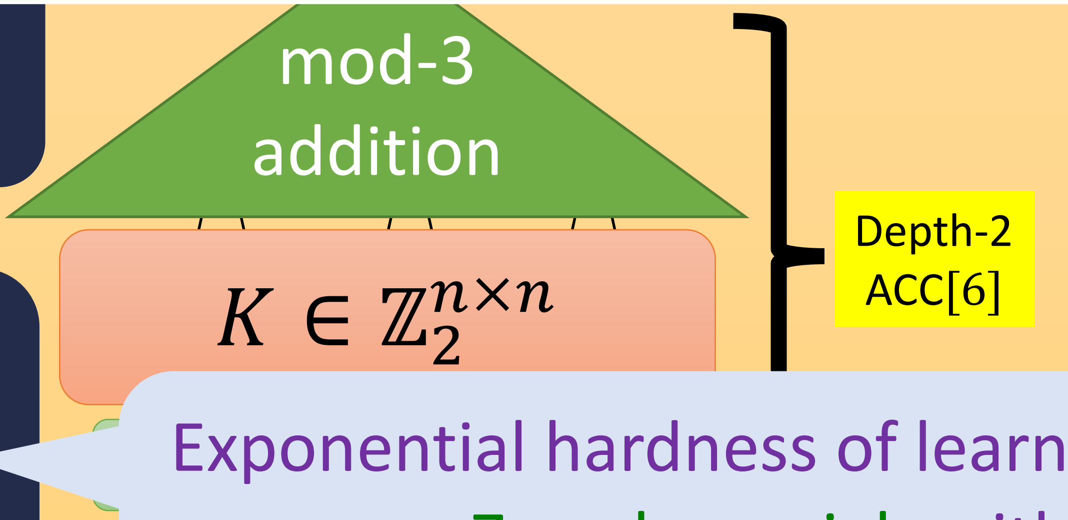
[Boneh-I-Passelègue-Sahai-Wu 18]

So far withstood analysis

[Cheon-Cho-Kim-Kim 21]

[Dinur-Goldfeder-Halevi-I-Kelkar-Sharma-Zaverucha 21]

Conjecture:
Exponential security



Also computable by:
* Sparse Z_3 polynomial
* Width-3 BP

Exponential hardness of learning sparse Z_3-polynomials with uniform inputs from $\{-1,1\}^n$

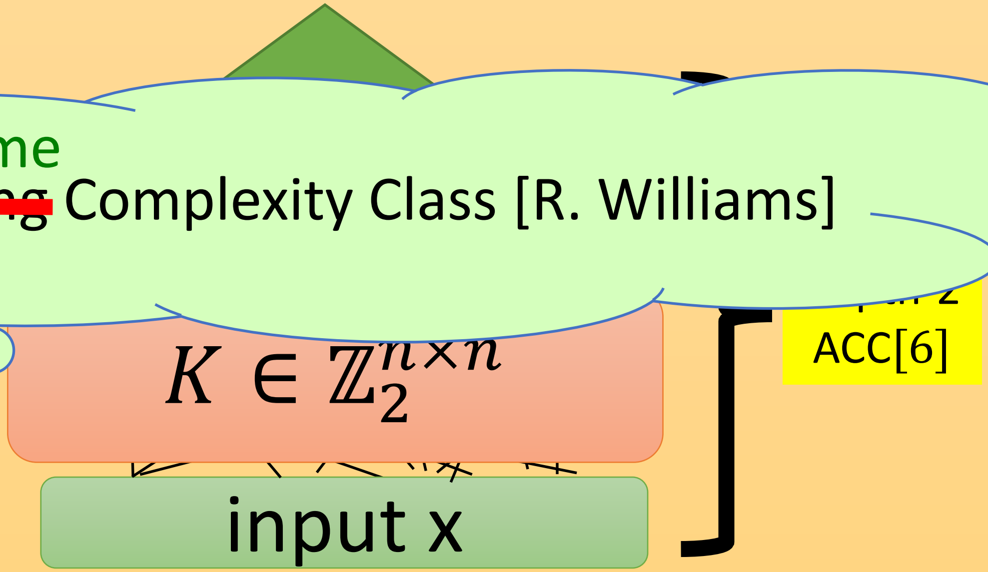
Mixing Moduli

[Boneh-I-Passelègue-Sahai-Wu 18]

