Pseudorandom functions in almost constant depth from low-noise LPN

Yu Yu

上海交通大學 Shanghai Jiao Tong University

Westone Cryptologic Research Center

joint work with 许荣 (John Steinberger)

清華大學 Tsinghua University

Outline

- Introduction to LPN
  - Decisional and Computational LPN
  - Asymptotic hardness of LPN
  - Related work
  - (randomized) PRFs and PRGs
- The road map
  - Overview of the LPN-based randomized PRG in $\mathrm{AC}_0(\mathrm{mod}\ 2)$
  - Bernoulli noise extractor in $\mathrm{AC}_0(\mathrm{mod}\ 2)$
  - Bernoulli-like noise sampler in AC_0
  - randomized PRG → randomized PRF
- Conclusion and open problems

Learning Parity with Noise (LPN)

Challenger: $a \stackrel{\$}{\leftarrow} \{0,1\}^{q\times n}, x \stackrel{\$}{\leftarrow} \{0,1\}^n, e \leftarrow \mathrm{Ber}_\mu^q, y := Ax + e$

Ber_μ :Bernoulli distribution of noise rate $0 < \mu < \frac{1}{2}$

Pr $[\mathrm{Ber}_\mu = 1] = \mu$

Pr $[\mathrm{Ber}_\mu = 0] = 1 - \mu$

Ber_μ^q: q-fold of Ber_μ

$$\overset{\mathbf{a}}{\begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}} \cdot \overset{\mathbf{x}}{\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}} + \overset{\mathbf{e}}{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}} = \overset{\mathbf{y}}{\begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}} \pmod 2$$

Search LPN: given a and y, find out s

Decisional LPN: distinguish (a, y) from (a, U_q)

[Blum et al.94, Katz&Shin06]: the two versions are (poly) equivalent

In fact: can use $x \leftarrow \mathrm{Ber}_\mu^n$ instead of $x \stackrel{\$}{\leftarrow} \{0,1\}^n$

Hardness of LPN

- worst-case hardness

  LPN (decoding random linear code) is NP-hard.

- average-case hardness
  - constant noise $\mu = O(1)$

    BKW (Blum,Kalai,Wasserman): $t = q = 2^{O(n/\log n)}$

    Lyubashevsky's tradeoff: $t = 2^{O(n/\log\log n)}$, $q = n^{1+\epsilon}$
  - low noise $\mu = n^{-c}$ (for constant $0 < c < 1$)

    $t = 2^{O(n^{1-c})}, q = n + O(1)$

- quantum resistance

Related Work

- public-key cryptography from LPN
  - CPA PKE from low-noise LPN [Alekhnovich 03]
  - CCA PKE from low-noise LPN [Dottling et al.12, Kiltz et al. 14]
  - CCA PKE from constant-noise LPN [Yu & J. Zhang C16]
- symmetric cryptography from LPN
  - Pseudorandom generators [Blum et al.93, Applebaum et al.09]
  - Authentication schemes [Hopper&Blum 01, Juels et al. 05,…]
    [Kiltz et al.11, Dodis et al.12, Lyu & Masny13, Cash et al.16]
  - Perfectly binding string commitment scheme [Jain et al. 12]
  - Pseudorandom functions from (low-noise) LPN?

This work

Main results

- Low-noise LPN implies
  - ■Polynomial-stretch pseudorandom generators (PRGs) in $\mathrm{AC}_0(\mathrm{mod}\ 2)$

    $\mathrm{AC}_0(\mathrm{mod}\ 2)$: polynomial-size, $O(1)$ -depth circuits with unbounded fan-in $\wedge, \vee, \oplus$.
  - ■Pseudorandom functions (PRFs) in $\widetilde{\mathrm{AC}}_0(\mathrm{mod}\ 2)$

    $\widetilde{\mathrm{AC}}_0(\mathrm{mod}\ 2)$: polynomial-size, $\omega(1)$ -depth circuits with unbounded fan-in $\wedge, \vee, \oplus$

[Razborov & Rudich 94]: good PRFs do NOT exist in $\mathrm{AC}_0(\mathrm{mod}\ 2)$

- More about the PRGs/PRFs:
  - ■weak seed/key of sublinear entropy & security ≈ LPN on linear size secret
  - ■uniform seed/key of size λ & security up to $2^{O(\lambda/\log\lambda)}$
- Technical tools:
  - ■Bernoulli noise extractor in $\mathrm{AC}_0(\mathrm{mod}\ 2)$

