Pseudorandom Functions, Linear Codes, and Substitution-Permutation Networks

Youlong Ding

The Hebrew University of Jerusalem

Based on joint work with Aayush Jain, Ilan Komargodski

[DJK'25] A New Approach for LPN-based Pseudorandom Functions: Low-Depth and Key-Homomorphic

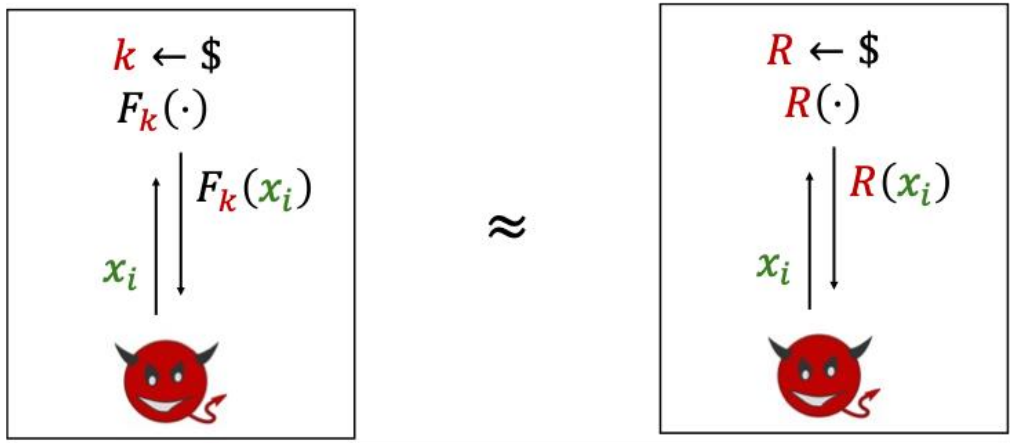
[DJK'26] Logarithmic-Depth Pseudorandom Functions from Well-Founded Code-Based Assumptions

Pseudorandom Functions (PRFs) [GGM86]

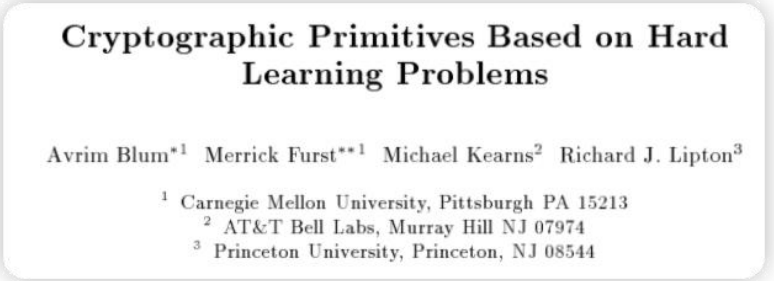
$$F\colon \mathcal{K} \times \{0,1\}^n \to \mathcal{Y}$$

- Efficiently computable deterministic function

- Indistinguishable from a random function



Learning Parity with Noise


Cryptographic Primitives Based on Hard Learning Problems

Avrim Blum*[1] Merrick Furst**[1] Michael Kearns[2] Richard J. Lipton[3]

[1] Carnegie Mellon University, Pittsburgh PA 15213
[2] AT&T Bell Labs, Murray Hill NJ 07974
[3] Princeton University, Princeton, NJ 08544

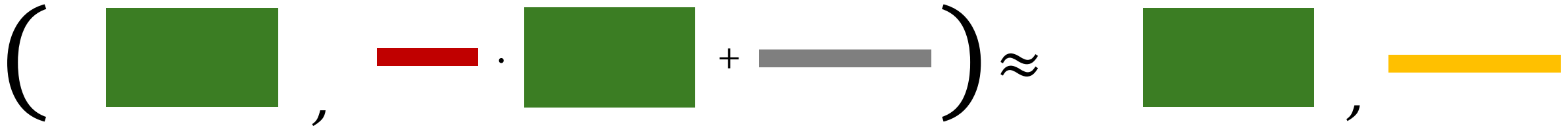
[BFKL93]

$$(A, sA + e) \approx (A, u)$$

- LPN: samples of the form $\langle a_i, s \rangle + e_i$ is random even given a_i

$$A \leftarrow \mathbb{Z}_2^{n \times m} \qquad s \leftarrow \mathbb{Z}_2^{n} \qquad e \leftarrow Ber_\mu^m$$

$$Ber_\mu = \begin{cases} 1 & \text{w.p. } \mu \\ 0 & \text{w.p. } 1-\mu \end{cases}$$



- Parameter: #sample $m(n)$, noise level $\mu(n) > 1/n$

Why NC1 PRF from LPN?

NC1 PRF

- Optimality
- Applications:
  - high-end cryptographic primitives [ARS+15,MJSC16,BIP+18,BCG+20]
  - MPC [DI06,IKOS08,GVW12,BJKL21
  - Circuit lower bounds [RR94,MV12]
  - Derandomization [NW88,Wil13]
  - Learning complexity [Val84,KV94]

LPN

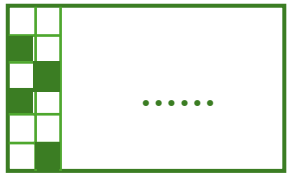
- Plausibly post-quantum secure
- Active area of research
- Equivalent to the search version
- “decoding random linear codes”

NC1 PRF meets LPN

- Curiosity: DDH [NR97], factoring [NRR02], k-Lin [LW09], LWE [BPR12]
- ?

