Pseudorandom Functions in Almost Constant Depth from Low-Noise LPN

Yu Yu* John Steinberger†

February 21, 2016

Abstract

Pseudorandom functions (PRFs) play a central role in symmetric cryptography. While in principle they can be built from any one-way functions by going through the generic HILL (SICOMP 1999) and GGM (JACM 1986) transforms, some of these steps are inherently sequential and far from practical. Naor, Reingold (FOCS 1997) and Rosen (SICOMP 2002) gave parallelizable constructions of PRFs in NC^2 and TC^0 based on concrete number-theoretic assumptions such as DDH, RSA, and factoring. Banerjee, Peikert, and Rosen (Eurocrypt 2012) constructed relatively more efficient PRFs in NC^1 and TC^0 based on "learning with errors" (LWE) for certain range of parameters. It remains an open problem whether parallelizable PRFs can be based on the "learning parity with noise" (LPN) problem for both theoretical interests and efficiency reasons (as the many modular multiplications and additions in LWE would then be simplified to AND and XOR operations under LPN).

In this paper, we give more efficient and parallelizable constructions of randomized PRFs from LPN under noise rate n^{-c} (for any constant $0 < c < 1$) and they can be implemented with a family of polynomial-size circuits with unbounded fan-in AND, OR and XOR gates of depth $\omega(1)$, where $\omega(1)$ can be any small super-constant (e.g., $\log\log\log n$ or even less). Our work complements the lower bound results by Razborov and Rudich (STOC 1994) that PRFs of beyond quasi-polynomial security are not contained in $AC^0(MOD_2)$, i.e., the class of polynomial-size, constant-depth circuit families with unbounded fan-in AND, OR, and XOR gates.

Furthermore, our constructions are security-lifting by exploiting the redundancy of low-noise LPN. We show that in addition to parallelizability (in almost constant depth) the PRF enjoys either of (or any tradeoff between) the following:

- A PRF on a weak key of sublinear entropy (or equivalently, a uniform key that leaks any $(1 - o(1))$-fraction) has comparable security to the underlying LPN on a linear size secret.
- A PRF with key length λ can have security up to $2^{O(\lambda/\log\lambda)}$, which goes much beyond the security level of the underlying low-noise LPN.

where adversary makes up to certain super-polynomial amount of queries.

Keywords: Foundations, Symmetric Cryptography, Low-depth PRFs, Learning Parity with Noise.

1 Introduction

Learning Parity with Noise. The computational version of learning parity with noise (LPN) assumption with parameters $n \in \mathbb{N}$ (length of secret), $q \in \mathbb{N}$ (number of queries) and $0 < \mu < 1/2$ (noise rate) postulates that it is computationally infeasible to recover the n-bit secret $s \in \{0,1\}^n$ given $(a{\cdot}s{\oplus}e,\ a)$, where a is a random $q{\times}n$ matrix, e follows Ber_μ^q, Ber_μ denotes the Bernoulli distribution with

*Shanghai Jiao Tong University. Email: `yuyuathk@gmail.com`
†Tsinghua University. Email:`jpsteinb@gmail.com`

parameter μ (i.e., $\Pr[\mathsf{Ber}_\mu = 1] = \mu$ and $\Pr[\mathsf{Ber}_\mu = 0] = 1 - \mu$), '$\cdot$' denotes matrix vector multiplication over GF(2) and '$\oplus$' denotes bitwise XOR. The decisional version of LPN simply assumes that $a \cdot s \oplus e$ is pseudorandom (i.e., computationally indistinguishable from uniform randomness) given a. The two versions are polynomially equivalent [12, 36, 5].

Hardness of LPN. The computational LPN problem represents a well-known NP-complete problem "decoding random linear codes" [9] and thus its worst-case hardness is well studied. LPN was also extensively studied in learning theory, and it was shown in [24] that an efficient algorithm for LPN would allow to learn several important function classes such as 2-DNF formulas, juntas, and any function with a sparse Fourier spectrum. Under a constant noise rate (i.e., $\mu = \Theta(1)$), the best known LPN solvers [13, 40] require time and query complexity both $2^{O(n/\log n)}$. The time complexity goes up to $2^{O(n/\log\log n)}$ when restricted to $q = \mathsf{poly}(n)$ queries [42], or even $2^{O(n)}$ given only $q = O(n)$ queries [45]. Under low noise rate $\mu = n^{-c}$ $(0 < c < 1)$, the security of LPN is less well understood: on the one hand, for $q = n + O(1)$ we can already do efficient distinguishing attacks with advantage $2^{-O(n^{1-c})}$ that match the statistical distance between the LPN samples and uniform randomness (see Remark 4.1); on the other hand, for (even super-)polynomial q the best known attacks [54, 15, 11, 39, 7] are not asymptotically better, i.e., still at the order of $2^{\Theta(n^{1-c})}$. We mention that LPN does not succumb to known quantum algorithms, which makes it a promising candidate for "post-quantum cryptography". Furthermore, LPN also enjoys simplicity and is more suited for weak-power devices (e.g., RFID tags) than other quantum-secure candidates such as LWE [52] [1].

LPN-based Cryptographic Applications. LPN was used as a basis for building lightweight authentication schemes against passive [31] and even active adversaries [35, 36] (see [1] for a more complete literature). Recently, Kiltz et al. [38] and Dodis et al. [20] constructed randomized MACs based on the hardness of LPN, which implies a two-round authentication scheme with man-in-the-middle security. Lyubashevsky and Masny [43] gave an more efficient three-round authentication scheme from LPN (without going through the MAC transformation) and recently Cash, Kiltz, and Tessaro [16] reduced the round complexity to 2 rounds. Applebaum et al. [4] showed how to constructed a linear-stretch[2] pseudorandom generator (PRG) from LPN. We mention other not-so-relevant applications such as public-key encryption schemes [3, 22, 37], oblivious transfer [19], commitment schemes and zero-knowledge proofs [33], and refer to a recent survey [49] on the current state-of-the-art about LPN.

Does LPN imply low-depth PRFs? Pseudorandom functions (PRFs) play a central role in symmetric cryptography. While in principle PRFs can be obtained via a generic transform from any one-way function [29, 26], these constructions are inherently sequential and too inefficient to compete with practical instantiations (e.g., the AES block cipher) built from scratch. Motivated by this, Naor, Reingold [46] and Rosen [47] gave direct constructions of PRFs from concrete number-theoretic assumptions (such as decision Diffie-Hellman, RSA, and factoring), which can be computed by low-depth circuits in NC^2 or even TC^0. However, these constructions mainly established the feasibility result and are far from practical as they require extensive preprocessing and many exponentiations in large multiplicative groups. Banerjee, Peikert, and Rosen [6] constructed relatively more efficient PRFs in NC^1 and TC^0 based on the "learning with errors" (LWE) assumption. More specifically, they observed that LWE for certain range of parameters implies a deterministic variant which they call "learning with rounding" (LWR), and that LWR in turn gives rise to pseudorandom synthesizers [46], a useful tool for building low-depth PRFs. Despite that LWE is generalized from LPN, the derandomization technique used for LWE [6] does not seemingly apply to LPN, and thus it is an interesting open problem if low-depth PRFs can be based on (even a low-noise variant of) LPN (see a discussion in [49, Footnote 18]). In fact, we don't even know how to build low-depth weak PRFs from LPN. Applebaum [4] observed that

[1] The inner product of LWE requires many multiplications modulo a large prime p (polynomial in the security parameter), and in contrast the same operation for LPN is simply an XOR sum of a few AND products.

[2] A PRG $G : \{0,1\}^{\ell_1} \to \{0,1\}^{\ell_2}$ has linear stretch if the stretch factor ℓ_2/ℓ_1 equals some constant greater than 1.

LPN implies "weak randomized pseudorandom functions", which require independent secret coins on every function evaluation, and Akavia et al. [2] obtained weak PRFs in "$AC^0 \circ MOD_2$" from a relevant non-standard hard learning assumption.

Our contributions. In this paper, we give constructions of low-depth PRFs from low-noise LPN (see Theorem 1.1 below), where the noise rate n^{-c} (for any constant $0<c<1$) encompasses the noise level of Alekhnovich [3] (i.e., $c=1/2$) and higher noise regime. Strictly speaking, the PRFs we obtain are not contained in $AC^0(MOD_2)$[3], but the circuit depth $\omega(1)$ can be arbitrarily small (e.g., $\log\log\log n$ or even less). This complements the negative result of Razborov and Rudich [51] (which is based on the works of Razborov and Smolensky [50, 53]) that PRFs with more than quasi-polynomial security do not exist in $AC^0(MOD_2)$.

Theorem 1.1 (main results, informal) *Assume that the LPN problem with secret length n and noise rate $\mu = n^{-c}$ (for any constant $0<c<1$) is $(q=1.001n,\ t=2^{O(n^{1-c})}, \epsilon=2^{-O(n^{1-c})})$-hard[4]. Then,*

1. *for any $d=\omega(1)$, there exists a $(q'=n^{d/3}, t-q'\mathsf{poly}(n), O(nq'\epsilon))$-randomized-PRF on any weak key of Rényi entropy no less than $O(n^{1-c}\cdot\log n)$, or on an $n^{1-\frac{c}{2}}$-bit uniform random key with any $(1-\frac{O(\log n)}{n^{c/2}})$-fraction of leakage (independent of the public coins of the PRF);*
2. *let $\lambda = \Theta(n^{1-c}\log n)$, for any $d=\omega(1)$, there exists a $(q'=\lambda^{\Theta(d)},\ t'=2^{O(\lambda/\log\lambda)},\ \epsilon'=2^{-O(\lambda/\log\lambda)})$-randomized PRF with key length λ;*

where both PRFs are computable by polynomial-size depth-$O(d)$ circuits with unbounded-fan-in AND, OR and XOR gates.

On lifted security. Note that there is nothing special with the factor 1.001, which can be replaced with any constant greater than 1. The first parallelizable PRF has security[5] comparable to the underlying LPN (with linear secret length) yet it uses a key of only sublinear entropy, or in the language of leakage resilient cryptography, a sublinear-size secret key with any $(1-o(1))$-fraction of leakage (independent of the public coins). From a different perspective, let the security parameter λ be the key length of the PRF, then the second PRF can have security up to $2^{O(\lambda/\log\lambda)}$ given any $n^{\Theta(d)}$ number of queries. We use security-preserving PRF constructions without relying on k-wise independent hash functions. This is crucial for low-depth constructions as recent works [34, 17] use (almost) $\omega(\log n)$-wise independent hash functions, which are not known to be computable in (almost) constant-depth even with unbounded fan-in gates. We remark that circuit depth $d=\omega(1)$ is independent of the time/advantage security of PRF, and is reflected only in the query complexity $q'=n^{\Theta(d)}$. This is reasonable in many scenarios as in practice the number of queries may depend not only on adversary's computing power but also on the amount of data available for cryptanalysis. It remains open whether the dependency of query complexity on circuit depth can be fully eliminated.

Bernoulli-like Randomness Extractor/Sampler. Of independent interests, we propose the following randomness extractor/sampler in constant depth and they are used in the first/second PRF constructions respectively.

- A Bernoulli randomness extractor in $AC^0(MOD_2)$ that converts almost all entropy of a weak Rényi entropy source into Bernoulli noise distributions.

[3] Recall that $AC^0(MOD_2)$ refers to the class of polynomial-size, constant-depth circuit families with unbounded fan-in AND, OR, and XOR gates.

[4] t and $1/\epsilon$ are upper bounded by $2^{O(n^{1-c})}$ due to known attacks.

[5] Informally, we say that a PRF has security T if it is $1/T$-indistinguishable from a random function for all oracle-aid distinguishers running in time T and making up to certain superpolynomial number of queries.

- A sampler in AC^0 that uses a short uniform seed and outputs a Bernoulli-like distribution of length m and noise rate μ, denoted as ψ_μ^m (see Algorithm 1).