WPRF candidate in ACC0

Conjecture:
Exponentia

Awesome
~~Annoying~~ Complexity Class [R. Williams]

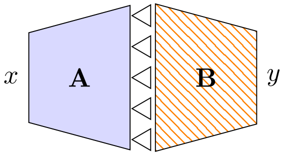


Easy to distribute!

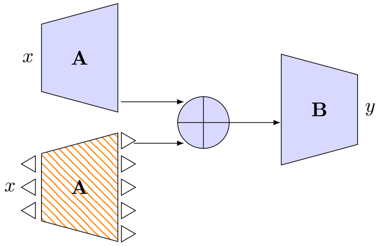
Fast Distributed Symmetric Crypto

[Dinur-Goldfeder-Halevi-I-Kelkar-Sharma-Zaverucha 21]

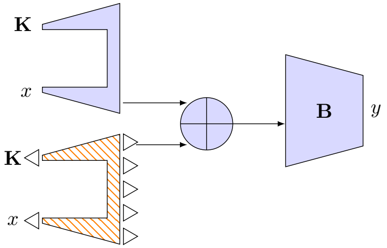
Candidates



(2, 3)-OWF



LPN-PRG



LPN-wPRF

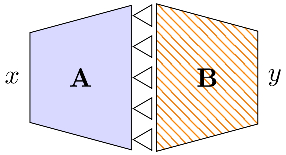
Analysis

Construction	Parameters (n, m, t)	Comment
(2, 3)-OWF	$(s, 3.13s, s/\log 3)$	aggressive
	$(s, 3.53s, s/\log 3)$	conservative
(2, 3)-wPRF	$(2s, 2s, s/\log 3)$	aggressive
	$(2.5s, 2.5s, s/\log 3)$	conservative
LPN-PRG	$(s, 3s, 2s)$	
LPN-wPRF	$(2s, 2s, s)$	

Protocols

Primitive	Construction	Param. (n, m, t)	Distributed 2PC (with preprocessing)		Distributed 3PC	Public-Input 2PC (with preprocessing)	
			Online Comm.	Prepr.	Online Comm.	Online Comm.	Prepr.
wPRF	(2, 3)-wPRF	(256, 256, 81)	(1536, 4, 2)	(2348, 662)	(1430, 4, 1)	(512, 2, 1)	(1324, 406)
	LPN-wPRF	(256, 256, 128)	(2860, 6, 3)	(4995, 1730)		(1324, 4, 2)	(3160, 918)
OWF	(2, 3)-OWF	(128, 452, 81)	(904, 2, 1)	(2337, 717)	(2525, 4, 1)	-	-
PRG	LPN-PRG	(128, 512, 256)	(1880, 4, 2)	(4334, 1227)		-	-

Practical post-quantum signatures



$(2,3)$-OWF

OWF Params (n,m,t)	KKW params (N,M,τ)	Sig. size (KB)
$(128,453,81)$	$(16,150,51)$	13.30
	$(16,168,45)$	12.48
	$(16,250,36)$	**11.54**
Picnic3-L1	$(16,250,36)$	12.60
$(128,453,81)$	$(64,151,45)$	13.59
	$(64,209,34)$	11.70
	$(64,343,27)$	**10.66**
Picnic2-L1	$(64,343,27)$	12.36

OWF Params (n,m,t)	KKW params (N,M,τ)	Sig. size (KB)
$(256,906,162)$	$(16,324,92)$	50.19
	$(16,400,79)$	47.08
	$(16,604,68)$	**45.82**
Picnic3-L5	$(16,604,68)$	48.72
$(256,906,162)$	$(64,322,82)$	51.23
	$(64,518,60)$	44.04
	$(64,604,57)$	**43.45**
Picnic2-L5	$(64,604,58)$	46.18

Table 4: Signature size estimates for Picnic using $(2,3)$-OWF, compared to Picnic using LowMC. The left table shows security level L1 (128 bits) with $N = 16$ and $N = 64$ parties, and the right table shows level L5 (256 bits).