Our Results

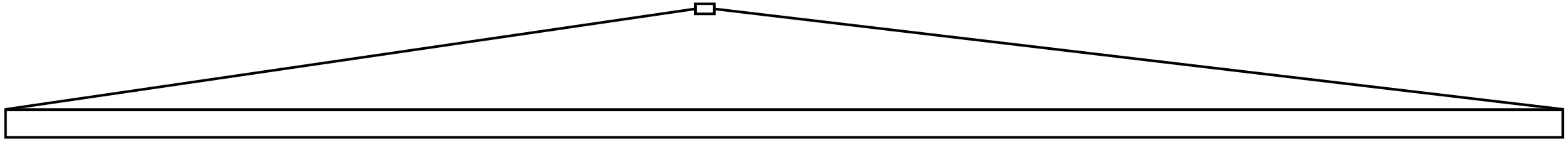
Result 0: Weak PRFs in NC^1 from LPN [DJK'25]
Result 1: PRFs in NC^1 from Sparse-LPN* [DJK'26]
Result 2: PRFs in NC^1 from Ring-LPN* [DJK'26]
Result 3: PRFs in NC^1 from quasi-poly LPN [DJK'26]



- LPN noise level: $\mu = n^{-o(1)}$ noise $\left(n^{-0.1} < \mu < 0.1\right)$
- Our parameter regime is not known to imply PKE
- First key-homomorphic PRFs in NC1: $F_{k_1+k_2}(x) \approx F_{k_1}(x) + F_{k_2}(x)$

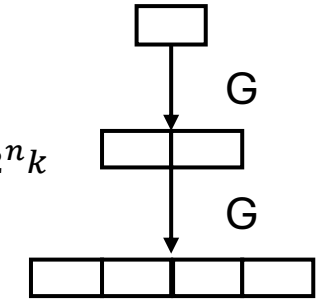
Revisiting [GGM86]

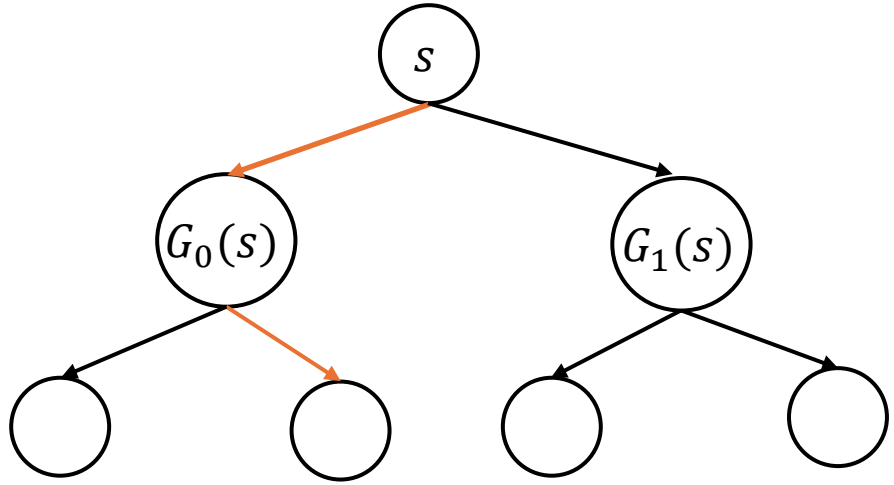
PRF is a PRG with exponential stretch



Repeat PRG $G: \{0,1\}^k \to \{0,1\}^{2k}$

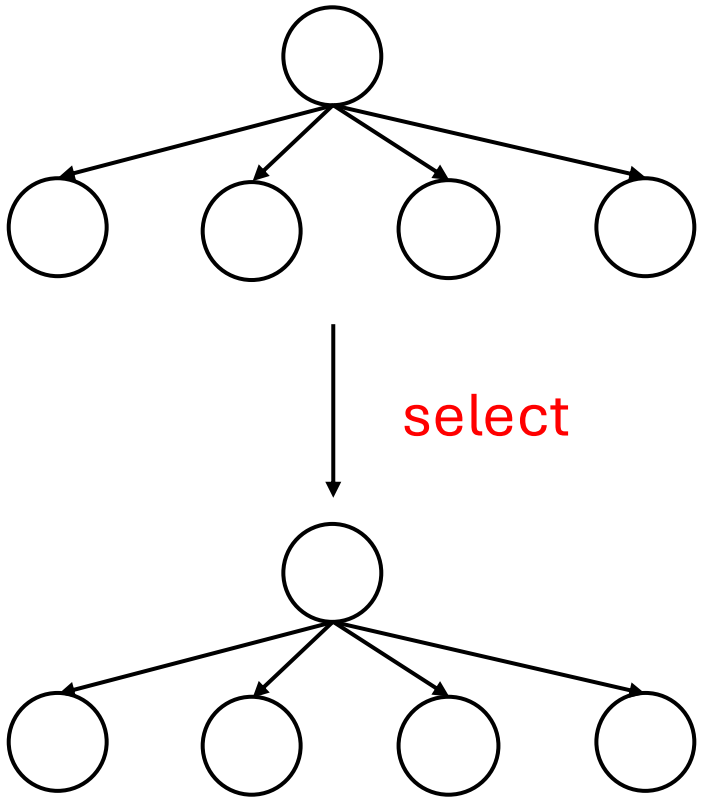
$G^n: \{0,1\}^k \to \{0,1\}^{2^n k}$





$F_s(01)$

NC1 PRF from PRG via GGM?



Poly-stretch local PRG $G: \{0,1\}^n \to \{0,1\}^{n^2}$
On $x = (x_1, \ldots, x_\ell), x_i \in [n]$:

$$F_s(x) = G_{x_\ell} \circ \cdots \circ G_{x_1}(s)$$

- $\ell = \omega(1)$ layers suffice (for super-poly domain size)
- If G is in NC^0, i.e., constant depth
- Then the depth of PRF is $O(\ell)$?
- $O(\ell + \ell \cdot \log n) = \omega(\log n)$, not in NC^1

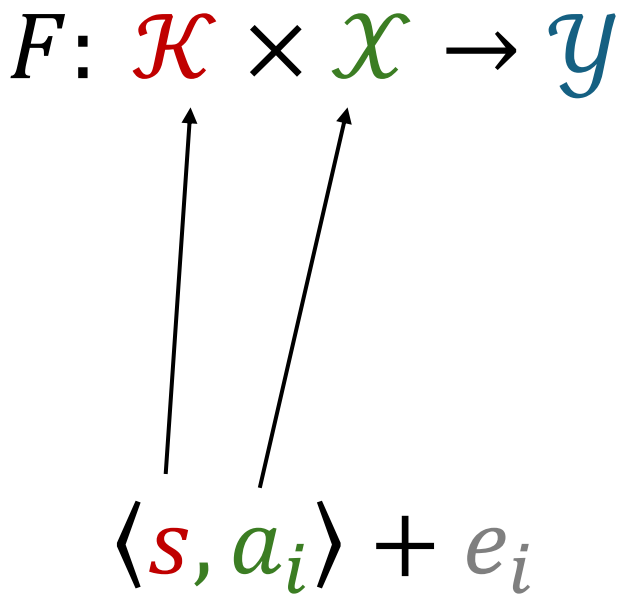
NC1 PRF from PRG via ~~GGM?~~ Specific Assumptions

- PRFs in NC^1 from:
  - DDH [NR97]
  - factoring [NRR02]
  - k-Linear [LW09]
  - Ring-LWE [BPR12]
  - (Variants of) LPN?