Alekhnovich's cryptosystem [3] considers a random distribution of length m that has exactly μm 1's, which we denote as $\chi_{\mu m}^m$. The problem of sampling $\chi_{\mu m}^m$ dates back to [12], but the authors only mention that it can be done efficiently, and it is not known whether $\chi_{\mu m}^m$ can be sampled in $AC^0(MOD_2)$. Instead, Applebaum et al. [4] propose the following sampler for Bernoulli distribution Ber_μ^q using uniform randomness. Let $w = w_1 \cdots w_n$ be an n-bit uniform random string, and for convenience assume that μ is a negative power of 2 (i.e., $\mu = 2^{-v}$ for integer v). Let $\mathsf{sample} : \{0,1\}^v \to \{0,1\}$ output the AND of its input bits, and let

$$e = (\mathsf{sample}(w_1 \cdots w_v), \cdots, \mathsf{sample}(w_{(q-1)v+1} \cdots w_{(q-1)v+v}))$$

so that $e \sim \mathsf{Ber}_\mu^q$ for any $q \le \lfloor n/\log(1/\mu) \rfloor$. Note that Ber_μ has Shannon entropy $\mathbf{H}_1(\mathsf{Ber}_\mu) = \Theta(\mu \log(1/\mu))$ (see Fact A.1), and thus the above converts a $(q\mathbf{H}_1(\mathsf{Ber}_\mu)/n) = O(\mu)$-fraction of the entropy into Bernoulli randomness. It was observed in [4] that conditioned on e source w remains of $(1-O(\mu))n$ bits of average min-entropy, which can be recycled into uniform randomness with a universal hash function h. That is, the two distributions are statistically close

$$(\ e,\ h(w)\ ,h\ ) \stackrel{s}{\sim} (\ \mathsf{Ber}_\mu^q,\ U_{(1-O(\mu))n}\ ,h\ )\ ,$$

where U_q denotes a uniform distribution over $\{0,1\}^q$. The work of [4] then proceeded to a construction of PRG under noise rate $\mu = \Theta(1)$. However, for $\mu = n^{-c}$ the above only samples an $O(n^{-c})$-fraction of entropy. To convert more entropy into Bernoulli distributions, one may need to apply the above sample-then-recycle process to the uniform randomness recycled from a previous round (e.g., $h(w)$ of the first round) and repeat the process many times. However, this method is sequential and requires a circuit of depth $\Omega(n^c)$ to convert any constant fraction of entropy. We propose a more efficient and parallelizable extractor in $AC^0(MOD_2)$. As shown in Figure 1, given any weak source of Rényi entropy $\Theta(n)$, we apply i.i.d. pairwise independent hash functions $h_1, \cdots, h_q$ (each of output length v) to w and then use sample on the bits extracted to get the Bernoulli distributions. We prove a lemma showing that this method can transform almost all entropy into Bernoulli distribution Ber_μ^q, namely, the number of extracted Bernoulli bits q can be up to $\Theta(n/\mathbf{H}_1(\mathsf{Ber}_\mu))$. This immediately gives an equivalent formulation of the standard LPN by reusing matrix a to randomize the hash functions. For example, for each $1 \le i \le q$ denote by a_i the i-th row of a, let h_i be described by a_i, and let i-th LPN sample be $\langle a_i, s\rangle \oplus \mathsf{sample}(h_i(w))$. Note that the algorithm is non-trivial as $(h_1(w), \cdots, h_q(w))$ can be of length $\Theta(n^{1+c})$, which is much greater than the entropy of w.

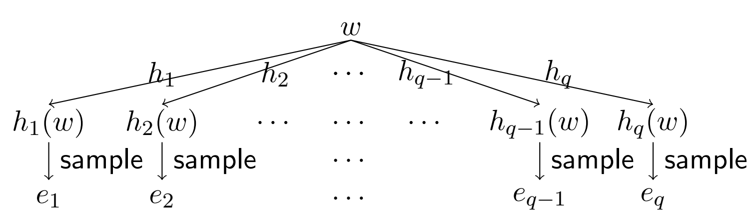

Figure 1: An illustration of the proposed Bernoulli randomness extractor in $AC^0(MOD_2)$.

The Bernoulli randomness extractor is used in the first PRF construction. For our second construction, we introduce a Bernoulli-like distribution ψ_μ^m that can be more efficiently sampled in AC^0 (i.e., without using XOR gates), and show that it can be used in place of Ber_μ^m with provable security.

PRGs and PRFs from LPN. It can be shown that standard LPN implies a variant where the secret s and noise vector e are sampled from Ber_μ^{n+q} or even ψ_μ^{n+q}. This allows us to obtain a randomized

PRG G_a with short seed and polynomial stretch, where a denotes the public coin. We then use the technique of Goldreich, Goldwasser and Micali [26] with a $n^{\Theta(1)}$-ary tree of depth $\omega(1)$ (reusing public coin a at every invocation of G_a) and construct a randomized PRF (see Definition 2.4) $F_{k,a}$ with input length $\omega(\log n)$, secret key k and public coin a. This already implies PRFs of arbitrary input length by Levin's trick [41], i.e., $\bar{F}_{(k,h),a}(x) \stackrel{\text{def}}{=} F_{k,a}(h(x))$ where h is a universal hash function from any fixed-length input to $\omega(\log n)$ bits. Note that $\bar{F}_{(k,h),a}$ is computable in depth $\omega(1)$ (i.e., the depth of the GGM tree) for any small $\omega(1)$. However, the security of the above does not go beyond $n^{\omega(1)}$ due to a birthday attack. To overcome this, we use a simple and parallel method [8, 44] by running a sub-linear number of independent[6] copies of $\bar{F}_{(k,h),a}$ and XORing their outputs, and we avoid key expansions by using pseudorandom keys (expanded using G_a or $F_{k,a}$) for all copies of $\bar{F}_{(k,h),a}$. We obtain our final security-preserving construction of PRFs by putting together all the above ingredients.

The rest of the paper is organized as follows: Section 2 gives background information about relevant notions and definitions. Section 3 presents the Bernoulli randomness extractor. Section 4 and Section 5 give the two constructions of PRFs respectively. We include in Appendix A well-known lemmas and inequalities used, and refer to Appendix B for all the proofs omitted in the main text.

2 Preliminaries

Notations and definitions. We use $[n]$ to denote set $\{1, \ldots, n\}$. We use capital letters[7] (e.g., X, Y) for random variables and distributions, standard letters (e.g., x, y) for values, and calligraphic letters (e.g. $\mathcal{X}$, $\mathcal{E}$) for sets and events. The support of a random variable X, denoted by $\mathsf{Supp}(X)$, refers to the set of values on which X takes with non-zero probability, i.e., $\{x : \Pr[X = x] > 0\}$. Denote by $|\mathcal{S}|$ the cardinality of set $\mathcal{S}$. We use Ber_μ to denote the Bernoulli distribution with parameter μ, i.e., $\Pr[\mathsf{Ber}_\mu = 1] = \mu$, $\Pr[\mathsf{Ber}_\mu = 0] = 1 - \mu$, while Ber_μ^q denotes the concatenation of q independent copies of Ber_μ. We use χ_i^q, $i \le q$, to denote a uniform distribution over $\{e \in \{0,1\}^q : |e| = i\}$, where $|e|$ denotes the Hamming weight of binary string e. For $n \in \mathbb{N}$, U_n denotes the uniform distribution over $\{0,1\}^n$ and independent of any other random variables in consideration, and $f(U_n)$ denotes the distribution induced by applying the function f to U_n. $X{\sim}D$ denotes that random variable X follows distribution D. We use $s \leftarrow S$ to denote sampling an element s according to distribution S, and let $s \xleftarrow{\$} \mathcal{S}$ denote sampling s uniformly from set $\mathcal{S}$.

Entropy definitions. For a random variable X and any $x \in \mathsf{Supp}(X)$, the sample-entropy of x with respect to X is defined as

$$\mathbf{H}_X(x) \stackrel{\text{def}}{=} \log(1/\Pr[X = x])$$

from which we define the Shannon entropy, Rényi entropy and min-entropy of X respectively, i.e.,

$$\mathbf{H}_1(X) \stackrel{\text{def}}{=} \mathbb{E}_{x \leftarrow X}[\ \mathbf{H}_X(x)\ ],\ \ \mathbf{H}_2 \stackrel{\text{def}}{=} -\log \sum_{x \in \mathsf{Supp}(X)} 2^{-2\mathbf{H}_X(x)},\ \ \ \mathbf{H}_\infty(X) \stackrel{\text{def}}{=} \min_{x \in \mathsf{Supp}(X)} \mathbf{H}_X(x).$$

For $0 < \mu < 1/2$, let $\mathbf{H}(\mu) \stackrel{\text{def}}{=} \mu \log(1/\mu) + (1-\mu)\log(1/(1-\mu))$ be the binary entropy function so that $\mathbf{H}(\mu) = \mathbf{H}_1(\mathsf{Ber}_\mu)$. We know that $\mathbf{H}_1(X) \ge \mathbf{H}_2(X) \ge \mathbf{H}_\infty(X)$ with equality when X is uniformly distributed. A random variable X of length n is called an (n, λ)-Rényi entropy (resp., min-entropy) source if $\mathbf{H}_2(X) \ge \lambda$ (resp., $\mathbf{H}_\infty(X) \ge \lambda$). The *statistical distance* between X and Y, denoted by $\mathsf{SD}(X, Y)$, is defined by

$$\mathsf{SD}(X, Y) \stackrel{\mathsf{def}}{=} \frac{1}{2} \sum_x |\Pr[X = x] - \Pr[Y = x]|$$

[6]By "independent" we mean that $\bar{F}_{(k,h),a}$ is evaluated on independent keys but still reusing the same public coin a.
[7]The two exceptions are G and F, which are reserved for PRGs and PRFs respectively.

We use $\mathsf{SD}(X, Y|Z)$ as a shorthand for $\mathsf{SD}((X, Z), (Y, Z))$.

Simplifying Notations. To simplify the presentation, we use the following simplified notations. Throughout, n is the security parameter and most other parameters are functions of n, and we often omit n when clear from the context. For example, $\mu = \mu(n) \in (0, 1/2)$, $q = q(n) \in \mathbb{N}$, $t = t(n) > 0$, $\epsilon = \epsilon(n) \in (0, 1)$, and $m = m(n) = \mathsf{poly}(n)$, where poly refers to some polynomial.

Definition 2.1 (computational/decisional LPN) *Let n be a security parameter, and let μ, q, t and ϵ all be functions of n. The **decisional** $\mathsf{LPN}_{\mu,n}$ problem (with secret length n and noise rate μ) is (q, t, ϵ)-hard if for every probabilistic distinguisher D running in time t we have*

$$\Big| \Pr_{A,S,E}[\ \mathsf{D}(A,\ A{\cdot}S \oplus E) = 1\ ] \ - \ \Pr_{A,U_q}[\ \mathsf{D}(A, U_q) = 1\ ]\Big| \ \le\ \epsilon \tag{1}$$

*where $A \sim U_{qn}$ is a $q \times n$ matrix, $S \sim U_n$ and $E \sim \mathsf{Ber}_\mu^q$. The **computational** $\mathsf{LPN}_{\mu,n}$ problem is (q, t, ϵ)-hard if for every probabilistic algorithm D running in time t we have*

$$\Pr_{A,S,E}[\ \mathsf{D}(A,\ A{\cdot}S \oplus E) = (S, E)\ ] \ \le\ \epsilon,$$

where $A \sim U_{qn}$, $S \sim U_n$ and $E \sim \mathsf{Ber}_\mu^q$.

Definition 2.2 (LPN variants) *The decisional/computational X-$\mathsf{LPN}_{\mu,n}$ is defined as per Definition 2.1 accordingly except that (S, E) follows distribution X.*
Note that standard $\mathsf{LPN}_{\mu,n}$ is a special case of X-$\mathsf{LPN}_{\mu,n}$ for $X \sim (U_n, \mathsf{Ber}_\mu^q)$.

In respect of the randomized feature of LPN, we generalize standard PRGs / PRFs to equivalent randomized variants, where the generator/function additionally uses some public coins for randomization, and that seed/key can be sampled from a weak source (independent of the public coins).

Definition 2.3 (randomized PRGs on weak seeds) *Let $\lambda \le \ell_1 < \ell_2, \ell_3, t, \epsilon$ be functions of security parameter n. An efficient function family ensemble $\mathcal{G} = \{G_a : \{0,1\}^{\ell_1} \to \{0,1\}^{\ell_2}, a \in \{0,1\}^{\ell_3}\}_{n\in\mathbb{N}}$ is a (t, ϵ) randomized PRG on (ℓ_1, λ)-weak seed if for every probabilistic distinguisher D of running time t and every (ℓ_1, λ)-Rényi entropy source K it holds that*

$$\Big| \Pr_{K,A\sim U_{\ell_3}}[\ \mathsf{D}(G_A(K), A) = 1\ ] \ - \ \Pr_{U_{\ell_2},A\sim U_{\ell_3}}[\ \mathsf{D}(U_{\ell_2}, A) = 1\ ]\Big| \ \le\ \epsilon\ .$$

The stretch factor of $\mathcal{G}$ is ℓ_2/ℓ_1. Standard (deterministic) PRGs are implied by defining $G'(k, a) \stackrel{def}{=} (G_a(k), a)$ for a uniform random k.

Definition 2.4 (randomized PRFs on weak keys) *Let $\lambda \le \ell_1, \ell_2, \ell_3, \ell, t, \epsilon$ be functions of security parameter n. An efficient function family ensemble $\mathcal{F} = \{F_{k,a} : \{0,1\}^{\ell} \to \{0,1\}^{\ell_2}, k \in \{0,1\}^{\ell_1}, a \in \{0,1\}^{\ell_3}\}_{n\in\mathbb{N}}$ is a (q, t, ϵ) randomized PRF on (ℓ_1, λ)-weak key if for every oracle-aided probabilistic distinguisher D of running time t and bounded by q queries and for every (ℓ_1, λ)-Rényi entropy source K we have*

$$\Big| \Pr_{K,A\sim U_{\ell_3}}[\ \mathsf{D}^{F_{K,A}}(A) = 1\ ] \ - \ \Pr_{R,A\sim U_{\ell_3}}[\ \mathsf{D}^{R}(A) = 1\ ]\Big| \ \le\ \epsilon(n),$$

where R denotes a random function distribution ensemble mapping from ℓ bits to ℓ_2 bits. Standard PRFs are a special case for empty a (or keeping $k' = (k, a)$ secret) on uniformly random key.

