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

Yuval Ishai
Technion

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

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

- **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

Encryption

MAC / Signature

Secure Computation

Learn m from c

Modify m

Learn input of other party from messages
Back to the 20th Century
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.”
Introducing Pseudo-Random Functions

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

“…one may conjecture that all functions $f$ that are “simple” (i.e., that are easy to evaluate given some hidden key) can be “approximately inferred” after temporary access to an oracle for $f$. … we show that this is not the case, under the assumption that one-way functions exist.

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!

“…one may conjecture that all functions $f$ that are "simple" (i.e., that are easy to evaluate given some hidden key) can be "approximately inferred" after temporary access to an oracle for $f$. … we show that this is not the case, under the assumption that one-way functions exist.

Weak PRF:
Hard to learn under the uniform distribution
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

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

“… 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

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

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

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

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

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

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

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

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

Only weak PRF

Only quasi-polynomial hardness
Simple PRFs

Different applications motivate different notions of simplicity

Natural Proof Barriers

Simple Hard-to-Learn Functions

market for simple hard-to-learn functions

Cryptographic Applications
Simple PRFs

Different applications motivate different notions of simplicity

Natural Proof Barriers

Simple PRFs

Cryptographic Applications

Market for simple hard-to-learn functions

$x_1, k_1 \quad x_2, k_2$

$y = F_k(x)$

MPC/FHE/ZK-friendly PRF
Motivating challenge: Asymptotically Optimal PRF

\[ F_k : \{0,1\}^n \rightarrow \{0,1\}^n \]

Efficiency: \( O(n) \)-size circuit

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

Any “provable” construction?
Motivating challenge:
Asymptotically Optimal PRF

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

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

... or even heuristic?
Motivating challenge: Asymptotically Optimal PRF

\[ F_K : \{0,1\}^n \rightarrow \{0,1\}^n \]

**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

Broken by natural attacks
Heuristic constructions resisting natural attacks

Lowest complexity
Simplest
Computational Complexity of Cryptography

Default model:
boolean circuits with bounded fan-in
Minimizing Circuit Size

- $\lambda = \text{security parameter}$

<table>
<thead>
<tr>
<th></th>
<th>Insecure</th>
<th>Secure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typical:</td>
<td>$s$</td>
<td>$s^{*}\text{poly}(\lambda)$</td>
</tr>
<tr>
<td>Dream goal....</td>
<td>$s$</td>
<td>$O(s)$ i.e. $O(s)+\text{poly}(\lambda)$</td>
</tr>
</tbody>
</table>

Crypto with “constant overhead”?
Universal Hashing
[Carter-Wegman77]

- Pairwise independence:
  - $x \neq x' \implies H_k(x), H_k(x')$ are uniform and independent
Complexity of Universal Hashing

• Standard constructions
  – $H_{a,b}(x) = ax + b$, $a, b \in \text{GF}(2^n)$
  – $H_{a,b}(x) = (a \cdot x) + b$, $a \in \mathbb{Z}_2^{2n-1}$, $b \in \mathbb{Z}_2^n$
  – 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

Spielman + ABNNR
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

<table>
<thead>
<tr>
<th>Assumption</th>
<th>Primitive</th>
</tr>
</thead>
<tbody>
<tr>
<td>none</td>
<td>Universal hashing</td>
</tr>
<tr>
<td></td>
<td>One-time MAC</td>
</tr>
<tr>
<td>OWF</td>
<td>MAC</td>
</tr>
<tr>
<td></td>
<td>“Shrinking” PRF</td>
</tr>
<tr>
<td>Lin-stretch local PRG</td>
<td>PRF, PKE Signatures</td>
</tr>
<tr>
<td>Poly-stretch local PRG</td>
<td>Secure Computation with semi-honest parties</td>
</tr>
</tbody>
</table>
Constant-Overhead Cryptography

Assumption
none
OWF

[Fan-Li-Yang 21]:
Circuit size 2n (over full basis) is sufficient and necessary!