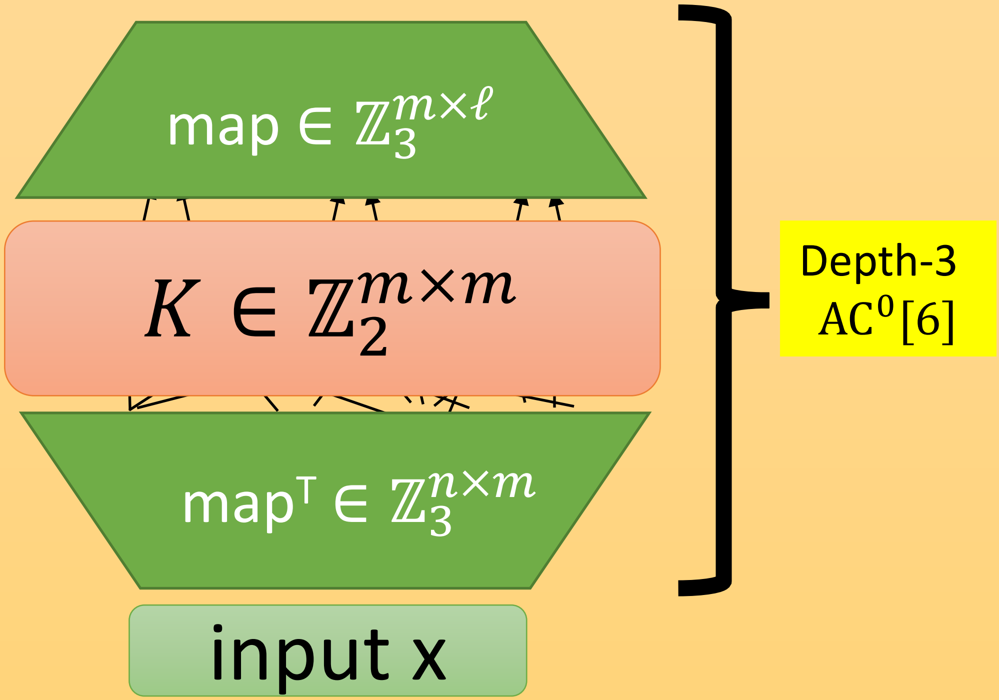
Mixing Moduli

[Boneh-I-Passelègue-Sahai-Wu 18]