Definition 2.5 (universal hashing) *A function family $\mathcal{H} = \{h_a\ :\ \{0,1\}^n \to \{0,1\}^m, a \in \{0,1\}^l\}$ is **universal** if for any $x_1 \ne x_2 \in \{0,1\}^n$ it holds that*

$$\Pr_{a \xleftarrow{\$} \{0,1\}^l}[\ h_a(x_1)\ =\ h_a(x_2)\ ] \le 2^{-m}.$$

Definition 2.6 (pairwise independent hashing) *A function family* $\mathcal{H} = \{ h_a\colon \{0,1\}^n \to \{0,1\}^m, a \in \{0,1\}^l \}$ *is* ***pairwise independent*** *if for any* $x_1 \neq x_2 \in \{0,1\}^n$ *and any* $v \in \{0,1\}^{2m}$ *it holds that*

$$\Pr_{a \xleftarrow{\$} \{0,1\}^l} [\ (h_a(x_1), h_a(x_2)) = v\ ] = 2^{-2m}.$$

Concrete constructions. We know that for every $m \leq n$ there exists a pairwise independent (and universal) $\mathcal{H}$ with description length $l = \Theta(n)$, where every $h \in \mathcal{H}$ can be computed in $\mathrm{AC}^0(\mathrm{MOD}_2)$. For example, $\mathcal{H}_1$ and $\mathcal{H}_2$ defined below are universal and pairwise independent respectively:

$$\mathcal{H}_1 = \{\ h_a : \{0,1\}^n \to \{0,1\}^m\ \mid\ h_a(x) \stackrel{\text{def}}{=} a \cdot x,\ a \in \{0,1\}^{n+m-1}\ \}$$

$$\mathcal{H}_2 = \left\{ h_{a,b} : \{0,1\}^n \to \{0,1\}^m \mid\ h_{a,b}(x) \stackrel{\text{def}}{=} a \cdot x \oplus b, a \in \{0,1\}^{n+m-1}, b \in \{0,1\}^m \right\}$$

where $a \in \{0,1\}^{n+m-1}$ is interpreted as an $m \times n$ Toeplitz matrix and '$\cdot$' and '$\oplus$' denote matrix-vector multiplication and addition over GF(2) respectively.

3 Bernoulli Randomness Extraction in $\mathrm{AC}^0(\mathrm{MOD}_2)$

First, we state below a variant of the lemma (e.g., [28]) that taking sufficiently many samples of i.i.d. random variables yields an "almost flat" joint random variable, i.e., the sample-entropy of most values is close to the Shannon entropy of the joint random variable. The proof is included in Appendix B for completeness.

Lemma 3.1 (Flattening Shannon entropy) *For any* $n \in \mathbb{N}$, $0 < \mu < 1/2$ *and for any* $\Delta > 0$ *define*

$$\mathcal{E} \stackrel{def}{=} \{\ \vec{e} \in \{0,1\}^q : \mathbf{H}_{\mathsf{Ber}_\mu^q}(\vec{e}) \leq (1+\Delta)q\mathbf{H}(\mu)\ \}. \tag{2}$$

Then, we have $\Pr[\ \mathsf{Ber}_\mu^q \in \mathcal{E}\ ] \geq 1 - \exp^{-\frac{\min(\Delta,\Delta^2)\mu q}{3}}$.

Lemma 3.2 states that the proposed Bernoulli randomness extractor (see Figure 1) extracts almost all entropy from a Rényi entropy (or min-entropy) source. We mention that the extractor can be considered as a parallelized version of the random bits recycler of Impagliazzo and Zuckerman [32] and the proof technique is also closely relevant to the crooked leftover hash lemma [21, 14].

Lemma 3.2 (Bernoulli randomness extraction) *For any* m, $v \in \mathbb{N}$ *and* $0 < \mu \leq 1/2$, *let* $W \in \mathcal{W}$ *be any* $(\lceil \log |\mathcal{W}| \rceil, m)$*-Rényi entropy source, let* $\mathcal{H}$ *be a family of pairwise independent hash functions mapping from* $\mathcal{W}$ *to* $\{0,1\}^v$, *let* $\vec{H} = (H_1, \ldots, H_q)$ *be a vector of i.i.d. random variables such that each* H_i *is uniformly distributed over* $\mathcal{H}$, *let* $\mathsf{sample} : \{0,1\}^v \to \{0,1\}$ *be any Boolean function such that* $\mathsf{sample}(U_v) \sim \mathsf{Ber}_\mu$. *Then, for any constant* $0 < \Delta \leq 1$ *it holds that*

$$\mathsf{SD}(\ \mathsf{Ber}_\mu^q,\ \mathsf{sample}(\vec{H}(W)) \mid \vec{H}\ ) \ \leq\ 2^{\left((1+\Delta)q\mathbf{H}(\mu)-m\right)/2}\ +\ \exp^{-\frac{\Delta^2\mu q}{3}}\ ,$$

where

$$\mathsf{sample}(\vec{H}(W)) \stackrel{def}{=} (\mathsf{sample}(H_1(W)), \ldots, \mathsf{sample}(H_q(W)))\ .$$

Remark 3.1 (On entropy loss) *The amount of entropy extracted (i.e.,* $q\mathbf{H}(\mu)$*) can be almost as large as entropy of the source (i.e.,* m*) by setting* $m = (1+2\Delta)q\mathbf{H}(\mu)$ *for any arbitrarily small constant* Δ. *Further, the leftover hash lemma falls into a special case for* $v = 1$ *(*sample *being an identity function) and* $\mu = 1/2$.

Proof. Let set $\mathcal{E}$ be defined as in (2). For any $\vec{e}\in\{0,1\}^q$ and $\vec{h}\in\mathcal{H}^q$, use shorthands $p_{\vec{h}}\stackrel{\text{def}}{=}\Pr[\vec{H}=\vec{h}]$, $p_{\vec{e}|\vec{h}}\stackrel{\text{def}}{=}\Pr[\ \mathsf{sample}(\vec{h}(W))=\vec{e}\ ]$ and $p_{\vec{e}}\stackrel{\text{def}}{=}\Pr[\ \mathsf{Ber}_\mu^q=\vec{e}\ ]$. We have

$$\begin{aligned}
&\mathsf{SD}\big(\ (\mathsf{Ber}_\mu^q,\vec{H}),\ (\mathsf{sample}(\vec{H}(W)),\vec{H})\ \big)\\
=\ & \frac{1}{2}\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\in\mathcal{E}} p_{\vec{h}}|\ p_{\vec{e}|\vec{h}}-p_{\vec{e}}\ |\ +\ \frac{1}{2}\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\notin\mathcal{E}} p_{\vec{h}}|\ p_{\vec{e}|\vec{h}}-p_{\vec{e}}\ |\\
\le\ & \frac{1}{2}\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\in\mathcal{E}}\Big(\ \sqrt{p_{\vec{h}}\cdot p_{\vec{e}}}\Big)\cdot\Big(\sqrt{\frac{p_{\vec{h}}}{p_{\vec{e}}}}\ |\ p_{\vec{e}|\vec{h}}-p_{\vec{e}}\ |\ \Big)\ +\ \frac{1}{2}\Big(\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\notin\mathcal{E}} p_{\vec{h}}p_{\vec{e}|\vec{h}}+\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\notin\mathcal{E}} p_{\vec{h}}p_{\vec{e}}\Big)\\
\le\ & \frac{1}{2}\sqrt{\Big(\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\in\mathcal{E}} p_{\vec{h}}\cdot p_{\vec{e}}\Big)\cdot\Big(\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\in\mathcal{E}}\frac{p_{\vec{h}}}{p_{\vec{e}}}\cdot\big(p_{\vec{e}|\vec{h}}-p_{\vec{e}}\ \big)^2\Big)}\ +\ \Pr[\mathsf{Ber}_\mu^q\notin\mathcal{E}]\\
\le\ & \frac{1}{2}\sqrt{1\cdot\sum_{\vec{e}\in\mathcal{E}}\Big(\sum_{\vec{h}\in\mathcal{H}^q}\frac{p_{\vec{h}}p_{\vec{e}|\vec{h}}^2}{p_{\vec{e}}}\ -\ 2\sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}|\vec{h}}\ +\ \sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}}\Big)}\ +\ \exp^{-\frac{\Delta^2\mu q}{3}}\\
\le\ & \frac{1}{2}\sqrt{|\mathcal{E}|\cdot 2^{-m}}\ +\ \exp^{-\frac{\Delta^2\mu q}{3}}\\
\le\ & 2^{\frac{(1+\Delta)q\mathbf{H}(\mu)-m}{2}}\ +\ \exp^{-\frac{\Delta^2\mu q}{3}}\ ,
\end{aligned}$$

where the second inequality is Cauchy-Schwarz, i.e., $|\sum a_ib_i|\le\sqrt{(\sum a_i^2)\cdot(\sum b_i)^2}$ and (3) below, the third inequality follows from *Lemma* 3.1, and the fourth inequality is due to (4) and (5), i.e., fix any $\vec{e}$ (and thus fix $p_{\vec{e}}$ as well) we can substitute $p_{\vec{e}}\cdot(2^{-m}+p_{\vec{e}})$ for $\sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}|\vec{h}}^2$, and $p_{\vec{e}}$ for both $\sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}|\vec{h}}$ and $\sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}}$, and the last inequality follows from the definition of $\mathcal{E}$ (see (2))

$$|\mathcal{E}|\ \le\ 1/\min_{\vec{e}\in\mathcal{E}}\Pr[\mathsf{Ber}_\mu^q=\vec{e}]\ \le\ 2^{(1+\Delta)q\mathbf{H}(\mu)}$$

which completes the proof.

Claim 1

$$\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\notin\mathcal{E}}p_{\vec{h}}p_{\vec{e}|\vec{h}}\ =\ \sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\notin\mathcal{E}}p_{\vec{h}}p_{\vec{e}}\ =\ \Pr[\mathsf{Ber}_\mu^q\notin\mathcal{E}] \tag{3}$$

$$\forall\vec{e}\in\{0,1\}^q:\sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}|\vec{h}}^2\ \le\ p_{\vec{e}}\cdot(2^{-m}+p_{\vec{e}}) \tag{4}$$

$$\forall\vec{e}\in\{0,1\}^q:\sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}|\vec{h}}=\sum_{\vec{h}\in\mathcal{H}^q}p_{\vec{h}}p_{\vec{e}}=p_{\vec{e}} \tag{5}$$

Proof. Let $\vec{H}(W)\stackrel{\text{def}}{=}(H_1(W),\dots,H_q(W))$. The pairwise independence of $\mathcal{H}$ implies that

$$\vec{H}(W)\ \sim\ (U_v^1,\dots,U_v^q)$$

holds even conditioned on any fixing of $W=w$, and thus $\mathsf{sample}(\vec{H}(W))\sim\mathsf{Ber}_\mu^q$. We have

$$\sum_{\vec{h}\in\mathcal{H}^q,\vec{e}\notin\mathcal{E}}p_{\vec{h}}p_{\vec{e}|\vec{h}}=\Pr[\ \mathsf{sample}(\vec{H}(W))\notin\mathcal{E}\ ]=\Pr[\ \mathsf{Ber}_\mu^q\notin\mathcal{E}\ ],$$

$$\forall \vec{e} \in \{0,1\}^q : \sum_{\vec{h}\in\mathcal{H}^q} p_{\vec{h}} p_{\vec{e}|\vec{h}} = \Pr[\ \mathsf{sample}(\vec{H}(W)) = \vec{e}\ ] \ = \ \Pr[\ \mathsf{Ber}_\mu^q = \vec{e}\ ] = p_{\vec{e}},$$

$$\sum_{\vec{h}\in\mathcal{H}^q, \vec{e}\notin\mathcal{E}} p_{\vec{h}} p_{\vec{e}} \ = \ \sum_{\vec{h}\in\mathcal{H}^q} p_{\vec{h}} \cdot \sum_{\vec{e}\notin\mathcal{E}} p_{\vec{e}} \ = \ \Pr[\ \mathsf{Ber}_\mu^q \notin \mathcal{E}\ ],$$

$$\forall \vec{e} \in \{0,1\}^q : \sum_{\vec{h}\in\mathcal{H}^q} p_{\vec{h}} p_{\vec{e}} \ = \ p_{\vec{e}} \cdot \sum_{\vec{h}\in\mathcal{H}^q} p_{\vec{h}} \ = \ p_{\vec{e}}.$$