    Rényi entropy source → Bernoulli distribution
  - ■Bernoulli-like noise sampler in AC_0

    Uniform randomness → Bernoulli-like distribution
  - ■ Security-preserving and depth-preserving domain extender for PRFs

(randomized) PRGs, PRFs and LPN

- $G_a: \{0,1\}^n \times \{0,1\}^m \rightarrow \{0,1\}^l$ $(n < l)$ is **randomized PRG** if

$$(G_a(U_n), a) \ \sim_c \ (U_l, a)$$

- $F_{k,a}: \{0,1\}^n \times \{0,1\}^n \times \{0,1\}^m \rightarrow \{0,1\}^l$ is **randomized PRF** if for every PPT A

$$|\Pr[A^{F_{k,a}}(a) = 1] \ - \Pr[A^R(a) = 1]| = negl(n)$$

where $R: \{0,1\}^m \rightarrow \{0,1\}^l$ is a random function.

- Can we obtain (randomized) PRGs and weak PRFs from LPN ?

try eliminating the noise (like LWR from LWE)

$$\underbrace{\langle a_1, x\rangle, \cdots, \langle a_i, x\rangle}_{L(\cdot)}, \cdots, \underbrace{\langle a_{q-i+1}, x\rangle, \cdots, \langle a_q, x\rangle}_{L(\cdot)}$$

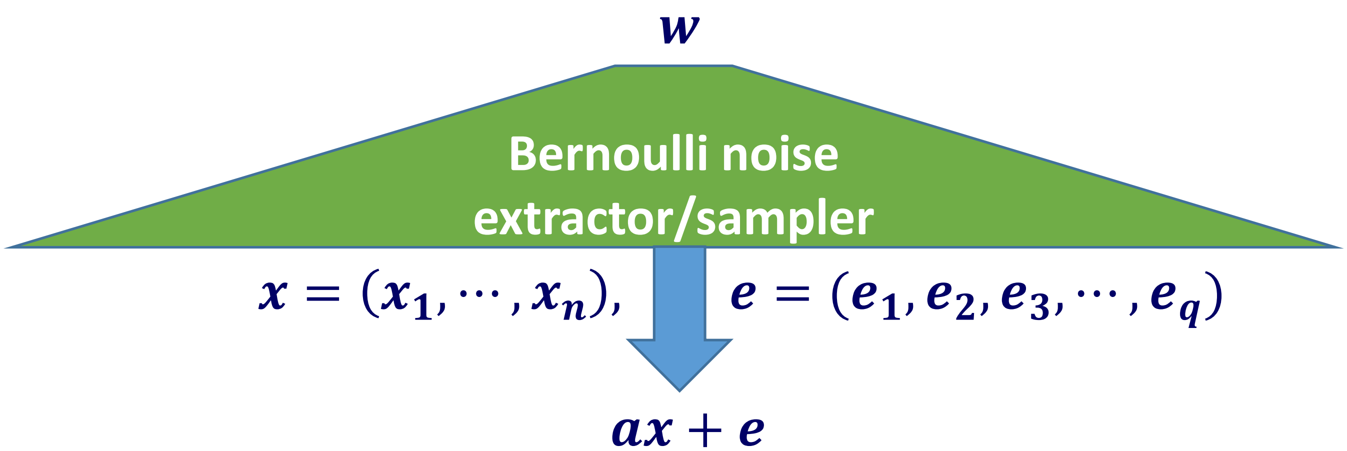
where $L(\cdot)$ is deterministic, $G_a(x) = L(a \cdot x)$, $F_x(a) = L(a \cdot x)$

[Akavia et al.14]: may not work!

our approach: convert entropy source w into Bernoulli noise

Overview: LPN-based randomized PRG

- Input: (possibly weak) seed w and public coin a
- Noise sampling: convert (almost all entropy of) w into Bernoulli-like noise (x, e)
- Output: $\boldsymbol{G_a}(\boldsymbol{w}) = \boldsymbol{a}x + \boldsymbol{e}$



- **Theorem**: Assume that the decisional LPN is ($q = \big(1 + O(1)\big)n,\ t,\ \varepsilon$)-hard on secret of size n and noise rate $\mu = n^{-c} (0 < c < 1)$,

then $\boldsymbol{G_a}$ is a ($t - poly(n),\ O(\varepsilon)$)-hard randomized PRG in $\mathrm{AC_0(mod\ 2)}$ on
  - ➢weak seed $\boldsymbol{w}$ of entropy $O(n^{1-c} \cdot \log n)$
  - ➢uniform seed $\boldsymbol{w}$ of size $O(n^{1-c} \cdot \log n)$