Strong PRF candidate in ACC0

Conjecture:
Exponential security

==> Natural proof barrier for ACC0



map $\in \mathbb{Z}_3^{m\times\ell}$

$K \in \mathbb{Z}_2^{m\times m}$

map$^\mathsf{T} \in \mathbb{Z}_3^{n\times m}$

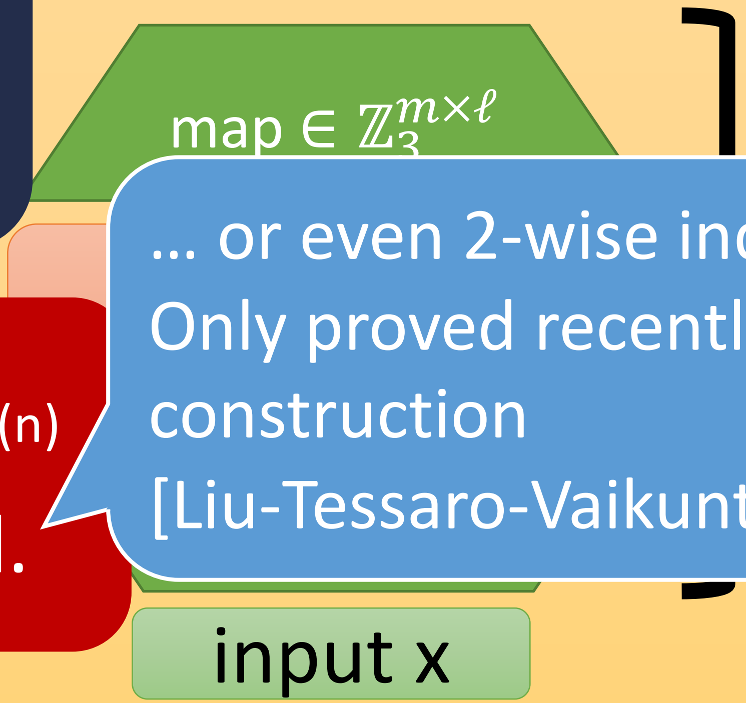
input x

Depth-3 $\mathrm{AC}^0[6]$

Mixing Moduli

[Boneh-I-Passelègue-Sahai-Wu 18]

Strong PRF candidate in ACC0



Lin-size map => asymptotically optimal PRF candidate

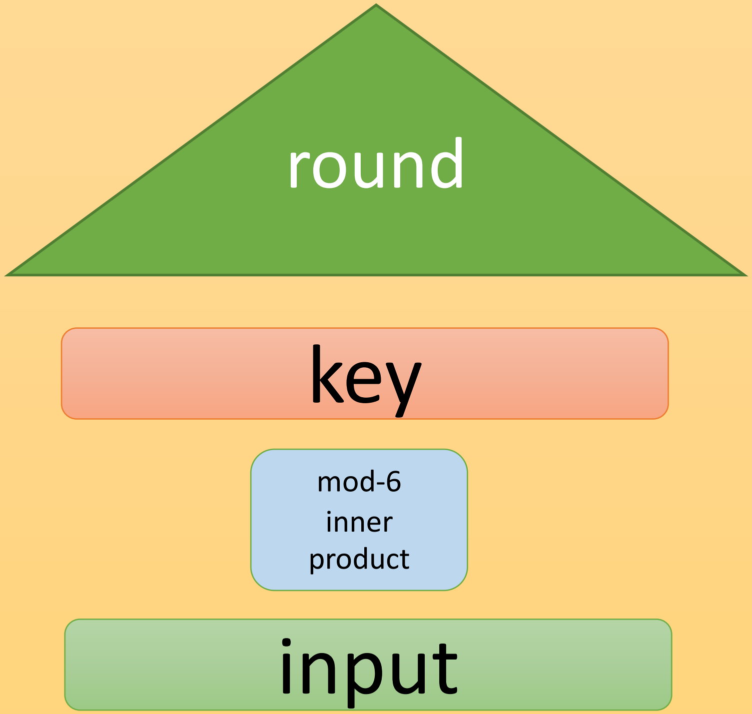
Open:
- Break in time $2^{o(n)}$
- Prove k-wise ind.

… or even 2-wise independence
Only proved recently for AES-like construction
[Liu-Tessaro-Vaikuntanathan 21]