Now fix any $\vec{e} \in \{0,1\}^q$, and let W_1 and W_2 be random variables that are i.i.d. to W, we have

$$\begin{aligned}
\sum_{\vec{h}\in\mathcal{H}^q} p_{\vec{h}} p^2_{\vec{e}|\vec{h}} \ &= \ \Pr_{W_1,W_2,\vec{H}}[\ \mathsf{sample}(\vec{H}(W_1)) = \mathsf{sample}(\vec{H}(W_2)) = \vec{e}\ ] \\
&\le \ \Pr_{W_1,W_2}[W_1 = W_2] \cdot \Pr_{W_1,\vec{H}}[\ \mathsf{sample}(\vec{H}(W_1)) = \vec{e}\ ] \\
&\quad + \Pr_{\vec{H}}[\mathsf{sample}(\vec{H}(w_1)) = \mathsf{sample}(\vec{H}(w_2)) = \vec{e} \mid w_1 \neq w_2] \\
&\le \ 2^{-m} \cdot p_{\vec{e}} \ + \ \Pr[\mathsf{Ber}_\mu^q = \vec{e}]^2 \ = 2^{-m} \cdot p_{\vec{e}} \ + \ p^2_{\vec{e}},
\end{aligned}$$

where the second inequality is again due to the pairwise independence of $\mathcal{H}$, i.e., for any $w_1 \neq w_2$, $\vec{H}(w_1)$ and $\vec{H}(w_2)$ are i.i.d. to $(U_v^1, \ldots, U_v^q)$ and thus the two distributions $\mathsf{sample}(\vec{H}(w_1))$ and $\mathsf{sample}(\vec{H}(w_2))$ are i.i.d. to Ber_μ^q. □

□

4 Parallelizable PRFs on Weak Keys

4.1 A Succinct Formulation of LPN

The authors of [22] observed that the secret of LPN is not necessary to be uniformly random and can be replaced with a Bernoulli distribution. We state a more quantitative version (than [22, Problem 2]) in Lemma 4.1 that Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ (see Definition 2.2) is implied by standard LPN for nearly the same parameters except that standard LPN needs n more samples. The proof follows by a simple reduction and is included in Appendix Appendix B.

Lemma 4.1 *Assume that the decisional (resp., computational)* $\mathsf{LPN}_{\mu,n}$ *problem is* (q, t, ϵ)*-hard, then the decisional (resp., computational)* Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ *problem is at least* $(q-(n+2),\ t-\mathsf{poly}(n+q),\ 2\epsilon)$*-hard.*

Remark 4.1 (On the security of low-noise LPN) *For $\mu = n^{-c}$, a trivial statistical test suggests (by the piling-up lemma) that any single sample of decisional* Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ *is* $(1/2+2^{-O(n^{1-c})})$*-biased to 0. In other words, decisional* Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ *is no more than* $(q = 1,\ t = O(1),\ \epsilon = 2^{-O(n^{1-c})})$*-hard and thus it follows (via the reduction of Lemma 4.1) that decisional* $\mathsf{LPN}_{\mu,n}$ *cannot have indistinguishability beyond* $(q = n+3, t = \mathsf{poly}(n), \epsilon = 2^{-O(n^{1-c})})$*. Asymptotically, this is also the current state-of-the-art attack on low-noise LPN using* $q = \mathsf{poly}(n)$ *or even more samples.*

4.2 A Direct Construction in Almost Constant Depth

To build a randomized PRG (on weak source w) from the succinct LPN, we first sample Bernoulli vector (s, e) from w (using random coins a), and then output $a{\cdot}s \oplus e$. Theorem 4.1 states that the above yields a randomized PRG on weak seed w and public coin a.

Theorem 4.1 (randomized PRGs from LPN) *Let n be a security parameter, let $\delta > 0$ be any constant, and let $\mu = n^{-c}$ for any $0 < c < 1$. Assume that decisional $\mathsf{LPN}_{\mu,n}$ problem is $((1+2\delta)n,\ t,\ \epsilon)$-hard, then $\mathcal{G} = \{G_a : \{0,1\}^{n^{1-\frac{c}{2}}} \to \{0,1\}^{\delta n}, a \in \{0,1\}^{\delta n \times n}\}_{n\in\mathbb{N}}$, where*

$$G_a(w) = a \cdot s \oplus e, s \in \{0,1\}^n, e \in \{0,1\}^{\delta n}$$

and $(s,e) = \mathsf{sample}(\vec{h_a}(w))$, is a $(t - \mathsf{poly}(n),\ O(\epsilon))$-randomized PRG on $(n^{1-\frac{c}{2}},\ 4c(1+\delta^2)n^{1-c}\cdot\log n)$-weak seed with stretch factor $\delta{\cdot}n^{\frac{c}{2}}$.

Proof. We have by Lemma 4.1 that $((1+2\delta)n, t, \epsilon)$-hard decisional $\mathsf{LPN}_{\mu,n}$ implies $(\delta n, t - \mathsf{poly}(n), 2\epsilon)$-hard decisional $\mathsf{Ber}_\mu^{n+\delta n}$-$\mathsf{LPN}_{\mu,n}$, so the conclusion follows if we could sample $(s,e) \xleftarrow{\$} \mathsf{Ber}_\mu^{n+\delta n}$ from w. This follows from Lemma 3.2 by choosing $q = n + \delta n$, $\Delta = \delta$, and $m = 4c(1+\delta)^2 n^{1-c}\cdot\log n$ such that the sampled noise vector is statistically close to $\mathsf{Ber}_\mu^{n+\delta n}$ except for an error bounded by

$$\begin{aligned} & 2^{\left((1+\Delta)q\mathbf{H}(\mu)-m\right)/2} + \exp^{-\frac{\Delta^2\mu q}{3}} \\ \leq\ & 2^{\left((1+\delta)^2 n\mathbf{H}(\mu)-2(1+\delta)^2 n\mathbf{H}(\mu)\right)/2} + 2^{-\Omega(n^{1-c})} \\ =\ & 2^{-\Omega(n^{1-c}\cdot\log n)} + 2^{-\Omega(n^{1-c})} \\ =\ & 2^{-\Omega(n^{1-c})} \end{aligned}$$

where recall by Fact A.1 that $\mu\log(1/\mu) < \mathbf{H}(\mu) < \mu(\log(1/\mu)+2)$ and thus $m > 2(1+\delta^2)n^{1-c}(c\log n + 2) > 2(1+\delta^2)n\mathbf{H}(\mu)$. We omit the above term since $\epsilon = 2^{-O(n^{1-c})}$ (see Remark 4.1). □

We state a variant of the theorem by Goldreich, Goldwasser and Micali [26] on building PRFs from PRGs, where we consider PRGs with stretch factor 2^v for $v = O(\log n)$ (i.e., a balanced 2^v-ary tree) and use randomized (instead of deterministic) PRG G_a, reusing public coin a at every invocation of G_a.

Theorem 4.2 (PRFs from PRGs [26]) *Let n be a security parameter, let $v = O(\log n)$, $\lambda \leq m = n^{O(1)}$, $\lambda = \mathsf{poly}(n)$, $t = t(n)$ and $\epsilon = \epsilon(n)$. Let $\mathcal{G} = \{G_a : \{0,1\}^m \to \{0,1\}^{2^v\cdot m}, a \in \mathcal{A}\}_{n\in\mathbb{N}}$ be a (t,ϵ) randomized PRG (with stretch factor 2^v) on (m,λ)-weak seed. Parse $G_a(k)$ as 2^v blocks of m-bit strings:*

$$G_a(k) \stackrel{def}{=} G_a^{0\cdots00}(k)\|G_a^{0\cdots01}(k)\|\cdots\|G_a^{1\cdots11}(k)$$

where $G_a^{i_1\cdots i_v}(k)$ denotes the $(i_1\cdots i_v)$-th m-bit block of $G_a(k)$. Then, for any $d \leq \mathsf{poly}(n)$ and $q = q(n)$, the function family ensemble $\mathcal{F} = \{F_{k,a} : \{0,1\}^{dv} \to \{0,1\}^{2^v\cdot m}, k \in \{0,1\}^m, a \in \mathcal{A}\}_{n\in\mathbb{N}}$, where

$$F_{k,a}(x_1\cdots x_{dv}) \stackrel{def}{=} G_a(\ G_a^{x_{(d-1)v+1}\cdots x_{dv}}(\cdots G_a^{x_{v+1}\cdots x_{2v}}(G_a^{x_1\cdots x_v}(k))\cdots)\ ),$$

is a $(q,\ t - q\cdot\mathsf{poly}(n),\ dq\epsilon)$ randomized PRF on (m,λ)-weak key.

On polynomial-size circuits. The above GGM tree has $\Theta(2^{dv})$ nodes and thus it may seem that for $dv = \omega(\log n)$ we need a circuit of super-polynomial size to evaluate $F_{k,p}$. This is not necessary since we can represent the PRF in the following alternative form:

$$F_{k,a} = G_a \ \circ\ \underbrace{\mathsf{mux}_{x_{(d-1)v+1}\cdots x_{dv}} \circ G_a}_{G_a^{x_{(d-1)v+1}\cdots x_{dv}}} \circ \cdots \circ \underbrace{\mathsf{mux}_{x_{v+1}\cdots x_{2v}} \circ G_a}_{G_a^{x_{v+1}\cdots x_{2v}}} \circ \underbrace{\mathsf{mux}_{x_1\cdots x_v} \circ G_a}_{G_a^{x_1\cdots x_v}}$$

where '$\circ$' denotes function composition, each multiplexer $\mathsf{mux}_{i_1\cdots i_v} : \{0,1\}^{2^v m} \to \{0,1\}^m$ simply selects as output the $(i_1\cdots i_v)$-th m-bit block of its input, and it can be implemented with $O(2^v\cdot m) = \mathsf{poly}(n)$ NOT and (unbounded fan-in) AND/OR gates of constant depth. Thus, for $v = O(\log n)$ function $F_{k,p}$ can be evaluated with a polynomial-size circuit of depth $O(d)$.

Lemma 4.2 (Levin's trick [41]) *For any $\ell \le n \in \mathbb{N}$, let R_1 be a random function distribution over $\{0,1\}^\ell \to \{0,1\}^n$, let $\mathcal{H}$ be a family of universal hash functions from n bits to ℓ bits, and let H_1 be a function distribution uniform over $\mathcal{H}$. Let $R_1 \circ H_1(x) \stackrel{def}{=} R_1(H_1(x))$ be a function distribution over $\{0,1\}^n \to \{0,1\}^n$. Then, for any $q \in \mathbb{N}$ and any oracle aided D bounded by q queries, we have*

$$\left| \Pr_{R_1,H_1}[\ \mathsf{D}^{R_1\circ H_1} = 1\ ] \ - \ \Pr_{R}[\ \mathsf{D}^{R} = 1\ ] \right| \ \le \ \frac{q^2}{2^{\ell+1}},$$

where R is a random function distribution from n bits to n bits.

Theorem 4.3 (A direct PRF) *Let n be a security parameter, and let $\mu = n^{-c}$ for constant $0 < c < 1$. Assume that decisional $\mathsf{LPN}_{\mu,n}$ problem is $(\alpha n, t, \epsilon)$-hard for any constant $\alpha > 1$, then for any (efficiently computable) $d = \omega(1) \le O(n)$ and any $q \le n^{d/3}$ there exists a $(q, t - q\,\mathsf{poly}(n), O(dq\epsilon) + q^2 n^{-d})$-randomized PRF on $(n^{1-\frac{c}{2}},\ O(n^{1-c}\log n))$[8]-weak key*

$$\bar{\mathcal{F}} = \{\bar{F}_{k,a} : \{0,1\}^n \to \{0,1\}^n, k \in \{0,1\}^{n^{1-\frac{c}{2}}}, a \in \{0,1\}^{O(n^2)}\}_{n\in\mathbb{N}} \tag{6}$$

which is computable by a uniform family of polynomial-size depth-$O(d)$ circuits with unbounded-fan-in AND, OR and XOR gates.