Bernoulli Noise Extractor

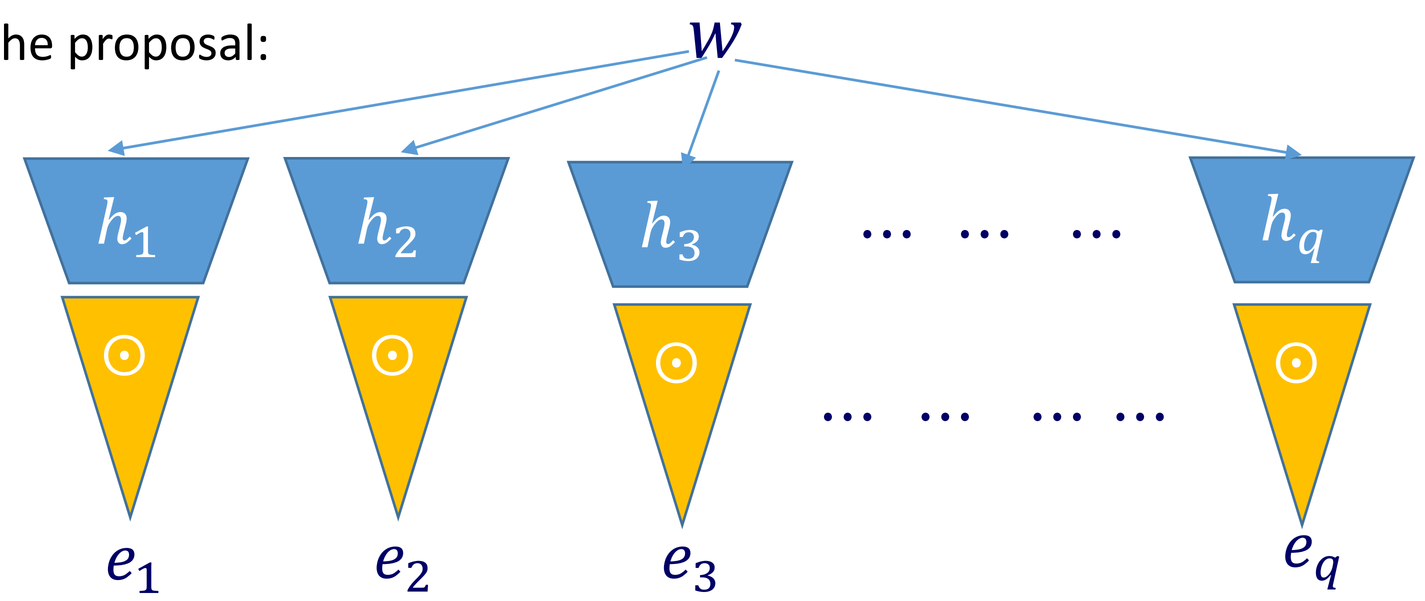
- Sample Ber_μ $\left(\mu = 2^{-i}\right)$: output $\odot(w_1, \cdots, w_i) = w_1 w_2 \cdots w_i$
- For $\mu = n^{-c}$ $(i = c\log n)$, Shannon entropy $\mathbf{H}(\mathrm{Ber}_\mu) \approx \mu\log(1/\mu)$

λ random bits → $(\lambda/\, i) = O(\lambda/\log n)$ Bernoulli bits

in theory: λ random bits → $\lambda/\mathrm{H}(\mathrm{Ber}_\mu) \approx O(\lambda n^c/\log n)$ Bernoulli bits

[Applebaum et al.09]: w remains a lot of entropy given the noise sampled

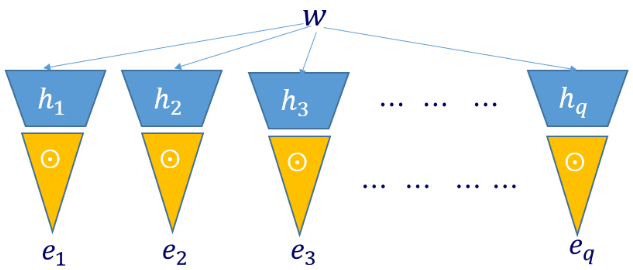
- The proposal:



$h_1, h_2, \cdots, h_q$: 2-wise independent hash functions (randomized by a)

Bernoulli Noise Extractor (cont'd)

- The extractor is in AC0 (mod 2)



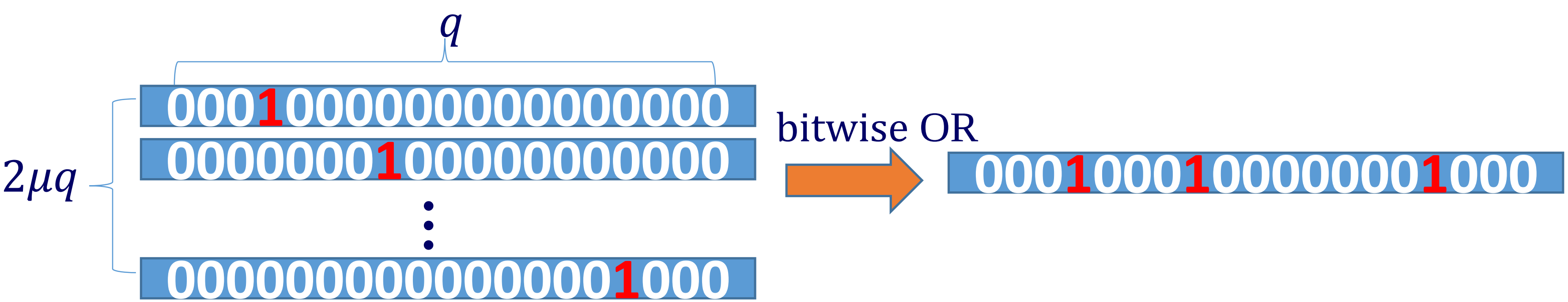
- **Theorem (informal)**: Let $h_1, h_2, \cdots, h_q$ be 2-wise independent hash functions, for any source w of Renyi entropy λ, for any constant $0 < \Delta \leq 1$,

$$\text{Stat-Dist}\left(\left(a, (e_1, \cdots e_q)\right), \left(a, \mathrm{Ber}_\mu^q\right) \right) < 2^{\frac{(1+\Delta)\mathrm{H}\left(\mathrm{Ber}_\mu^q\right) - \lambda}{2}} + 2^{-\Delta^2 \mu q/3}$$

- Parameters: $\mu = n^{-c}$, set $q = \Omega(n)$, $\lambda = (1+2\Delta)\mathrm{H}\left(\mathrm{Ber}_\mu^q\right) = \Omega(n^{1-c} \cdot \log n)$
- PRG's stretch: $\frac{\text{output length}}{\text{input length}} = \frac{q-n}{\lambda} = n^{\Omega(1)}$
- Proof: $\boxed{\text{Cauchy-Schwarz + 2-wise independence like the crooked LHL [Dodis \& Smith 05]}}$ + $\boxed{\text{flattening Shannon entropy Chernoff [HILL99]}}$

An alternative: Bernoulli noise sampler

- Use uniform randomness (~~weak random source~~), and do it in AC_0 ($\cancel{\mathrm{AC}_0(\mathrm{mod}\ 2)}$)
- The idea: take conjunction of $2\mu q$ copies of random Hamming-weight-1 distributions



- The above distribution (denoted as ψ_μ^q) need $2\mu q(\log q)$ uniform random bits
- Asymptotically optimal: for $\mu = n^{-c}, q = poly(n)$, $2\mu q \log q = O(\mathrm{H}(\mathrm{Ber}_\mu^q))$
- PRG: $\boldsymbol{G_a}(\boldsymbol{w}) = \boldsymbol{a}x + \boldsymbol{e}$ by sampling $(x, e) \leftarrow \psi_\mu^{n+q}$ from uniform w
- Theorem: $\boldsymbol{G_a}$ is a randomized PRG of seed length $O(n^{1-c}\log n)$ with comparable security to the underlying standard LPN of secret size n .

Proof. (1) computational LPN → computational ψ_μ^{n+q}-LPN

(2) computational ψ_μ^{n+q} LPN → decisional ψ_μ^{n+q}-LPN

sample-preserving reduction by [Applebaum et al.07]

Randomized PRGs to PRFs

- Given **randomized PRG** $G_a\colon \{0,1\}^n \times \{0,1\}^m \to \{0,1\}^{n^2}$ in $\mathrm{AC}_0(\mathrm{mod}\ 2)$

  how to construct a **PRF** in $\widetilde{\mathrm{AC}}_0(\mathrm{mod}\ 2)$?