Mixing Moduli

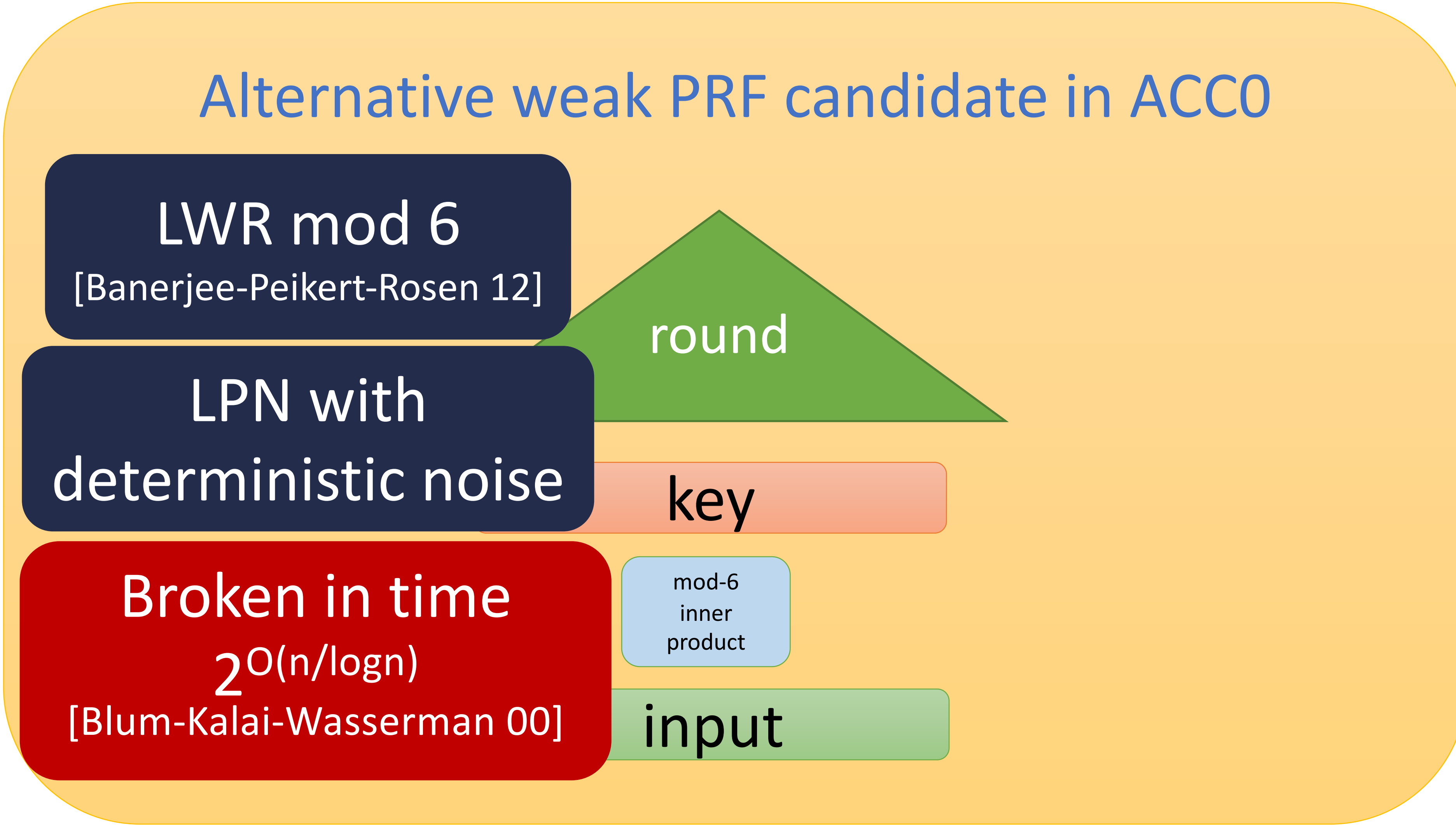
[Boneh-I-Passelègue-Sahai-Wu 18]

Alternative weak PRF candidate in ACC0



Mixing Moduli

[Boneh-I-Passelègue-Sahai-Wu 18]



Alternative weak PRF candidate in ACC0

LWR mod 6

[Banerjee-Peikert-Rosen 12]

LPN with deterministic noise

Broken in time

$2^{O(n/logn)}$

[Blum-Kalai-Wasserman 00]