Weak PRFs from LPN [DJK25]

Weak PRF:

$$\left(r_i, F_k(r_i)\right) \approx (r_i, u_i)$$

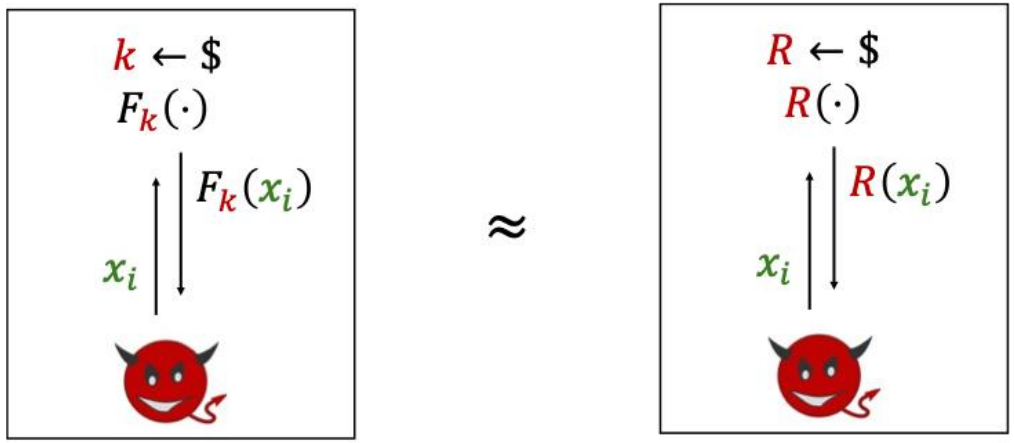


Pseudorandom Functions (PRFs) [GGM86]

$$F\colon \mathcal{K} \times \{0,1\}^n \to \mathcal{Y}$$

- Efficiently computable deterministic function
  - Why deterministic?

- Indistinguishable from a random function



Weak PRFs from LPN: Trivial Derandomization

- $\widetilde{F}_s(a_i) = \langle s, a_i \rangle + e_i$
- $F_s(a_i) = \langle s, a_i \rangle$

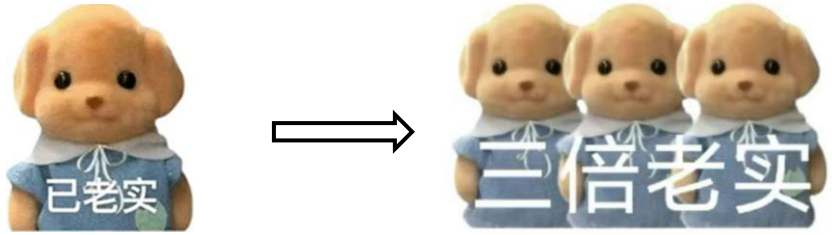
$\widetilde{F}_s(a_i) = F_s(a_i)$ except with prob. $\mu \geq 1/poly$

Weak PRFs from LPN: Repeat?

$$F_{s_1}(a_i) = \langle s_1, a_i \rangle$$
$$F_{s_2}(a_i) = \langle s_2, a_i \rangle$$
$$F_{s_3}(a_i) = \langle s_3, a_i \rangle$$

Extractor

- $\widetilde{F}_s(a_i) = F_s(a_i)$ except $1/poly$
- Apply amplification lemma?



- $\{\widetilde{F}_s(a_i)\}_i \neq \{F_s(a_i)\}_i$ w.h.p.

Weak PRFs from LPN [DJK25]

- We would like $\widetilde{F}_s(a_i) = F_s(a_i)$ except negl
- By a union bound, $\{\widetilde{F}_s(a_i)\}_i = \{F_s(a_i)\}_i$ except negl

Weak PRFs from LPN [DJK25]

- $F_s(a_i; r) = \langle s, a_i \rangle + e_i$ deterministic

Recursive Derand.

$F_{s_1}(a_i; r_1)$

$F_{s_2}(a_i; r_2)$

$F_{s_3}(a_i; 0)$

Weak PRFs from LPN [DJK25]

Bernoulli Sampler

S-Ber: $\{0,1\}^{\ell} \to \{0,1\}$

$F_{S=(s_1,\dots)}(a_i)$:

$F_{s_1}(a_i; r_1)$

$F_{\dots}^{\ell}(a_i; r_2)$

$F_{\dots}^{\ell^2}(a_i; 0)$

- It works, but why?

Weak PRFs from LPN: Wolf & Dog

$\widetilde{F}_s(a_i)$: $=$ $F_S(a_i)$:

$F_{s_1}(a_i; \tilde{e}_1)$ $=$ $F_{s_1}(a_i; e_1)$

w.p. 1

$-$ $=$ $-$

w.p. 1

S−Ber $\circ$ $F^{\ell}_{s_2}(a_i; \tilde{e}_2)$ S−Ber $\circ$ $F^{\ell}_{s_2}(a_i; e_2)$

$-$ $=$ $-$

w.p. $1-\mu'$

S−Ber $\circ$ $F^{\ell^2}_{s_3}(a_i; \tilde{e}_3)$ S−Ber $\circ$ $F^{\ell^2}_{s_3}(a_i; \mathbf{0})$

freshly sampled

$\neq \mathbf{0}$

w.p. $\mu' = nonnegl$

- $\widetilde{F}_s(a_i)$ is pseudorandom under LPN
- $\widetilde{F}_s(a_i) \neq F_s(a_i)$ with negl. prob.
  - Iff " $\neq$ " happens for all layers, i.e., $\mu'^{\ell} = negl(n)$