Primitive
Universal hashing
One-time MAC
MAC “Shrinking” PRF

Lin-stretch local PRG
PRF, PKE
Signatures

Poly-stretch local PRG
Secure Computation
with semi-honest parties
## Constant Overhead for Other Primitives

<table>
<thead>
<tr>
<th>Assumption</th>
<th>Primitive</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Binary-SVP</strong></td>
<td>Collision-Resistant Hashing?</td>
</tr>
<tr>
<td>[Applebaum-Haramaty-I-Kushilevitz-Vaikuntanathan17]</td>
<td></td>
</tr>
<tr>
<td><strong>Exp-secure Local OWF</strong></td>
<td>Exp-secure TDF? PRG?</td>
</tr>
<tr>
<td>[Baron-I-Ostrovsky16]</td>
<td></td>
</tr>
<tr>
<td><strong>New Candidate</strong></td>
<td>Exp-secure PRF?</td>
</tr>
<tr>
<td>[Boneh-I-Passelègue-Sahai-Wu18]</td>
<td></td>
</tr>
<tr>
<td>No candidate</td>
<td>Zero-knowledge proofs? Succinct arguments?</td>
</tr>
<tr>
<td>No candidate</td>
<td>Secure computation with malicious parties?</td>
</tr>
</tbody>
</table>
Constant Overhead for Other Primitives

Assumption

Binary-SVP

[Applebaum-Haramat-Kushilevitz-Vaikuntanathan17]

Exp-secure Local OWF

[Baron-I-Ostrovsky16]

New Candidate

[Natural proof barrier for linear-size circuits]

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

Exp-secure PRF?

Collision-Resistant Hashing?

Exp-secure TDF? PRG?

No Candidate

No Candidate

No Candidate

No Candidate

No Candidate

Later in the talk…
### Constant Overhead for Other Primitives

<table>
<thead>
<tr>
<th>Assumption</th>
<th>Primitive</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Binary-SVP</strong></td>
<td>Collision-Resistant Hashing?</td>
</tr>
<tr>
<td>[Applebaum-Haramaty-I-Kushilevitz-Vaikuntanathan17]</td>
<td></td>
</tr>
<tr>
<td><strong>Exp-secure Local OWF</strong></td>
<td>Exp-secure TDF? PRG?</td>
</tr>
<tr>
<td>[Baron-I-Ostrovsky16]</td>
<td></td>
</tr>
<tr>
<td><strong>New Candidate</strong></td>
<td>Exp-secure PRF?</td>
</tr>
<tr>
<td>[Boyle-Bourse-Nielsen-Ostrovsky15]</td>
<td></td>
</tr>
<tr>
<td>• Yes for arithmetic circuits</td>
<td>Zero-knowledge proofs? Succinct arguments?</td>
</tr>
<tr>
<td>[Bootle-Cerulli-Ghadafi-Groth-Hajiabadi-Jakobsen17]</td>
<td></td>
</tr>
<tr>
<td>[Applebaum-Damgård-I-Nielsen-Zichron17]</td>
<td></td>
</tr>
<tr>
<td>• Best overhead for Boolean: polylog((\lambda))</td>
<td>Secure computation with malicious parties?</td>
</tr>
<tr>
<td>[Damgård-I-Krøigaard10]</td>
<td></td>
</tr>
</tbody>
</table>
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
    • $\text{AC0}$: AND/OR/NOT
    • $\text{AC0}[\text{mod}_p]$: $+$ parity / $\mod_p$ for prime $p$
    • $\text{ACC0}$: $+$ $\mod_m$ for composite $m$
  – Linear-size circuits

• Assumptions
  – Standard, heuristic
Taxonomy of Constructions

- Security type
  - Weak vs. Strong

- Security level
  - Polynomial, Quasipolynomial, Subexponential, Exponential

- Complexity class
  - Constant-depth circuits with unbounded fan-in
    - \( \text{AC0} \): \text{AND/OR/NOT} circuits
    - \( \text{AC0}[\text{mod}_p] \): + parity / \text{mod}_p for prime \( p \)
    - \( \text{ACC0} \): + \text{mod}_m for composite \( m \)
  - Linear-size circuits

- Assumptions
  - Standard, heuristic
    - Viewing key \( k \) as fixed
      - No strong PRFs with better than \( q\text{poly} \) security \cite{RR94}
    - TC0: Strong PRFs under standard cryptographic assumptions \cite{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
    - $AC^0$: AND/OR/NOT
    - $AC^0[\text{mod}_p]$: + parity / mod$_p$ for prime $p$
    - $ACC^0$: + mod$_m$ for composite $m$
  
  - 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]