Proof. For $\mu = n^{-c}$, we have by Theorem 4.1 that the decisional $(\alpha n, t, \epsilon)$-hard $\mathsf{LPN}_{\mu,n}$ implies a $(t - \mathsf{poly}(n), O(\epsilon))$ randomized PRG in $\mathrm{AC}^0(\mathrm{MOD}_2)$ on $(n^{1-\frac{c}{2}},\ O\ (n^{1-c}\log n)\ )$-weak seed k and public coin $a \in \{0,1\}^{O(n^2)}$ with stretch factor $2^v = n^{\frac{c}{2}}$. We plug it into the GGM construction (see Theorem 4.2) with tree depth $d' = 2d/c$ to get a $(q, t - q\,\mathsf{poly}(n), O(dq\epsilon))$ randomized PRF on weak keys (of same parameters) with input length $d'v = d\log n$ and output length $2^v \cdot n^{1-\frac{c}{2}} = n$ as below:

$$\mathcal{F} = \{F_{k,a} : \{0,1\}^{d\log n} \to \{0,1\}^n, k \in \{0,1\}^{n^{1-\frac{c}{2}}}, a \in \{0,1\}^{O(n^2)}\}_{n\in\mathbb{N}}. \tag{7}$$

Now we expand k (e.g., by evaluating $F_{k,a}$ on a few fixed points) into a pseudorandom $(\bar{k}, \bar{h}_1)$, where $\bar{k} \in \{0,1\}^{n^{1-\frac{c}{2}}}$ and $\bar{h}_1$ describes a universal hash function from n bits to $\ell = d\log n$ bits. Motivated by Levin's trick, we define a domain-extended PRF $\bar{F}_{k,a}(x) \stackrel{\text{def}}{=} F_{\bar{k},a} \circ \bar{h}_1(x)$. For any oracle-aided distinguisher D running in time $t - q\mathsf{poly}(n)$ and making q queries, denote with $\delta_{\mathsf{D}}(F_1, F_2) \stackrel{\text{def}}{=} \left|\Pr[\ \mathsf{D}^{F_1}(A) = 1\ ] - \Pr[\ \mathsf{D}^{F_2}(A) = 1\ ]\right|$ the advantage of D (who gets public coin A as additional input) in distinguishing between function oracles F_1 and F_2. Therefore, we have by a triangle inequality

$$\begin{aligned}\delta_{\mathsf{D}}(F_{\bar{K},A} \circ \bar{H}_1, R) \ &\le\ \delta_{\mathsf{D}}(F_{\bar{K},A} \circ \bar{H}_1, F_{K,A} \circ H_1) + \delta_{\mathsf{D}}(F_{K,A} \circ H_1, R_1 \circ H_1) \ + \ \delta_{\mathsf{D}}(R_1 \circ H_1,\ R) \\ &\le\ O(dq\epsilon) + q^2 n^{-d},\end{aligned}$$

where advantage is upper bounded by three terms, namely, the indistinguishability between $(\bar{K}, \bar{H}_1)$ and truly random (K, H_1), that between $F_{K,A}$ and random function R_1 (of the same input/output lengths as $F_{K,A}$), and that due to Lemma 4.2. Note that A is independent of R_1, H_1 and R. □

4.3 Going Beyond the Birthday Barrier

Unfortunately, for small $d = \omega(1)$ the security of the above PRF does not go beyond super-polynomial (cf. term $q^2 n^{-d}$) due to a birthday attack. This situation could be handled using security-preserving constructions. Note the techniques from [34, 17] need (almost) $\Omega(d\log n)$-wise independent hash functions which we don't know how to compute with unbounded fan-in gates of depth $O(d)$. Thus, we use a more intuitive and depth-preserving approach below by simply running a few independent copies and

[8] Here the big-Oh omits a constant dependent on c and α.

XORing their outputs. The essential idea dates backs to [8, 44] and the technique receives renewed interest recently in some different contexts [23, 25]. We mention that an alternative (and possibly more efficient) approach is to use the second security-preserving domain extension technique from [10] that requires a few pairwise independent hash functions and makes only a constant number of calls to the underlying small-domain PRFs. This yields the PRF stated in Theorem 4.4.

Lemma 4.3 (Generalized Levin's Trick [8, 44]) *For any $\kappa, \ell \leq n \in \mathbb{N}$, let $R_1, \ldots, R_\kappa$ be independent random function distributions over $\{0,1\}^\ell \to \{0,1\}^n$, let $\mathcal{H}$ be a family of universal hash functions from n bits to ℓ bits, and let $H_1, \cdots, H_\kappa$ be independent function distributions all uniform over $\mathcal{H}$. Let $F_{\vec{R},\vec{H}}$ be a function distribution (induced by $\vec{R} = (R_1, \ldots, R_\kappa)$ and $\vec{H} = (H_1, \ldots, H_\kappa)$) over $\{0,1\}^n \to \{0,1\}^n$ defined as*

$$F_{\vec{R},\vec{H}}(x) \stackrel{def}{=} \bigoplus_{i=1}^{\kappa} R_i(H_i(x)). \tag{8}$$

Then, for any $q \in \mathbb{N}$ and any oracle aided D bounded by q queries, we have

$$\left|\Pr[\ \mathsf{D}^{F_{\vec{R},\vec{H}}} = 1\ ]\ -\ \Pr[\ \mathsf{D}^{R} = 1\ ]\right|\ \leq\ \frac{q^{\kappa+1}}{2^{\kappa\ell}}$$

where R is a random function distribution over $\{0,1\}^n \to \{0,1\}^n$.

Finally, we get the first security-preserving construction below. To have comparable security to LPN with secret size n, it suffices to use a key of entropy $O(n^{1-c} \cdot \log n)$, or a uniform key of size $n^{1-\frac{c}{2}}$ with any $(1 - O(n^{-\frac{c}{2}} \log n))$-fraction of leakage (see Fact A.7), provided that leakage is independent of public coin a.

Theorem 4.4 (A security-preserving PRF on weak key) *Let n be a security parameter, and let $\mu = n^{-c}$ for constant $0 < c < 1$. Assume that the decisional $\mathsf{LPN}_{\mu,n}$ problem is $(\alpha n, t, \epsilon)$-hard for any constant $\alpha > 1$, then for any (efficiently computable) $d = \omega(1) \leq O(n)$ and any $q\ \leq\ n^{d/3}$ there exists a $(q,\ t - q\mathsf{poly}(n),\ O(dq\epsilon))$- randomized PRF on $(n^{1-\frac{c}{2}}, O(n^{1-c} \cdot \log n))$-weak key*

$$\hat{\mathcal{F}} = \{\hat{F}_{k,a} : \{0,1\}^n \to \{0,1\}^n, k \in \{0,1\}^{n^{1-\frac{c}{2}}}, a \in \{0,1\}^{O(n^2)}\}_{n\in\mathbb{N}}$$

which are computable by a uniform family of polynomial-size depth-$O(d)$ circuits with unbounded-fan-in AND, OR and XOR gates.

Proof sketch. Following the proof of Theorem 4.3, we get a $(q,\ t - q\mathsf{poly}(n),\ O(dq\epsilon)\ )$- randomized PRF $\mathcal{F} = \{F_{k,a}\}_{n\in\mathbb{N}}$ on weak keys (see (7)) with input length $d \log n$ and of depth $O(d)$. We define $\mathcal{F}' = \{F'_{(\vec{k},\vec{h}),a} : \{0,1\}^n \to \{0,1\}^n, \vec{k} \in \{0,1\}^{O(\kappa n^{1-\frac{c}{2}})}, \vec{h} \in \mathcal{H}^\kappa, a \in \{0,1\}^{O(n^2)}\}_{n\in\mathbb{N}}$ where

$$F'_{(\vec{k},\vec{h}),a}(x)\ \stackrel{\text{def}}{=}\ \bigoplus_{i=1}^{\kappa} F_{k_i,a}(h_i(x)),\ \vec{k} = (k_1, \cdots, k_\kappa),\ \vec{h} = (h_1, \cdots, h_\kappa)\ .$$

Let $\delta_{\mathsf{D}}(F_1, F_2) \stackrel{\text{def}}{=} \left|\ \Pr[\ \mathsf{D}^{F_1}(A) = 1\ ] - \ \Pr[\ \mathsf{D}^{F_2}(A) = 1\ ]\right|$. We have that for any oracle-aided distinguisher running in time $t - q\mathsf{poly}(n)$ and making up to q queries, we have by a triangle inequality that

$$\begin{aligned}
\delta_{\mathsf{D}}(\ {F'}_{(\vec{K},\vec{H}),A},\ R\ ) \quad &\leq \quad \delta_{\mathsf{D}}(\ {F'}_{(\vec{K},\vec{H}),A},\ F_{\vec{R},\vec{H}}\ )\ +\ \delta_{\mathsf{D}}(\ F_{\vec{R},\vec{H}},\ R\ ) \\
&\leq \quad O(\kappa d q\epsilon)\ +\ n^{d(1-2\kappa)/3} \\
&= \quad O(\kappa d q\epsilon)\ +\ 2^{-\omega(n^{1-c})}\ =\ O(\kappa d q\epsilon)\ ,
\end{aligned}$$

where $F_{\vec{R},\vec{H}}$ is defined as per (8), the first term of the second inequality is due to a hybrid argument (replacing every $F_{K_i,A}$ with R_i one at a time), the second term of the second inequality follows from Lemma 4.3 with $\ell = d\log n$ and $q \le n^{d/3}$, and the equalities follow by setting $\kappa = n^{1-c}$ to make the first term dominant. Therefore, $F'_{(\vec{k},\vec{h}),a}$ is almost the PRF as desired except that it uses a long key $(\vec{k},\vec{h})$, which can be replaced with a pseudorandom one. That is, let $\hat{F}_{k,a}(x) \stackrel{\text{def}}{=} F'_{(\vec{k},\vec{h}),a}(x)$ and $(\vec{k},\vec{h}) \stackrel{\text{def}}{=} F_{k,a}(1) \parallel F_{k,a}(2) \parallel \cdots \parallel F_{k,a}(O(\kappa))$, which adds only a layer of gates of depth $O(d)$. □

5 An Alternative PRF with a Short Uniform Key

In this section, we introduce an alternative construction based on a variant of LPN (reducible from standard LPN) whose noise vector can be sampled in AC^0 (i.e., without using XOR gates). We state the end results in Theorem 5.1 that standard LPN with n-bit secret implies a low-depth PRF with key size $\Theta(n^{1-c}\log n)$. Concretely (and ideally), assume that computational LPN is $(q = 1.001n, t = 2^{n^{1-c}/3}, \epsilon = 2^{-n^{1-c}/12})$-hard, and let $\lambda = \Theta(n^{1-c}\log n)$, then for any $\omega(1) = d = O(\lambda/\log^2\lambda)$ there exists a parallelizable $(q' = \lambda^{\Theta(d)}, t' = 2^{\Theta(\lambda/\log\lambda)}, \epsilon' = 2^{-\Theta(\lambda/\log\lambda)})$)-randomized PRF computable in depth $O(d)$ with secret key length λ and public coin length $O(\lambda^{\frac{1+c}{1-c}})$.

5.1 Main Results and Roadmap

Theorem 5.1 (A PRF with a compact uniform key) *Let n be a security parameter, and let $\mu = n^{-c}$ for constant $0 < c < 1$. Assume that the computational $\mathsf{LPN}_{\mu,n}$ problem is $(\alpha n, t, \epsilon)$-hard for any constant $\alpha > 1$ and efficiently computable ϵ, then for any (efficiently computable) $d = \omega(1) \le O(n)$ and any $q' \le n^{d/3}$ there exists a $(q', \Theta(t\cdot\epsilon^2 n^{1-2c}), O(dq'n^2\epsilon))$- randomized PRF on uniform key*

$$\tilde{\mathcal{F}} = \{\tilde{F}_{k,a} : \{0,1\}^n \to \{0,1\}^n, k \in \{0,1\}^{\Theta(n^{1-c}\cdot\log n)}, a \in \{0,1\}^{O(n^2)}\}_{n\in\mathbb{N}}$$

which are computable by a uniform family of polynomial-size depth-$O(d)$ circuits with unbounded-fan-in AND, OR and XOR gates.

We sketch the steps below to prove Theorem 5.1, where 'C-' and 'D-' stand for 'computational' and 'decisional' respectively.

1. Introduce distribution ψ_μ^m that can be sampled in AC^0.
2. $((1+\Theta(1))n,t,\epsilon)$-hard C- $\mathsf{LPN}_{\mu,n} \implies (\Theta(n), t-\mathsf{poly}(n), 2\epsilon)$-hard C- Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ (by Lemma 4.1).
3. $(\Theta(n), t, \epsilon)$-hard C- Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n} \implies (\Theta(n), t-\mathsf{poly}(n), O(n^{3/2-c}\epsilon))$-hard C- ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ (by Lemma 5.4).
4. $(\Theta(n), t, \epsilon)$-hard C- ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n} \implies (\Theta(n), \Omega(t(\epsilon/n)^2), 2\epsilon)$-hard D- ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ (by Theorem 5.2).
5. $(\Theta(n), t, \epsilon)$-hard D- ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n} \implies (q, t-q\,\mathsf{poly}(n), O(dq'\epsilon))$- randomized PRF for any $d = \omega(1)$ and $q' \le n^{d/3}$, where the PRF has key length $\Theta(n^{1-c}\log n)$ and can be computed by polynomial-size depth-$O(d)$ circuits with unbounded-fan-in AND, OR and XOR gates. This is stated as Theorem 5.3.