Weak PRFs from LPN: Unrolling Recursion

- Parameter: $\mu = 2^{-\ell}$, $\ell = \omega(1)$, recursion depth $\tau = \omega(1)$
- Secret key: s, $(s_1^{(1)}, \dots, s_\ell^{(1)}), \dots, (s_1^{(\tau)}, \dots, s_{\ell^\tau}^{(\tau)}) \in \mathbb{Z}_2^n$
- On input $a \in \mathbb{Z}_2^n$:
  - For $\alpha = \tau, \tau - 1, \dots, 1$:
    - If $\alpha = \tau$: $r^{(\tau)} \leftarrow \left(\left\langle a, s_1^{(\tau)} \right\rangle, \dots, \left\langle a, s_{\ell^\tau}^{(\tau)} \right\rangle\right) \in \mathbb{Z}_2^{\ell^\tau}$
    - If $\alpha < \tau$: $r^{(\alpha)} \leftarrow \left(\left\langle a, s_1^{(\alpha)} \right\rangle + e_1^{(\alpha)}, \dots, \left\langle a, s_{\ell^\alpha}^{(\alpha)} \right\rangle + e_{\ell^\alpha}^{(\alpha)}\right) \in \mathbb{Z}_2^{\ell^\alpha}$
    - $e^{(\alpha-1)} \leftarrow \text{S-Ber}\left(r^{(\alpha)}\right) \in \mathbb{Z}_2^{\ell^{\alpha-1}}$ (Applying S-Ber on each length-ℓ block)
  - Output $\langle a, s \rangle + e^{(0)}$

all inner products of $\langle a, s \rangle$ can be done in parallel

Evaluation Depth: $O(\log n + \tau \log \ell) = O(\log n)$

Moving to PRFs

- $G(s;r) = sA + e, A \in \{0,1\}^{n \times n^2}$
- a poly-stretch "PRG" $G: \{0,1\}^n \to \{0,1\}^{n^2}$
- $G(s;r)_i = sA_i + e_i$
- Via GGM:

$$F_s(x; r_x = (r_{x_1}, r_{x_{12}})) = (sA_{x_1} + e_{x_1})A_{x_2} + e_{x_{12}}$$

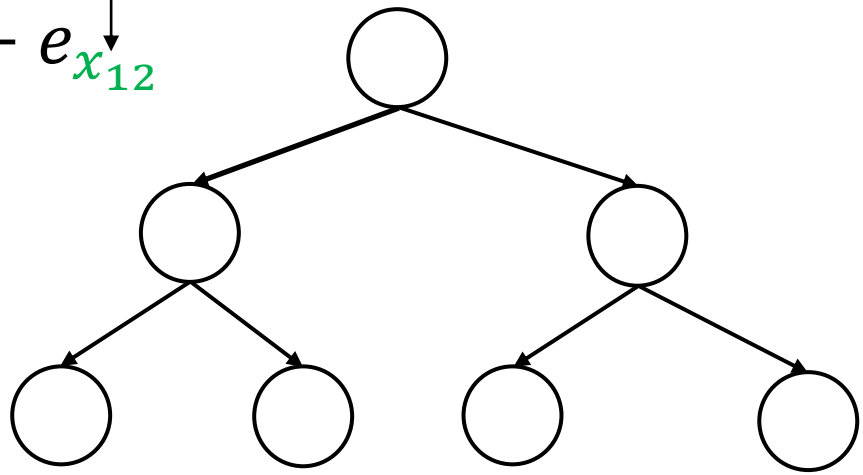
- Recursive Derandomization:

$$F_s(x; r_{x_1}, r_{x_{12}})$$

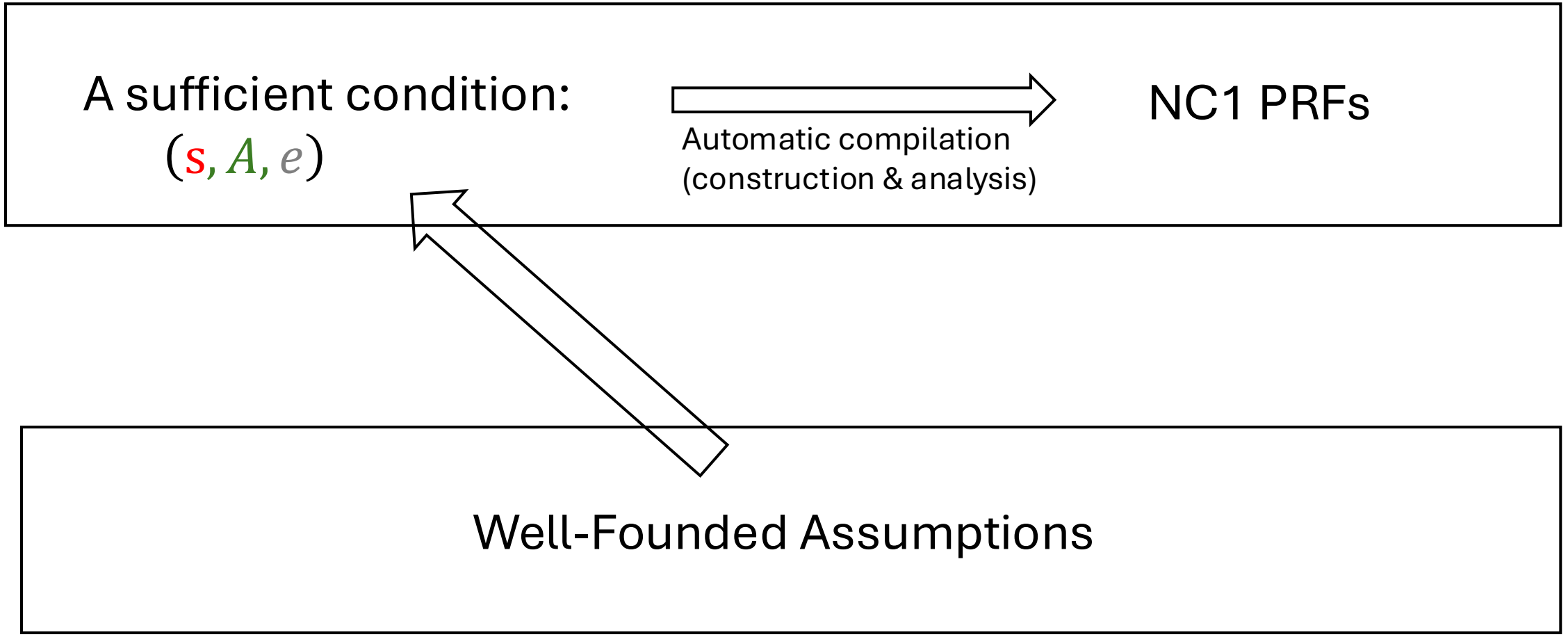
$$F_{\dots}(x_1; r_{x_1}) \qquad F_{\dots}(x_{12}; r_{x_1}, r_{x_{12}})$$