① a PRF of input size $\omega(\log n)$: n-ary GGM tree of depth $d = \omega(1)$

$$G_a\ (k) \stackrel{\text{def}}{=} \underbrace{\underbrace{G_a^{0\cdots00}(k)}_{\text{n bits}}\ \underbrace{G_a^{0\cdots01}(k)}_{\text{n bits}} \cdots \underbrace{G_a^{1\cdots11}(k)}_{n\ \text{bits}}}_{\text{n blocks}}$$

$$F_{k,a}(x_1 \cdots x_{d\log n}) \stackrel{\text{def}}{=} G_a^{x_{(d-1)\log n+1} \cdots x_{d\log n}}(\cdots G_a^{x_{\log n+1} \cdots x_{2\log n}}(G_a^{x_1 \cdots x_{\log n}}(k))\cdots)$$

② Domain extension from $\{0,1\}^{\omega(\log n)}$ to $\{0,1\}^{\mathrm{n}}$ (w. security & depth preserved)

Generalized Levin's trick:

$$F'_{k,a}(x) \stackrel{\text{def}}{=} F_{k_1,a}(h_1(x)) \oplus F_{k_2,a}(h_2(x)) \oplus \cdots \oplus F_{k_l,a}(h_l(x))$$

universal hash functions $h_1, \cdots, h_l\colon \{0,1\}^{\mathrm{n}} \to \{0,1\}^{\omega(\log n)}$, $\quad k \stackrel{\text{def}}{=} (k_1, h_1, \cdots, k_l h_l)$

Randomized PRGs to PRFs (cont'd)

Theorem [Generalized Levin's trick]: For random functions $R_1, \cdots, R_l : \{0,1\}^{\omega(logn)} \to \{0,1\}^n$ and universal hash functions $h_1, \cdots, h_l : \{0,1\}^n \to \{0,1\}^{\omega(logn)}$, let

$$R'(x) \stackrel{\text{def}}{=} R_1(h_1(x)) \oplus R_2(h_2(x)) \oplus \cdots \oplus R_l(h_l(x))$$

Then, R' is $q\left(\frac{q}{n^{\omega(1)}}\right)^l$-indistinguishable from random function $\{0,1\}^n \to \{0,1\}^n$ for any (computationally unbounded) adversary making up to q oracle queries.

- See [Bellare et al.99] [Maurer 02][Dottling,Schröder15] [Gazi&Tessaro15]
- Our proof: using the Patarin's H-coefficient technique
- Security is preserved for q=$n^{\omega(1)}$ and $l = O(n/\text{log}n)$

Theorem [The PRF] Assume the decisional LPN is ($q = \big(1 + O(1)\big)n,\ t,\ \varepsilon$)-hard on secret of size n and noise rate $\mu = n^{-c}(0 < c < 1)$, then for any $\omega(1)$ there exists ($q = n^{\omega(1)}, t - poly(q,n),\ O(dq\varepsilon)$)-hard randomized PRF $F'_{k,a}$ in $\widetilde{\text{AC}}_0(\text{mod } 2)$ of depth $\omega(1)$ on any weak key k of entropy $O(n^{1-c} \cdot \log n)$.

Conclusion and open problems

From low-noise LPN we construct:

- Polynomial-stretch pseudorandom generators (PRGs) in $\mathrm{AC_0(mod\ 2)}$
- Pseudorandom functions (PRFs) in $\widetilde{\mathrm{AC}}_0\mathrm{(mod\ 2)}$
  - Same (actually better) t/ϵ security than the underlying LPN
    seed/key of entropy $\lambda = n^{1-c}\log n$ with t/ϵ security up to $2^{O(n^{1-c})} = 2^{O(\lambda/\log\lambda)}$
  - Query complexity $q = n^{\omega(1)}$. $\omega(1)$: depth of the circuit.

- Open problems
  - LPN-based PRFs in constant depth
    - weak PRFs in $\mathrm{AC_0(mod\ 2)}$
    - PRFs in $\mathrm{TC_0}$
  - More cryptomania objects from LPN?
    - Collision Resistant Hash Function (CRHF)
    - Fully Homomorphic Encryption (FHE)
    - Etc.

Thank you!