round

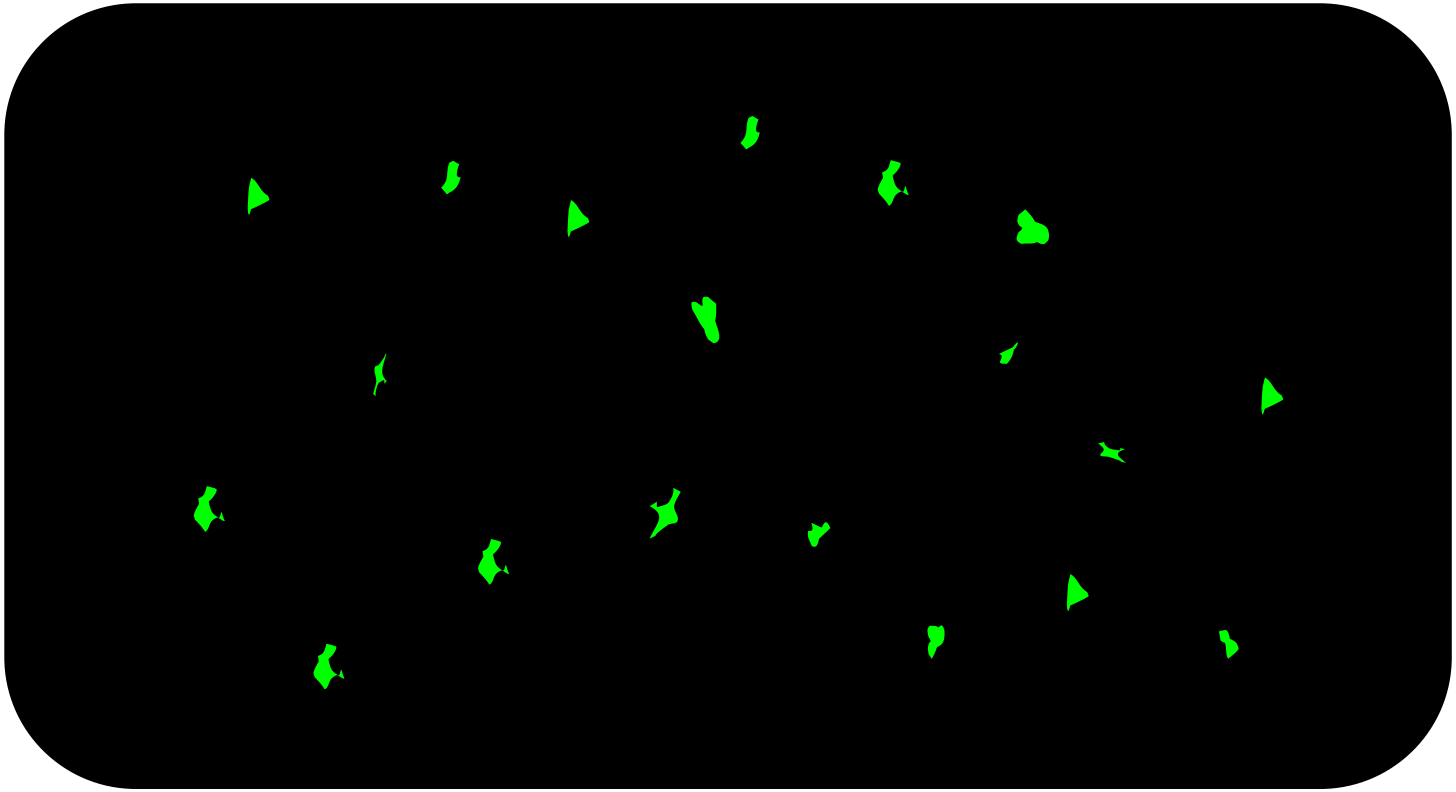
key

mod-6 inner product

input

Conclusion

- Simple hard-to-learn functions are useful!
- Many gaps in our understanding
  - Much more "dark matter" to be explored



Conclusion

- Simple hard-to-learn functions are useful!
- Many gaps in our understanding
  - Much more "dark matter" to be explored
- Introducing new assumptions can help
  - Responsibly, based on evidence, when called for
  - Critical for progress on some fronts
  - More analysis is needed
- Joint mission of several communities
  - Cryptography, cryptanalysis
  - Computational learning theory
  - Complexity theory, Algorithms, ...

Thank you!

The research leading to these results has received funding from the European Union's Horizon 2020 Research and Innovation Program under grant agreement

no. 742754 – ERC – NTSC

European
Research
Council