$$F_{\dots}(x_1; 0) \qquad F_{\dots}(x_1; 0) \qquad F_{\dots}(x_{12}; 0, 0)$$

Important:
- Avoid "Re-randomized" LPN
- Avoid "Correlated" LPN

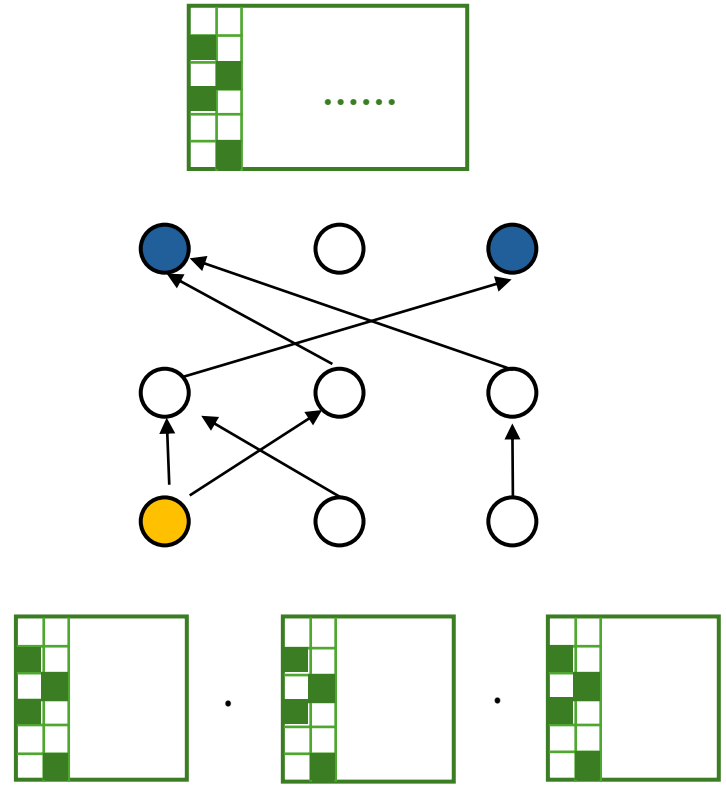


A Unified Framework [DJK26]



A sufficient condition [DJK26]

1. $(A, sA + e) \approx (A, u), A = (A_1, A_2, \dots, A_n) \in \mathbb{F}^{n \times n^2}$
2. Each column of A has sparsity $\lambda^{o(1)}$
3. The reachability set of $\omega(1)$-step walk on the layered graph induced by $(A_{x_1}, \dots, A_{x_k})$ can be computed in $O(\log \lambda)$ depth.
4. The $\omega(1)$-product $\prod A_{x_i}$ can be computed in $O(\log \lambda)$ depth

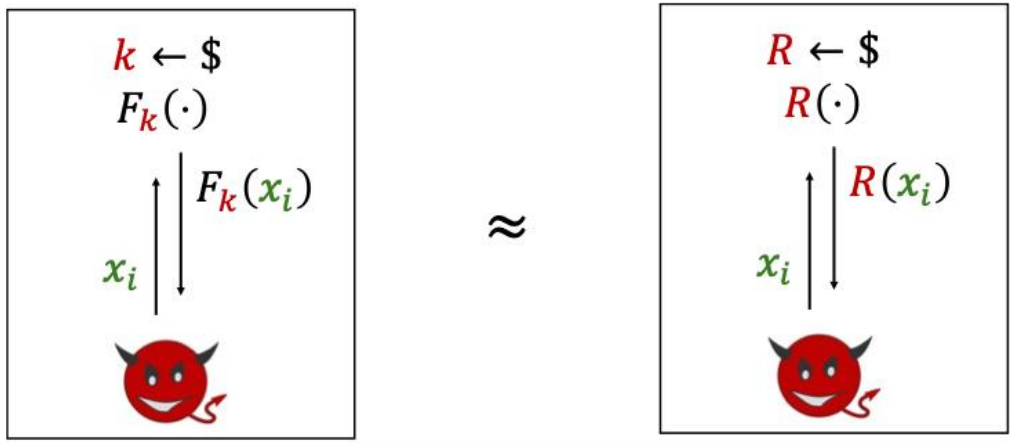


Pseudorandom Functions (PRFs) [GGM86]

$$F\colon \mathcal{K} \times \{0,1\}^n \to \mathcal{Y}$$

- Efficiently computable deterministic function
  - Why deterministic?
  - Mathematically defined *function* ⇐ An evaluation *algorithm*

- Indistinguishable from a random function



Pseudorandom Functions (PRFs)

Randomized function $\tilde{F}$

Deterministic function
$F: \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$