Algorithm 1 Sampling distribution ψ_μ^m in AC^0

Require: $2\mu m \log m$ random bits (assume WLOG that m is a power of 2)
Ensure: ψ_μ^m satisfies Lemma 5.1

1: Sample random $z_1, \ldots, z_{2\mu m}$ of Hamming weight 1, i.e., for every $i \in [m]$ $z_i \xleftarrow{\$} \{z \in \{0,1\}^m : |z| = 1\}$. {E.g., to sample z_1 with randomness $r_1 \ldots r_{\log m}$, simply let each $(b_1 \ldots b_{\log m})$-th bit of z_1 to be $r_1^{b_1} \wedge \cdots \wedge r_{\log m}^{b_{\log m}}$, where $r_j^{b_j} \stackrel{\text{def}}{=} r_j$ for $b_j = 0$ and $r_j^{b_j} \stackrel{\text{def}}{=} \neg r_j$ otherwise. Note that AC^0 allows NOT gates at the input level. }
2: Output the bitwise-OR of the vectors $z_1, \ldots, z_{2\mu m}$.
{Note: we take a bitwise-OR (***not bitwise-XOR***) of the vectors.}

5.2 Distribution ψ_μ^m and the ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ Problem

We introduce a distribution ψ_μ^m that can be sampled in AC^0 and show that ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ is implied by Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ (and thus by standard LPN). Further, for $\mu = n^{-c}$ sampling ψ_μ^m needs $\Theta(mn^{-c}\log n)$ random bits, which asymptotically match the Shannon entropy of Ber_μ^m.

Lemma 5.1 *The distribution ψ_μ^m (sampled as per Algorithm 1) is $2^{-\Omega(\mu m \log(1/\mu))}$-close to a convex combination of $\chi_{\mu m}^m$, $\chi_{\mu m+1}^m$, $\ldots$, $\chi_{2\mu m}^m$.*

Proof. It is easy to see that ψ_μ^m is a convex combination of χ_1^m, χ_2^m, $\ldots, \chi_{2\mu m}^m$ as conditioned on $|\psi_\mu^m| = i$ (for any i) ψ_μ^m hits every $y \in \{0,1\}^m$ of Hamming weight $|y| = i$ with equal probability. Hence, it remains to show that those χ_j^m's with Hamming weight $j < \mu m$ sum to a fraction less than $2^{-\mu m(\log(1/\mu)-2)}$, i.e.,

$$\begin{aligned}\Pr[|\psi_\mu^m| < \mu m] &= \sum_{y \in \{0,1\}^m : |y| < \mu m} \Pr[\psi_\mu^m = y] \\ &< \mu^{2\mu m} \cdot 2^{m\mathbf{H}(\mu) - \frac{\log m}{2} + O(1)} \\ &< \mu^{2\mu m} \cdot 2^{\mu m(\log(1/\mu)+2)+O(1)} = 2^{\mu m(-\log(1/\mu)+2)+O(1)}\end{aligned}$$

where the first inequality is due to the partial sum of binomial coefficients (see Fact A.5) and that for any fixed y with $|y| < \mu m$ $\psi_\mu^m = y$ happens only if the bit 1 of every z_i (see Algorithm 1) hits the 1's of y (each with probability less than μ independently) and the second inequality is Fact A.1. □

By definition of ψ_μ^{n+q} the sampled (s, e) has Hamming weight no greater than $2\mu(n+q)$ and the following lemma states that ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ is almost injective.

Lemma 5.2 **(ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ is almost injective)** *For $q = \Omega(n)$, define set $\mathcal{Y} \stackrel{\text{def}}{=} \{(s,e) \in \{0,1\}^{n+q} : |(s,e)| \le (n+q)/\log n\}$. Then, for every $(s,e) \in \mathcal{Y}$,*

$$\Pr_{a \leftarrow U_{qn}} \left[\ \exists (s', e') \in \mathcal{Y} : (s', e') \neq (s, e) \wedge\ as \oplus e = as' \oplus e'\ \right] = 2^{-\Omega(q)} .$$

Proof. Let $\mathcal{H} \stackrel{\text{def}}{=} \{h_a : \{0,1\}^{n+q} \to \{0,1\}^q, a \in \{0,1\}^{qn}, h_a(s,e) \stackrel{\text{def}}{=} as \oplus e\}$ and it is not hard to see that $\mathcal{H}$ is a family of universal hash functions. We have

$$\log|\mathcal{Y}| = \log \sum_{i=0}^{(n+q)/\log n} \binom{n+q}{i} = O\big((n+q)\log\log n/\log n\big) = o(q) ,$$

where the approximation is due to Fact A.5 and the conclusion immediately follows from Lemma 5.3 (with proof reproduced in Appendix B). □

Lemma 5.3 (The injective hash lemma (e.g. [55])) *For any integers $l_1 \le l_2, m$, let $\mathcal{Y}$ be any set of size $|\mathcal{Y}| \le 2^{l_1}$, and let $\mathcal{H} \stackrel{\mathsf{def}}{=} \{h_a : \{0,1\}^m \to \{0,1\}^{l_2}, a \in \mathcal{A}, \mathcal{Y} \subseteq \{0,1\}^m\}$ be a family of universal hash functions. Then, for every $y \in \mathcal{Y}$ we have*

$$\Pr_{a \xleftarrow{\$} \mathcal{A}} [\ \exists y' \in \mathcal{Y}:\ y' \neq y \wedge h_a(y') = h_a(y)\ ] \le 2^{l_1 - l_2}\ .$$

5.3 Computational Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ → Computational ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$

Lemma 5.4 non-trivially extends the well-known fact that the computational LPN implies the computational exact LPN, i.e., $(U_n, \chi_{\mu q}^q)$-$\mathsf{LPN}_{\mu,n}$.

Lemma 5.4 *Let $q = \Omega(n)$, $\mu = n^{-c}$ $(0 < c < 1)$ and $\epsilon = 2^{-O(n^{1-c})}$. Assume that the computational Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ problem is (q, t, ϵ)-hard, then the computational ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ problem is $(q, t - \mathsf{poly}(n+q), O(\mu(n+q)^{3/2}\epsilon))$-hard.*

Proof. Let $m = n + q$ and write $\mathsf{Adv}_\mathsf{D}(X) \stackrel{\mathsf{def}}{=} \Pr_{a \xleftarrow{\$} U_{qn}, (s,e) \leftarrow X}[\ \mathsf{D}(a, a{\cdot}s \oplus e) = (s, e)\ ]$. Towards a contradiction we assume that there exists D such that $\mathsf{Adv}_\mathsf{D}(\psi_\mu^m) > \epsilon'$, and we assume WLOG that on input (a, z) D always outputs (s', e') with $|(s', e')| \le 2\mu m$. That is, even if it fails to find any (s', e') satisfying $as' \oplus e' = z$ and $|(s', e')| \le 2\mu m$ it just outputs a zero vector. Lemma 5.1 states that ψ_μ^m is $2^{-\Omega(\mu n(\log(1/\mu))}$-close to a convex combination of $\chi_{\mu m}^m$, $\chi_{\mu m+1}^m$, $\ldots$, $\chi_{2\mu m}^m$, and thus there exists $j \in \{\mu m, \mu m + 1, \ldots, 2\mu m\}$ such that $\mathsf{Adv}_\mathsf{D}(\chi_j^m) > \epsilon' - 2^{-\Omega(n^{1-c}\log n)} > \epsilon'/2$, which further implies that $\mathsf{Adv}_\mathsf{D}(\mathsf{Ber}_{j/m}^m) = \Omega(\epsilon'/\sqrt{m})$ as $\mathsf{Ber}_{j/m}^m$ is a convex combination of χ_0^m, $\ldots$, χ_m^m, of which it hits χ_j^m with probability $\Omega(1/\sqrt{m})$ by Lemma 5.5. Next, we define D' as in Algorithm 2.

Algorithm 2 a Ber_μ^m-$\mathsf{LPN}_{\mu,n}$ solver D'

Require: a random Ber_μ^m-$\mathsf{LPN}_{\mu,n}$ instance $(a, z = a{\cdot}s \oplus e)$ as input
Ensure: a good chance to find out (s, e)

1: Sample $j^* \xleftarrow{\$} \{\mu m, \mu m + 1, \ldots, 2\mu m\}$ as a guess about j.
2: Compute $\mu' = j^*/m$.
3: $(s_1, e_1) \leftarrow \mathsf{Ber}^m_{\frac{\mu'-\mu}{1-2\mu}}$. {This makes $(a, z \oplus (as_1 \oplus e_1))$ a random $\mathsf{Ber}_{\mu'}^m$-$\mathsf{LPN}_{\mu',n}$ sample by the piling-up lemma (see Fact A.6)}
4: $(s', e') \leftarrow \mathsf{D}(\ a,\ z \oplus (as_1 \oplus e_1)\ )$.
5: Output $(s' \oplus s_1, e' \oplus e_1)$. {$\mathsf{D}'$ succeeds iff $(s' \oplus s_1, e' \oplus e_1) = (s, e)$}

We denote $\mathcal{E}_{suc}$ the event that D succeeds in finding (s', e') such that $as' \oplus e' = z \oplus (as_1 \oplus e_1)$ and thus we have $a(s' \oplus s_1) \oplus (e' \oplus e_1) = z = as \oplus e$, where values are sampled as defined above. This however does not immediately imply $(s, e) = (s' \oplus s_1, e' \oplus e_1)$ unless conditioned on the event $\mathcal{E}_{inj}$ that $h_a(s, e) \stackrel{\mathsf{def}}{=} a{\cdot}s \oplus e$ is injective on input (s, e).

$$\begin{aligned}
&\Pr_{a \leftarrow U_{qn},\ (s,e) \leftarrow \mathsf{Ber}_\mu^m,\ (s_1,e_1) \leftarrow \mathsf{Ber}^m_{\frac{\mu'-\mu}{1-2\mu}},\ s' \leftarrow \mathsf{D}(a, y \oplus (as_1 \oplus e_1))}[\ (s' \oplus s_1, e' \oplus e_1) = (s, e)\ ] \\
&\geq \Pr[\mathcal{E}_{suc} \wedge \mathcal{E}_{inj}] \\
&\geq \Pr[\mathcal{E}_{suc}] - \Pr[\neg\mathcal{E}_{inj}] \\
&\geq \Pr[j^* = j] \cdot \mathsf{Adv}_\mathsf{D}(\mathsf{Ber}_{j/m}^m) - 2^{-\Omega(m/\log^2 n)} \\
&= \Omega(\epsilon'/\mu m^{3/2}),
\end{aligned}$$

where the bound on event $\neg\mathcal{E}_{inj}$ is given below. We reach a contradiction by setting $\varepsilon' = \Omega(1)\cdot\mu m^{3/2}\epsilon$ for a large enough $\Omega(1)$ so that D' solves Ber_μ^m-$\mathsf{LPN}_{\mu,n}$ with probability greater than ϵ.

$$\begin{aligned}
& \Pr[\neg\mathcal{E}_{inj}] \\
\le\ & \Pr[\neg\mathcal{E}_{inj} \wedge (s,e)\in\mathcal{Y} \wedge (s'\oplus s_1, e'\oplus e_1)\in\mathcal{Y}] \\
& \qquad + \quad \Pr[(s,e)\notin\mathcal{Y} \vee (s'\oplus s_1, e'\oplus e_1)\notin\mathcal{Y}] \\
\le\ & 2^{-\Omega(m)} + \Pr[(s,e)\notin\mathcal{Y}] + \Pr[(s'\oplus s_1, e'\oplus e_1)\notin\mathcal{Y}\ ] \\
\le\ & 2^{-\Omega(m)} + \Pr_{(s,e)\leftarrow\mathsf{Ber}_\mu^m}[\ |(s,e)| \ge m/\log n\ ] + \Pr_{(s_1,e_1)\leftarrow\mathsf{Ber}^m_{\frac{\mu'-\mu}{1-2\mu}}}[\ |(s_1,e_1)| \ge (\frac{1}{\log n} - 2\mu)m\ ] \\
=\ & 2^{-\Omega(m/\log^2 n)},
\end{aligned}$$

where $\mathcal{Y} \stackrel{\text{def}}{=} \{(s,e)\in\{0,1\}^m : |(s,e)| < m/\log n\}$, the second inequality is from Lemma 5.2, the third inequality is that $|(u\oplus w)|\ge\kappa$ implies $|w|\ge\kappa-|u|$ and by definition of D string (s',e') has Hamming weight no greater than $2\mu m$, and the last inequality is a typical Chernoff-Hoeffding bound. □

Lemma 5.5 *For $0<\mu'<1/2$ and $m\in\mathbb{N}$, we have that*

$$\Pr\left[\ |\mathsf{Ber}_{\mu'}^m| \ = \ \lceil\mu' m\rceil\ \right] = \Omega(1/\sqrt{m}).$$

5.4 C- ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ → D- ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ → $\omega(1)$-depth PRFs

Next we show that the computational ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ problem implies its decisional counterpart. The theorem below is implicit in [5][9] and the case for ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ falls into a special case. Note that ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ is almost injective by Lemma 5.2, and thus its computational and decisional versions are equivalent in a sample-preserving manner. In fact, Theorem 5.2 holds even without the injective condition, albeit with looser bounds.