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

Depth-3 $AC^0[2]$

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

input
Take 2

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

\[ f_s,K(x) = \langle x, s \rangle \oplus g(K \cdot x \mod 2) \]

for \( s \in \{0, 1\}^n \), \( K \in \{0, 1\}^{(n-1) \times n} \), where \( g(x) = \bigvee_{i=1}^{\lambda} \bigwedge_{j=1}^{\lambda} \bigvee_{k=1}^{w} x_{ijk} \) is a DNF (the so-called TRIBES function). Since \( f_s,K(x) \) can be written as \( \neg \langle x, s \rangle \land g(K \cdot x) \lor \langle x, s \rangle \land \neg g(K \cdot x) \), it indeed belongs to \( \text{AC}_0 \circ \text{MOD}_2 \).

Unfortunately, this candidate was broken in [13] by algebraic attack. In our candidate, we address this issue by simply adding a layer of OR gates after the parity layer, replacing the noise function with:

\[ g(x) = \bigvee_{i=1}^{\lambda} \bigwedge_{j=1}^{\lambda} \bigvee_{k=1}^{w} x_{ijk}. \]

We conjecture that our candidate is a subexponentially secure WPRF. We observe that our candidate resists the same classes of attacks as addressed for the ABGKR candidate. However, we are further able to prove that our candidate construction has high rational degree, thus circumventing the algebraic attacks under which the ABGKR candidate was insecure.

We also study the resistance of our candidate against linear attacks, a large class of attacks that includes most state-of-the-art attacks on learning parity problems (such as the learning parity with noise assumption), whose structure bears connections to our candidate. We put forth a conjecture which, if true, implies that our candidate (as well as the WPRF candidates of [1, 14]) cannot be broken by linear attacks.

We view our results as providing a strong indication that \( \text{AC}_0 \circ \text{MOD}_2 \) may not be learnable under the uniform distribution. We compare our results to known results regarding low-complexity PRFs on Table 1. As shown in the Table, our work fills gaps in our understanding of the complexity of weak PRFs.
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 AC0 ∘ MOD2.

**CONJECTURE 1:**
There exists a WPRF in AC0 ∘ MOD2.

**CONJECTURE 2:**
There does not exist a WPRF in 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

Sparse polynomial

key

input
Depth-2 WPRF?

Candidate WPRF by XNF formulas

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

Sparse multivariate \( \mathbb{F}_2 \)-polynomials in inputs and their negation

Secure under variable-density variant of LPN

Best possible security: \( 2^{\sqrt{n}} \)

[Hellerstein-Servedio 07]

Applications:
- Correlated PRFs
- XOR-RKA security

key

\( \oplus \)

input
WPRF by XNF

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

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

Bigger j \(\Rightarrow\) more bias towards 0
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

mod-3 addition

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

input $x$

Depth-2 ACC[6]

$\oplus$ addition

Mixing Moduli

Depth-2

ACC[6]
Mixing Moduli
[Boneh-I-Passelègue-Sahai-Wu 18]

Conjecture: Exponential security

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

- Exponential hardness of learning $\mod_3 \circ \text{XOR}$ 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 $\mathbb{Z}_3$ polynomial
* Width-3 BP

Exponential hardness of learning sparse $\mathbb{Z}_3$-polynomials with uniform inputs from $\{-1,1\}^n$

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

Depth-2 ACC[6]
Mixing Moduli
[Boneh-I-Passelègue-Sahai-Wu 18]

WPRF candidate in ACC0

Conjecture: Exponential

Awesome

Annoying Complexity Class [R. Williams]

Easy to distribute!

\[ K \in \mathbb{Z}_{2}^{n \times n} \]

Input x

\[ \times \mod 3 \]
Fast Distributed Symmetric Crypto
[Dinur-Goldfeder-Halevi-I-Kelkar-Sharma-Zaverucha 21]