approximate

Randomized algorithm $A(k, x)$

induce

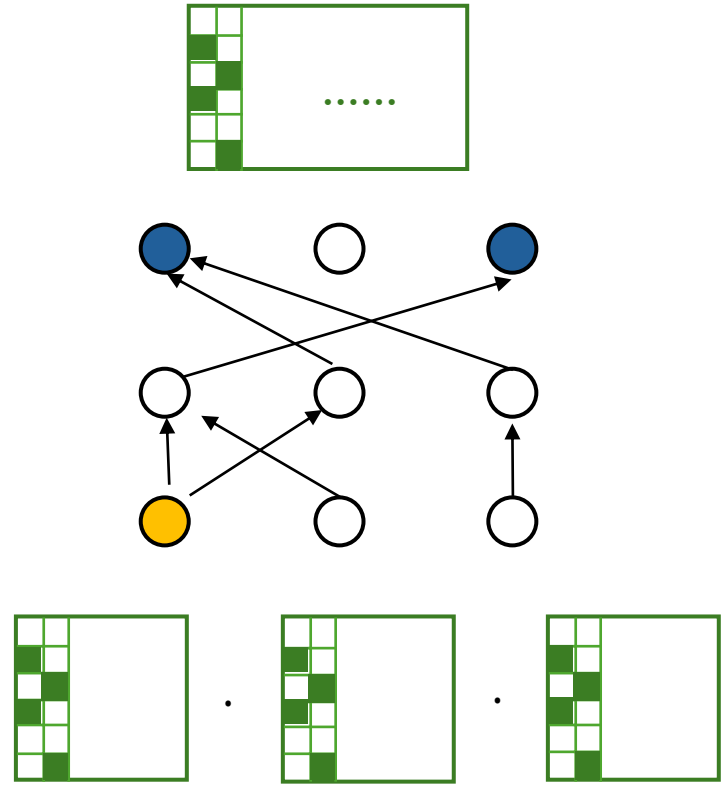
Deterministic function
$F': \mathcal{K} \times \{0,1\}^* \times \mathcal{X} \to \mathcal{Y}$

$\approx$(using assumption)

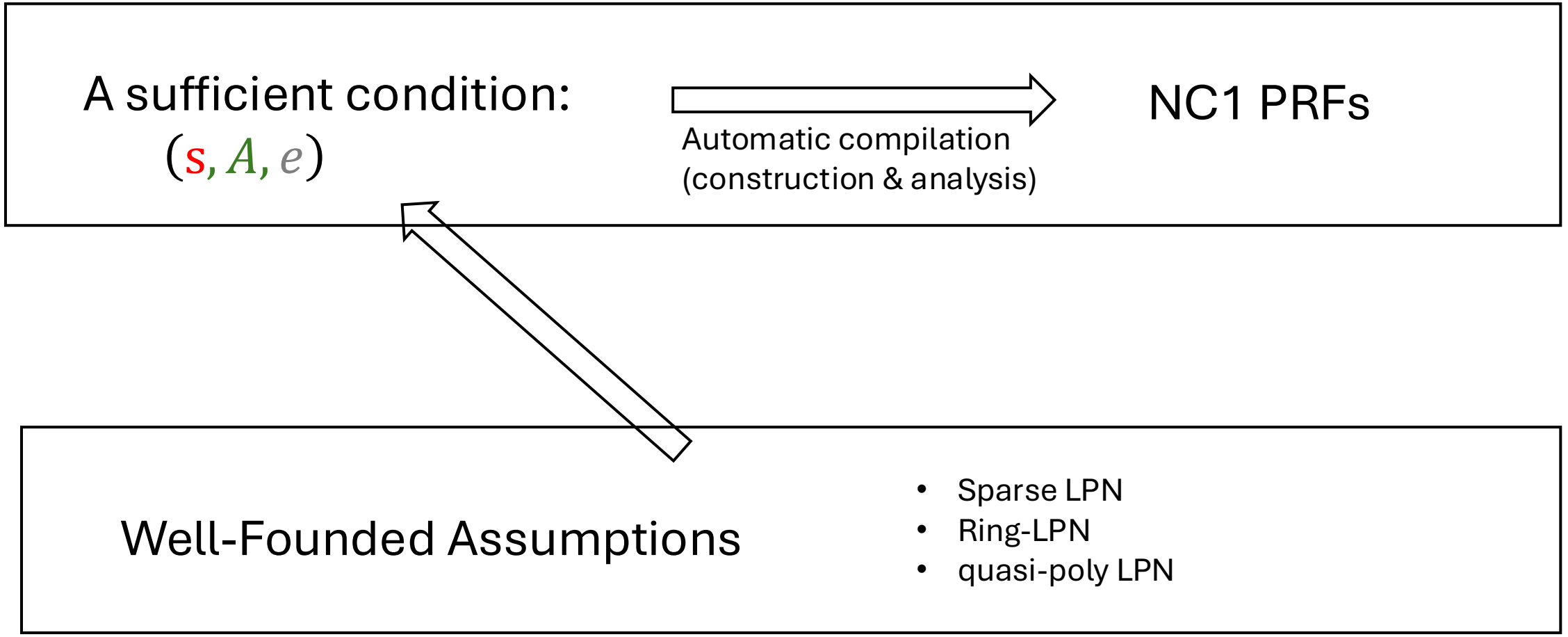
Truly random function R

A sufficient condition [DJK26]

1. $(A, sA + e) \approx (A, u), A = (A_1, A_2, \dots, A_n) \in \mathbb{F}^{n \times n^2}$
2. Each column of A has sparsity $\lambda^{o(1)}$
3. The reachability set of $\omega(1)$-step walk on the layered graph induced by $(A_{x_1}, \dots, A_{x_k})$ can be approximated in $O(\log \lambda)$ depth.
4. The $\omega(1)$-product $\prod A_{x_i}$ can be approximated in $O(\log \lambda)$ depth



A Unified Framework [DJK26]

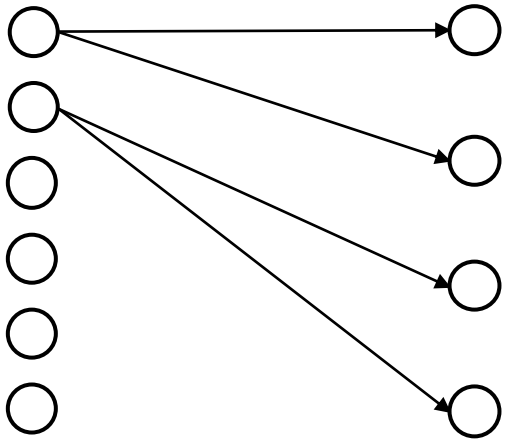


NC1 PRF from Sparse LPN w.r.t. a *sublinear*-depth expander

$$sA + e, A = (A_1, A_2, \dots, A_n) \in \{0,1\}^{n \times n^2}$$

Expander:

$$\phi_A : [n^2] \to [n]^d$$



$$\forall S \subseteq [n], s.t. |S| \leq N, \text{ it holds that}$$

$$|\phi(S)| > \frac{1}{2} \cdot d|S|$$

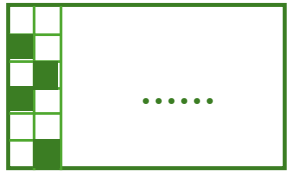
This will imply that

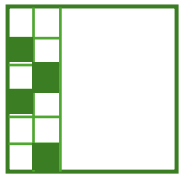
any N columns of A is linearly independent