Theorem 5.2 (Sample preserving reduction [5]) *If the computational X-$\mathsf{LPN}_{\mu,n}$ is (q,t,ϵ)-hard for any efficiently computable ϵ, and it satisfies the injective condition, i.e., for any $(s,e)\in\mathit{Supp}(X)$ it holds that*

$$\Pr_{a\leftarrow U_{qn}}[\ \exists(s',e')\in \mathit{Supp}(X):\ (s',e')\neq(s,e)\wedge a\cdot s\oplus e = a\cdot s'\oplus e'\ ] \le 2^{-\Omega(n)}.$$

Then, the decisional X-$\mathsf{LPN}_{\mu,n}$ is $(q,\Omega(t(\epsilon/n)^2),2\epsilon)$-hard.

Theorem 5.3 (Decisional ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ → PRF) *Let n be a security parameter, and let $\mu = n^{-c}$ for any constant $0<c<1$. Assume that the decisional ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ problem is $(\delta n,t,\epsilon)$-hard for any constant $\delta>0$, then for any (efficiently computable) $d=\omega(1)\le O(n)$ and any $q' \le n^{d/3}$ there exists a $(q', t-q'\mathsf{poly}(n), O(dq'\epsilon))$- randomized PRF (on uniform key) with key length $\Theta(n^{1-c}\log n)$ and public coin size $O(n^2)$, which are computable by a uniform family of polynomial-size depth-$O(d)$ circuits with unbounded-fan-in AND, OR and XOR gates.*

Proof sketch. The proof is essentially the same as that of Theorem 4.4, replacing the Bernoulli randomness extractor with the ψ_μ^{n+q} sampler. That is, decisional ψ_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$ for $q=\Theta(n)$ implies a constant-depth polynomial-stretch randomized PRG on seed length $2\mu(n+q)\log(n+q) = \Theta(n^{1-c}\log n)$ and output length $\Theta(n)$, which in turn implies a nearly constant-depth randomized PRF, where the technique in Lemma 4.3 is also used to make the construction security preserving. □

[9] Lemma 4.4 from the full version of [5] states a variant of Theorem 5.2 for uniformly random a and s, and arbitrary e. However, by checking its proof it actually only requires the matrix a to be uniform and independent of (s,e).

Acknowledgments

Yu Yu is more than grateful to Alon Rosen for motivating this work and many helpful suggestions, and he also thanks Siyao Guo for useful comments. The authors thank Ilan Komargodski for pointing out that the domain extension technique from [10] can also be applied to our constructions with improved efficiency. Yu Yu was supported by the National Basic Research Program of China Grant number 2013CB338004, the National Natural Science Foundation of China Grant (Nos. 61472249, 61572192). John Steinberger was funded by National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61361136003, and by the China Ministry of Education grant number 20121088050.

References

[1] Related work on LPN-based authentication schemes. http://www.ecrypt.eu.org/lightweight/index.php/HB.

[2] Adi Akavia, Andrej Bogdanov, Siyao Guo, Akshay Kamath, and Alon Rosen. Candidate weak pseudorandom functions in $AC^0 \circ MOD_2$. In *Innovations in Theoretical Computer Science, ITCS'14*, pages 251–260, 2014.

[3] Michael Alekhnovich. More on average case vs approximation complexity. In *44th Annual Symposium on Foundations of Computer Science (FOCS 2003)*, pages 298–307, Cambridge, Massachusetts, 2003. IEEE.

[4] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *Advances in Cryptology - CRYPTO 2009*, pages 595–618, 2009.

[5] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography with constant input locality. In *Advances in Cryptology - CRYPTO 2007*, pages 92–110, 2007. Full version available at http://www.eng.tau.ac.il/~bennyap/pubs/input-locality-full-revised-1.pdf.

[6] Abhishek Banerjee, Chris Peikert, and Alon Rosen. Pseudorandom functions and lattices. In *Advances in Cryptology - EUROCRYPT 2012*, pages 719–737, 2012.

[7] Anja Becker, Antoine Joux, Alexander May, and Alexander Meurer. Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding. In *Advances in Cryptology - EUROCRYPT 2012*, pages 520–536, 2012.

[8] Mihir Bellare, Oded Goldreich, and Hugo Krawczyk. Stateless evaluation of pseudorandom functions: Security beyond the birthday barrier. In *Advances in Cryptology - CRYPTO '99*, pages 270–287, 1999.

[9] Elwyn Berlekamp, Robert J. McEliece, and H.C.A. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.

[10] Itay Berman, Iftach Haitner, Ilan Komargodski, and Moni Naor. Hardness preserving reductions via cuckoo hashing. In *Proceedings of the 10th Theory of Cryptography Conference (TCC 2013)*, pages 40–59, 2013.

[11] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Smaller decoding exponents: Ball-collision decoding. In *Advances in Cryptology - CRYPTO 2011*, pages 743–760, 2011.

[12] Avrim Blum, Merrick L. Furst, Michael J. Kearns, and Richard J. Lipton. Cryptographic primitives based on hard learning problems. In *Advances in Cryptology – CRYPTO' 93*, pages 278–291, 1993.

[13] Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *Journal of the ACM*, 50(4):506–519, 2003.

[14] Alexandra Boldyreva, Serge Fehr, and Adam ONeill. On notions of security for deterministic encryption, and efficient constructions without random oracles. In *Advances in Cryptology–CRYPTO 2008*, pages 335–359. Springer, 2008.

[15] Anne Canteaut and Florent Chabaud. A new algorithm for finding minimum-weight words in a linear code: Application to McEliece's cryptosystem and to narrow-sense BCH codes of length 511. *IEEE Transactions on Information Theory*, 44(1):367–378, 1998.

[16] David Cash, Eike Kiltz, and Stefano Tessaro. Two-round man-in-the-middle security from LPN. In *Proceedings of the 13th Theory of Cryptography (TCC 2016-A)*, pages 225–248, 2016.

[17] Nishanth Chandran and Sanjam Garg. Balancing output length and query bound in hardness preserving constructions of pseudorandom functions. In *Proceedings of the 15th International Conference on Cryptology in India (INDOCRYPT 2014)*, pages 89–103, 2014.

[18] Shan Chen and John Steinberger. Tight security bounds for key-alternating ciphers. In *Advances in Cryptology–EUROCRYPT 2014*, pages 327–350. Springer, 2014.

[19] Bernardo David, Rafael Dowsley, and Anderson C. A. Nascimento. Universally composable oblivious transfer based on a variant of LPN. In *Proceedings of the 13th International Conference on Cryptology and Network Security (CANS 2014)*, pages 143–158, 2014.

[20] Yevgeniy Dodis, Eike Kiltz, Krzysztof Pietrzak, and Daniel Wichs. Message authentication, revisited. In *Advances in Cryptology – EUROCRYPT 2012*, pages 355–374, 2012.

[21] Yevgeniy Dodis and Adam Smith. Entropic security and the encryption of high entropy messages. In *2nd Theory of Cryptography Conference – TCC 2005*, pages 556–577, 2005.

[22] Nico Döttling, Jörn Müller-Quade, and Anderson C. A. Nascimento. IND-CCA secure cryptography based on a variant of the LPN problem. In *Advances in Cryptology – ASIACRYPT 2012*, pages 485–503, 2012.

[23] Nico Döttling and Dominique Schröder. Efficient pseudorandom functions via on-the-fly adaptation. In *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference*, pages 329–350, 2015.

[24] Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. New results for learning noisy parities and halfspaces. In *47th Symposium on Foundations of Computer Science*, pages 563–574, Berkeley, CA, USA, October 21–24 2006. IEEE.

[25] Peter Gazi and Stefano Tessaro. Secret-key cryptography from ideal primitives: A systematic overview. In *2015 IEEE Information Theory Workshop (ITW 2015)*, pages 1–5, 2015.

[26] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, October 1986.

[27] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1994.

[28] Iftach Haitner, Omer Reingold, and Salil P. Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. In *Proceedings of the 42nd ACM Symposium on the Theory of Computing*, pages 437–446, 2010.

[29] J. Håstad, R. Impagliazzo, L.A. Levin, and M. Luby. Construction of pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[30] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[31] Nicholas J. Hopper and Manuel Blum. Secure human identification protocols. In *Advances in Cryptology – ASIACRYPT 2011*, pages 52–66, 2001.

[32] Russell Impagliazzo and David Zuckerman. How to recycle random bits. In *30th Annual Symposium on Foundations of Computer Science*, pages 248–253, Research Triangle Park, North Carolina, 30 October–1 November 1989. IEEE.

[33] Abhishek Jain, Stephan Krenn, Krzysztof Pietrzak, and Aris Tentes. Commitments and efficient zero-knowledge proofsfrom learning parity with noise. In *Advances in Cryptology – ASIACRYPT 2012*, pages 663–680, 2012.

[34] Abhishek Jain, Krzysztof Pietrzak, and Aris Tentes. Hardness preserving constructions of pseudorandom functions. In *Proceedings of the 9th Theory of Cryptography Conference (TCC 2013)*, pages 369–382, 2012.

[35] Ari Juels and Stephen A. Weis. Authenticating pervasive devices with human protocols. In Victor Shoup, editor, *Advances in Cryptology—CRYPTO 2005*, volume 3621 of *LNCS*, pages 293–308. Springer-Verlag, 14–18 August 2005.

[36] Jonathan Katz and Ji Sun Shin. Parallel and concurrent security of the hb and hb^+ protocols. In Serge Vaudenay, editor, *Advances in Cryptology—EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 73–87. Springer-Verlag, 2006.

[37] Eike Kiltz, Daniel Masny, and Krzysztof Pietrzak. Simple chosen-ciphertext security from low-noise lpn. In *Proceeding of the 17th Conference on Theory and Practice in Public Key Cryptography (PKC 2014)*, pages 1–18, 2003.

[38] Eike Kiltz, Krzysztof Pietrzak, David Cash, Abhishek Jain, and Daniele Venturi. Efficient authentication from hard learning problems. In *Advances in Cryptology – EUROCRYPT 2011*, pages 7–26, 2011.

[39] Paul Kirchner. Improved generalized birthday attack. Cryptology ePrint Archive, Report 2011/377, 2011. http://eprint.iacr.org/2011/377.

[40] Eric Levieil and Pierre-Alain Fouque. An improved lpn algorithm. In *Proceedings of the 5th Conference on Security and Cryptography for Networks (SCN 2006)*, pages 348–359, 2006.

[41] Leonid A. Levin. One-way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.

[42] Vadim Lyubashevsky. The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem. In *Proceedings of the 9th International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM 2005)*, pages 378–389, 2005.

[43] Vadim Lyubashevsky and Daniel Masny. Man-in-the-middle secure authentication schemes from lpn and weak prfs. In *Advances in Cryptology - CRYPTO 2013*, pages 308–325, 2013.

[44] Ueli M. Maurer. Indistinguishability of random systems. In *Advances in Cryptology - EUROCRYPT 2002*, pages 110–132, 2002.

[45] Alexander May, Alexander Meurer, and Enrico Thomae. Decoding random linear codes in $\tilde{O}(2^{0.054n})$. In *Advances in Cryptology – ASIACRYPT 2011*, pages 107–124, 2011.

[46] Moni Naor and Omer Reingold. Number-theoretic constructions of efficient pseudo-random functions. In *38th Annual Symposium on Foundations of Computer Science*, pages 458–467, Miami Beach, Florida, 20–22 October 1997. IEEE.

[47] Moni Naor, Omer Reingold, and Alon Rosen. Pseudo-random functions and factoring. *Electronic Colloquium on Computational Complexity (ECCC)*, TR01-064, 2001.

[48] Jacques Patarin. The coefficients h technique. In *Selected Areas in Cryptography*, pages 328–345. Springer, 2008.

[49] Krzysztof Pietrzak. Cryptography from learning parity with noise. In *Proceedings of the Theory and Practice of Computer Science (SOFTSEM 2012)*, pages 99–114, 2012.

[50] Alexander A. Razborov. Lower bounds on the size of bounded depth networks over a complete basis with logical addition. *Mathematische Zametki*, 41:598–607, 1986. English Translation in *Mathematical Notes of the Academy of Sciences of the USSR.*

[51] Alexander A. Razborov and Steven Rudich. Natural proofs. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on the Theory of Computing*, pages 204–213, Montréal, Québec, Canada, 23–25 May 1994.

[52] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC 2005).*

[53] Roman Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC 1987)*, pages 77–82, 1987.

[54] Jacques Stern. A method for finding codewords of small weight. In *Coding Theory and Applications, 3rd International Colloquium*, pages 106–113, 1988.

[55] Yu Yu, Dawu Gu, Xiangxue Li, and Jian Weng. (almost) optimal constructions of UOWHFs from 1-to-1, regular one-way functions and beyond. In *Advances in Cryptology – CRYPTO 2015*, pages 209–229, 2015.