Candidates

Analysis

Protocols

<table>
<thead>
<tr>
<th>Construction</th>
<th>Parameters</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2, 3)-OWF</td>
<td>(s, 3.13s, s/ log 3)</td>
<td>aggressive</td>
</tr>
<tr>
<td></td>
<td>(s, 3.53s, s/ log 3)</td>
<td>conservative</td>
</tr>
<tr>
<td>(2, 3)-wPRF</td>
<td>(2s, 2s, s/ log 3)</td>
<td>aggressive</td>
</tr>
<tr>
<td></td>
<td>(2.5s, 2.5s, s/ log 3)</td>
<td>conservative</td>
</tr>
<tr>
<td>LPN-PRG</td>
<td>(s, 3s, 2s)</td>
<td></td>
</tr>
<tr>
<td>LPN-wPRF</td>
<td>(2s, 2s, s)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Primitive</th>
<th>Construction</th>
<th>Param. (n, m, t)</th>
<th>Distributed 2PC (with preprocessing)</th>
<th>Distributed 3PC</th>
<th>Public-Input 2PC (with preprocessing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>wPRF</td>
<td>(2, 3)-wPRF</td>
<td>(256, 256, 81)</td>
<td>(1536, 4, 2)</td>
<td>(2348, 662)</td>
<td>(1430, 4, 1)</td>
</tr>
<tr>
<td></td>
<td>LPN-wPRF</td>
<td>(256, 256, 128)</td>
<td>(2860, 6, 3)</td>
<td>(4995, 1730)</td>
<td>(1324, 4, 2)</td>
</tr>
<tr>
<td>OWF</td>
<td>(2, 3)-OWF</td>
<td>(128, 452, 81)</td>
<td>(904, 2, 1)</td>
<td>(2337, 717)</td>
<td>(2525, 4, 1)</td>
</tr>
<tr>
<td>PRG</td>
<td>LPN-PRG</td>
<td>(128, 512, 256)</td>
<td>(1880, 4, 2)</td>
<td>(4334, 1227)</td>
<td>-</td>
</tr>
</tbody>
</table>
 Practical post-quantum signatures

<table>
<thead>
<tr>
<th>OWF Params</th>
<th>KKW params</th>
<th>Sig. size (KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(128, 453, 81)</td>
<td>(16, 150, 51)</td>
<td>13.30</td>
</tr>
<tr>
<td>Picnic3-L1</td>
<td>(16, 168, 45)</td>
<td>12.48</td>
</tr>
<tr>
<td></td>
<td>(16, 250, 36)</td>
<td><strong>11.54</strong></td>
</tr>
<tr>
<td></td>
<td>(16, 250, 36)</td>
<td>12.60</td>
</tr>
<tr>
<td>(128, 453, 81)</td>
<td>(64, 151, 45)</td>
<td>13.59</td>
</tr>
<tr>
<td>Picnic2-L1</td>
<td>(64, 209, 34)</td>
<td>11.70</td>
</tr>
<tr>
<td></td>
<td>(64, 343, 27)</td>
<td><strong>10.66</strong></td>
</tr>
<tr>
<td></td>
<td>(64, 343, 27)</td>
<td>12.36</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OWF Params</th>
<th>KKW params</th>
<th>Sig. size (KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(256, 906, 162)</td>
<td>(16, 324, 92)</td>
<td>50.19</td>
</tr>
<tr>
<td>Picnic3-L5</td>
<td>(16, 400, 79)</td>
<td>47.08</td>
</tr>
<tr>
<td></td>
<td>(16, 604, 68)</td>
<td><strong>45.82</strong></td>
</tr>
<tr>
<td></td>
<td>(16, 604, 68)</td>
<td>48.72</td>
</tr>
<tr>
<td>(256, 906, 162)</td>
<td>(64, 322, 82)</td>
<td>51.23</td>
</tr>
<tr>
<td>Picnic2-L5</td>
<td>(64, 518, 60)</td>
<td>44.04</td>
</tr>
<tr>
<td></td>
<td>(64, 604, 57)</td>
<td><strong>43.45</strong></td>
</tr>
<tr>
<td></td>
<td>(64, 604, 58)</td>
<td>46.18</td>
</tr>
</tbody>
</table>

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

$\Rightarrow$ Natural proof barrier for ACC0

$K \in \mathbb{Z}_{2}^{m \times m}$

$\text{map} \in \mathbb{Z}_{3}^{m \times \ell}$

$\text{map}^T \in \mathbb{Z}_{3}^{n \times m}$

input x

Depth-3 $AC^0[6]$
Mixing Moduli
[Boneh-I-Passelègue-Sahai-Wu 18]

Strong PRF candidate in ACC0

Lin-size map $\Rightarrow$ asymptotically optimal PRF candidate

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

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

... 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

round

key

mod-6 inner product

input
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
  $O(\frac{n}{\log n})$
  [Blum-Kalai-Wasserman 00]
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, ...
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