- Random expander: Requires *linear* depth to evaluate ϕ, i.e., $O(\log n)$

NC1 PRF from Sparse LPN w.r.t. a *sublinear*-depth expander

$$sA + e, A = (A_1, A_2, \dots, A_n) \in \{0,1\}^{n \times n^2}$$





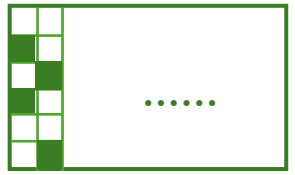
- Given an sublinear-depth expander $\phi_A : [n^2] \to [n]^d$
  - Depth of ϕ_A is $o(\log n)$
  - [TUZ01],[GUV09],[KT22]
- $\phi_{A_i} : [n] \to [n]^d$
- We can traversal the layered graph
  - $\phi_{A_k} \circ \cdots \circ \phi_{A_{x_2}} \circ \phi_{A_{x_1}}(i)$
  - $\phi_{A_i}(j) \coloneqq \phi_A(j + i \cdot n)$
  - This gives the reachability set of a k-step walk (*with multiplicity*)

- To compute the reachability set: traversal + deduplicate
- To compute multiplication: traversal + counting (mod 2)

More on Sparse LPN w.r.t. a *sublinear*-depth expander

$$sA + e, A = (A_1, A_2, \dots, A_n) \in \{0,1\}^{n \times n^2}$$

$$\phi_A : [n^2] \to [n]^d$$



Remark 1 It can be shown that if the matrix A_n happens to be "degenerate" (for example, contains two equal rows, which occurs with probability $1/n$) then one can distinguish vectors b_1 and b_2 with probability roughly $1/n$. This gives an algorithm that distinguishes (A, b_1) and (A, b_2) with success $1/n^2$. We believe that no algorithm can do substantially better than this bound, and if A_n is an expander (which occurs with probability $1 - O(1/n)$) then the distributions of b_1 and b_2 are indistinguishable. Thus, we could specify in Problem 2 that A is chosen uniformly from the set of good expanders and assume its $1/n^{\Omega(1)}$-intractability (but this would sacrifice the property of being samplable).

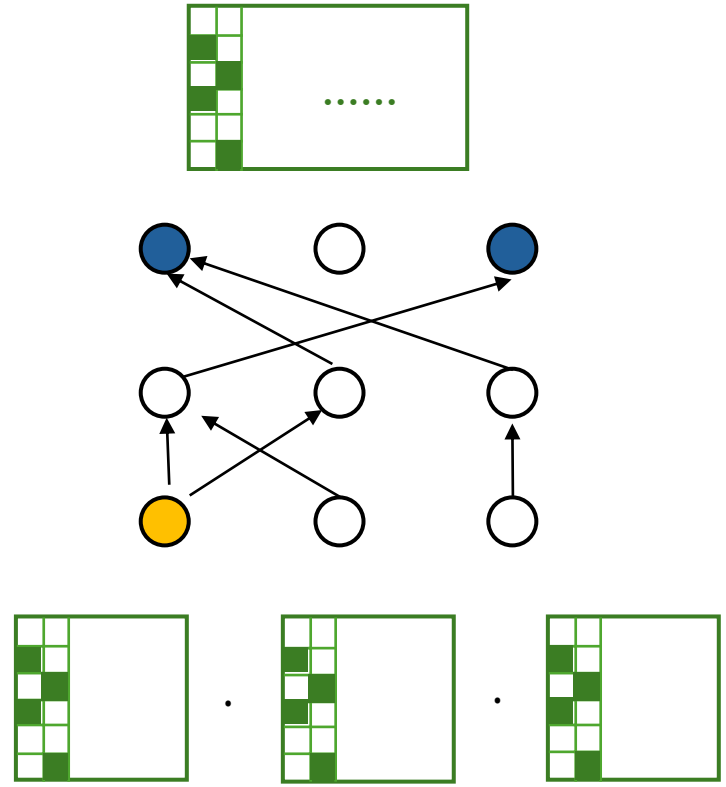
- [Alekhnovich03]: Conjectured that SparseLPN holds w.r.t. any expander (⋆)
- For our results, we only need to assume that
  - There exists a *sublinear-depth* expander for each sparse LPN holds. (⋆⋆)
  - *sublinear-depth v.s. linear depth*.

[MST'19][BLRS'26]

- The attack works for specific choice of the expander.
- "Cautionary counter evidence" that (⋆) might be false
- Compared to: DDH in any group (unconditionally false)

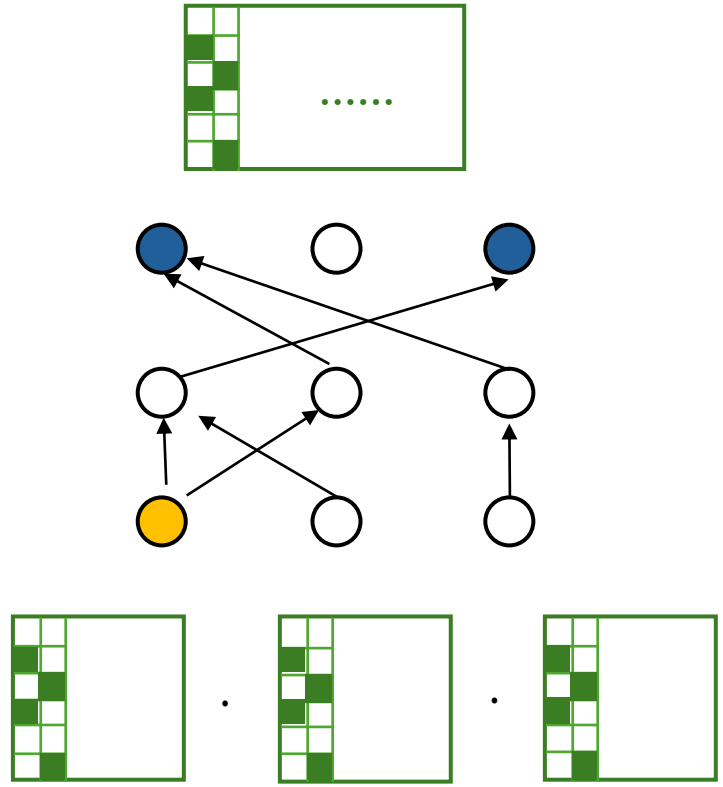
NC1 PRF from Ring-LPN

1. $(A, sA + e) \approx (A, u), A = (A_1, A_2, \ldots, A_n) \in \mathbb{F}^{n \times n^2}$
2. Each column of A has sparsity $\lambda^{o(1)}$
3. The reachability set of $\omega(1)$-step walk on the layered graph induced by $(A_{x_1}, \ldots, A_{x_k})$ can be approximated in $O(\log \lambda)$ depth.
4. The $\omega(1)$-product $\prod A_{x_i}$ can be approximated in $O(\log \lambda)$ depth



NC1 PRF from quasi-poly LPN

1. $(A, sA + e) \approx (A, u), A = (A_1, A_2, \dots, A_n) \in \mathbb{F}^{n \times n^2}$
2. Each column of A has sparsity $\lambda^{o(1)}$
3. The reachability set of $\omega(1)$-step walk on the layered graph induced by $(A_{x_1}, \dots, A_{x_k})$ can be approximated in $O(\log \lambda)$ depth.
4. The $\omega(1)$-product $\prod A_{x_i}$ can be approximated in $O(\log \lambda)$ depth



Why NC1 PRF from LPN?

NC1 PRF

- Enable high-end cryptographic primitives [ARS+15,MJSC16,BIP+18,BCG+20]
- Central measure of complexity
  - [DI06,IKOS08,GVW12,BJKL21]
- Circuit lower bounds [RR94,MV12]
- Derandomization [NW88,Wil13]
- Learning complexity [Val84,KV94]

LPN

- Plausibly post-quantum secure
- not known to imply PKE/CRH
- Equivalent to the search version
- “decoding random linear codes”

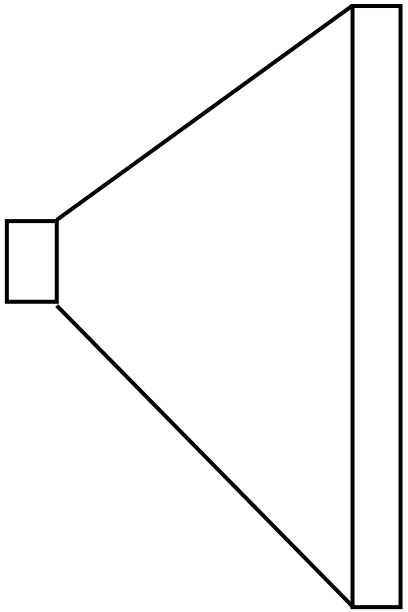
NC1 PRF meets LPN

- Curiosity
- First Key-Homomorphic NC1 PRF
- ?

PRFs in the wild

Theoretical constructions:

PRG with exponential stretch

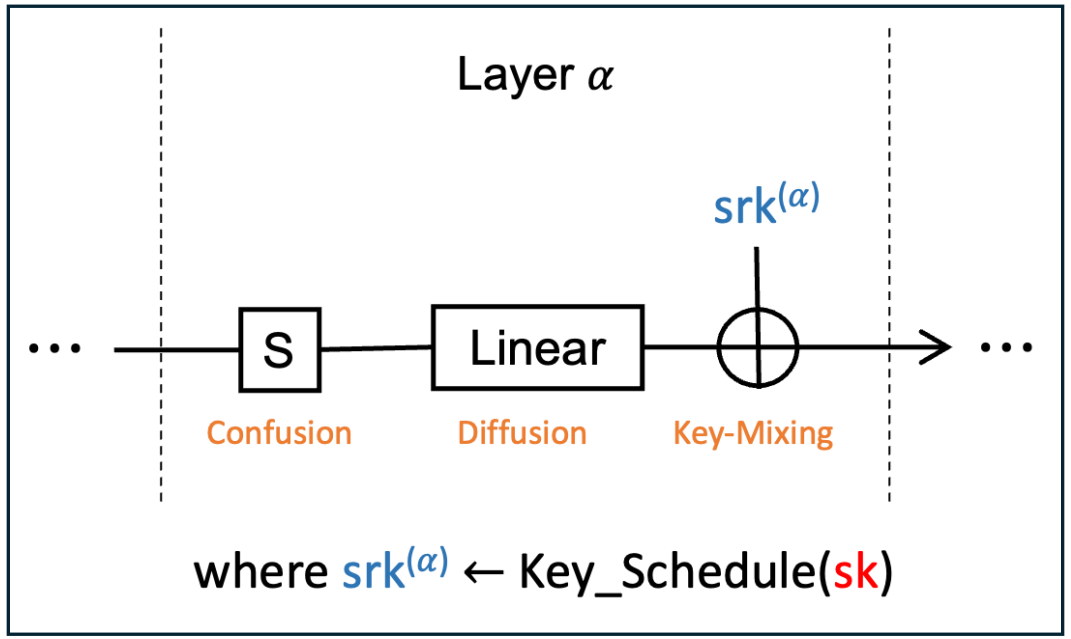


Practical constructions:

“Complicated” Transformation

$$f(x) = x^{\pi} + sinx + \frac{1}{x} + \cdots$$

Substitution-Permutation Networks [Shannon49]



- Key Schedule: $\text{srk}^{(\alpha)} \leftarrow \langle s, A_x \rangle$
- S := S-Ber $= \prod x_i$
- Linear: A'_x

S-Box

$x \mapsto x^{-1}$ (AES)
$x \mapsto x^3$ (MiMC)
…

Sample Bernoulli noise
$x \mapsto \prod x_i$

Open Problem:

- Subexponential security?
- Beyond Ring-LPN over cyclotomic rings Z_q[X]/(X^n +1)?

- PRFs in NC1 from LPN?
- More explicit constructions of sublinear-depth expanders?