A Well-known Facts, Lemmas and Inequalities

Fact A.1 *Let* $\mathbf{H}(\mu) \stackrel{def}{=} \mu\log(1/\mu) + (1-\mu)\log(1/(1-\mu))$ *be the binary entropy function. Then, for any* $0 < \mu < 1/2$ *it holds that*

$$\mu\log(1/\mu) < \mathbf{H}(\mu) < \mu(\log(1/\mu) + 2).$$

Proof.

$$
\begin{aligned}
& \mu\log(1/\mu) \\
< \;& \Big(\mathbf{H}(\mu) = \mu\log(1/\mu) + (1-\mu)\log(1/(1-\mu))\Big) \\
= \;& \mu\log(1/\mu) + (1-\mu)\log(1+\frac{\mu}{1-\mu}) \\
= \;& \mu\log(1/\mu) + (1-\mu)\frac{\ln(1+\frac{\mu}{1-\mu})}{\ln 2} \\
\leq \;& \mu\log(1/\mu) + \frac{\mu}{\ln 2} \;<\; \mu(\log(1/\mu)+2)\ ,
\end{aligned}
$$

where the first inequality is due to $(1-\mu)\log(1/(1-\mu)) > 0$, the second one follows from the elementary inequality $\ln(1+x) \leq x$ for any $x > 0$, and the last inequality is simply $1 < 2\ln 2$. □

Lemma A.1 (Chernoff bound) *For any $n \in \mathbb{N}$, let $X_1, \ldots, X_n$ be independent random variables and let $\bar{X} = \sum_{i=1}^{n} X_i$, where $\Pr[0\leq X_i \leq 1] = 1$ holds for every $1 \leq i \leq n$. Then, for any $\Delta_1 > 0$ and $0 < \Delta_2 < 1$,*

$$
\begin{aligned}
\Pr[\ \bar{X} \ > \ (1+\Delta_1)\cdot\mathbb{E}[\bar{X}]\ ] &< \exp^{-\frac{\min(\Delta_1,\Delta_1^2)}{3}\mathbb{E}[\bar{X}]}\ , \\
\Pr[\ \bar{X} \ < \ (1-\Delta_2)\cdot\mathbb{E}[\bar{X}]\ ] &< \exp^{-\frac{\Delta_2^2}{2}\mathbb{E}[\bar{X}]}\ .
\end{aligned}
$$

Theorem A.1 (The Hoeffding bound [30]) *Let $q \in \mathbb{N}$, and let $\xi_1, \xi_2, \ldots, \xi_q$ be independent random variables such that for each $1 \leq i \leq q$ it holds that $\Pr[a_i \leq \xi_i \leq b_i] = 1$. Then, for any $t > 0$ we have*

$$
\Pr\Big[\ \Big|\sum_{i=1}^{q}\xi_i \ - \ \mathbb{E}[\sum_{i=1}^{q}\xi_i]\Big| \ \geq \ t\ \Big] \ \leq \ 2\exp^{-\frac{2t^2}{\sum_{i=1}^{q}(b_i-a_i)^2}}\ .
$$

Fact A.2 *For any $\sigma \in \mathbb{N}^+$, the probability that a random $(n+\sigma)\times n$ Boolean matrix $M \sim U_{(n+\sigma)\times n}$ has full rank (i.e., rank n) is at least $1-2^{-\sigma+1}$.*

Proof. Consider matrix M being sampled column by column, and denote $\mathcal{E}_i$ to be the event that "column i is non-zero and neither is it any linear combination of the preceding columns (i.e., columns 1 to $i-1$)".

$$
\begin{aligned}
\Pr[\ M \text{ has full rank }\ ] \ &= \ \Pr[\mathcal{E}_1]\cdot\Pr[\mathcal{E}_2|\mathcal{E}_1]\cdot\cdots\cdot\Pr[\mathcal{E}_n|\mathcal{E}_{n-1}] \\
&= \ (1-2^{-(n+\sigma)})\cdot(1-2^{-(n+\sigma)+1})\cdot\cdots\cdot(1-2^{-(n+\sigma)+n-1}) \\
&> \ 2^{-\left(2^{-(n+\sigma)+1}+2^{-(n+\sigma)+2}+\ldots+2^{-(n+\sigma)+n}\right)} \\
&> \ 2^{-2^{-\sigma+1}} \\
&> \ \exp^{-2^{-\sigma+1}} \\
&> \ 1-2^{-\sigma+1}
\end{aligned}
$$

where the first inequality is due to Fact A.4 and the last follows from Fact A.3. □

Fact A.3 *For any $x > 0$ it holds that $\exp^{-x} > 1-x$.*

Fact A.4 *For any $0 < x < \frac{2-\sqrt{2}}{2}$ it holds that $1-x > 2^{-(\frac{2+\sqrt{2}}{2})x} > 2^{-2x}$.*

Fact A.5 (A partial sum of binomial coefficients ([27], p.492)) *For any $0 < \mu < 1/2$, and any $m \in \mathbb{N}$*

$$\sum_{i=0}^{m\mu} \binom{m}{i} \;=\; 2^{m\mathbf{H}(\mu) - \frac{\log m}{2} + O(1)}$$

where $\mathbf{H}(\mu) \stackrel{def}{=} \mu\log(1/\mu) + (1-\mu)\log(1/(1-\mu))$ is the binary entropy function.

Fact A.6 (Piling-up Lemma) *For any $0 < \mu \le \mu' < 1/2$, $(\mathsf{Ber}_\mu \oplus \mathsf{Ber}_{\frac{\mu'-\mu}{1-2\mu}}) \sim \mathsf{Ber}_{\mu'}$.*

Fact A.7 (Min-entropy source conditioned on leakage) *Let X be any random variable over support $\mathcal{X}$ with $\mathbf{H}_\infty(X) \ge l_1$, let $f : \mathcal{X} \to \{0,1\}^{l_2}$ be any function. Then, for any $0 < \varepsilon < 1$, there exists a set $\mathcal{X}_1 \times \mathcal{Y}_1 \subseteq \mathcal{X} \times \{0,1\}^{l_2}$ such that $\Pr[\,(X, f(X)) \in (\mathcal{X}_1 \times \mathcal{Y}_1)\,] \ge 1 - \varepsilon$ and for every $(x,y) \in (\mathcal{X}_1 \times \mathcal{Y}_1)$*

$$\Pr[\,X = x \mid f(X) = y\,] \le 2^{-(l_1 - l_2 - \log(1/\varepsilon))}.$$

B Lemmas and Proofs Omitted

Proof of Lemma 3.1. Recall that $\mathbf{H}(\mu) \stackrel{\text{def}}{=} \mu\log(1/\mu) + (1-\mu)\log(1/(1-\mu))$ equals to $\mathbf{H}_1(\mathsf{Ber}_\mu)$. Parse Ber_μ^q as Boolean variables $E_1,\ldots,E_q$, and for each $1 \le i \le q$ define

$$\xi_i \stackrel{\text{def}}{=} \begin{cases} 1, \text{ if } E_i = 1 \\ \frac{\log(\frac{1}{1-\mu})}{\log(\frac{1}{\mu})}, \text{ if } E_i = 0 \end{cases}$$

and thus we have that $\xi_1, \ldots, \xi_q$ are i.i.d. over $\{\frac{\log(1/(1-\mu))}{\log(1/\mu)}, 1\}$, each of expectation $\mathbf{H}(\mu)/\log(1/\mu)$.

$$\begin{aligned} &\Pr\left[\, \mathsf{Ber}_\mu^q \in \mathcal{E} \,\right] \\ =\;\; & 1 \;-\; \Pr\left[\, \sum_{i=1}^{q} \xi_i \;>\; (1+\Delta)\cdot\frac{q\mathbf{H}(\mu)}{\log(1/\mu)} \,\right] \\ >\;\; & 1 \;-\; \exp^{-\frac{\min(\Delta,\Delta^2)q\mathbf{H}(\mu)}{3\log(1/\mu)}} \;>\; 1 \;-\; \exp^{-\frac{\min(\Delta,\Delta^2)\mu q}{3}} \;, \end{aligned}$$

where the inequality follows from the Chernoff bound (see Lemma A.1) and we recall $\mathbf{H}(\mu) > \mu\log(1/\mu)$ by Fact A.1. □

Proof of Lemma 4.1.

Decisional $\mathsf{LPN}_{\mu,n}$ $\to$ decisional Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$

Assume for contradiction there exists a distinguisher D that

$$\Pr_{A,S,E}[\,\mathsf{D}(A,\; A{\cdot}S \oplus E) = 1\,] \;-\; \Pr_{A,U_{q-(n+2)}}[\,\mathsf{D}(A, U_{q-(n+2)}) = 1\,] \;>\; 2\epsilon,$$

where $A \sim U_{(q-(n+2))n}$, $S \sim \mathsf{Ber}_\mu^n$ and $E \sim \mathsf{Ber}_\mu^{q-(n+2)}$. To complete the proof, we show that there exists another D' (of nearly the same complexity as D) that on input $(a', b) \in \{0,1\}^{qn} \times \{0,1\}^q$ that distinguishes $(A', A' \cdot X \oplus \mathsf{Ber}_\mu^q)$ from (A', U_q) for $A' \sim U_{qn}$ and $X \sim U_n$ with advantage more than ϵ. We parse the $q \times n$ matrix a' and q-bit b as

$$a' = \begin{bmatrix} m \\ a \end{bmatrix},\; b = (b_m, b_a) \tag{9}$$

where m and a are $(n+2) \times n$ and $(q-(n+2)) \times n$ matrices respectively, $b_m \in \{0,1\}^{n+2}$ and $b_a \in \{0,1\}^{q-(n+2)}$. Algorithm D' does the following: it first checks whether m has full rank or not, and if not it outputs a random bit. Otherwise (i.e., m has full rank), D' outputs $\mathsf{D}(a\bar{m}^{-1}, (a\bar{m}^{-1}){\cdot}b_{\bar{m}} \oplus b_a)$, where $\bar{m}$ is an $n \times n$ invertible submatrix of m and $b_{\bar{m}}$ is the corresponding[10] substring of b_m. Now we

[10] E.g., if $\bar{m}$ is the submatrix of m by keeping only the first n rows, then $b_{\bar{m}}$ is the n-bit prefix of b_m.

give the lower bound of the advantage in distinguishing the two distributions. On the one hand, when $(a',b) \leftarrow (A', (A' \cdot X) \oplus \mathsf{Ber}_\mu^q)$ and conditioned on that $\bar{m}$ is invertible, we have that

$$\begin{aligned} \bar{m} \cdot x \oplus s = b_{\bar{m}} \\ a \cdot x \oplus e = b_a \end{aligned} \tag{10}$$

where $a \leftarrow U_{(q-(n+2))n}$, $x \leftarrow U_n$, $s \leftarrow \mathsf{Ber}_\mu^n$, and $e \leftarrow \mathsf{Ber}_\mu^{q-(n+2)}$, and it follows (by elimination of x) that $b_a = (a\bar{m}^{-1})s \oplus (a\bar{m}^{-1})b_{\bar{m}} \oplus e$, and thus $(a\bar{m}^{-1})b_{\bar{m}} \oplus b_a = (a\bar{m}^{-1})s \oplus e$. On the other hand, when $(a',b) \leftarrow (U_{qn}, U_q)$ and conditioned on an invertible m it holds that $(a\bar{m}^{-1}, (a\bar{m}^{-1}) \cdot b_{\bar{m}} \oplus b_a)$ follows $(U_{(q-(n+2))n}, U_{q-(n+2)})$. Therefore, for $A \sim U_{(q-(n+2))n}$, $S \sim \mathsf{Ber}_\mu^n$ and $E \sim \mathsf{Ber}_\mu^{q-(n+2)}$ we have

$$\begin{aligned} & \Pr[\ \mathsf{D}'(U_{qn}, U_{qn} \cdot U_n \oplus \mathsf{Ber}_\mu^q) = 1\ ] \ - \ \Pr[\ \mathsf{D}'(U_{qn}, U_q) = 1\ ] \\ \geq \quad & \Pr[\mathcal{E}_f] \cdot \left(\Pr_{A,S,E}[\mathsf{D}(A,\ A \cdot S \oplus E) = 1] - \Pr_{A, U_{q-(1+\delta)n}}[\ \mathsf{D}(A, U_{q-(1+\delta)n}) = 1\ ] \right) \\ > \quad & (1 - 2^{-1}) 2\epsilon \ = \ \epsilon \end{aligned}$$

where $\mathcal{E}_f$ denotes the event that $m \leftarrow U_{(n+2)\times n}$ has full rank whose lower bound probability is given in Fact A.2.

Computational $\mathsf{LPN}_{\mu,n}$ $\rightarrow$ computational Ber_μ^{n+q}-$\mathsf{LPN}_{\mu,n}$

The reduction follows steps similar to that of the decisional version. Assume for contradiction there exists a distinguisher D that

$$\Pr_{A,S,E}[\ \mathsf{D}(A,\ A \cdot S \oplus E) = (S,E)\ ] \ > \ 2\epsilon,$$

where $A \sim U_{(q-(n+2))n}$, $S \sim \mathsf{Ber}_\mu^n$ and $E \sim \mathsf{Ber}_\mu^{q-(n+2)}$, then there exists another D' that on input $(a', b = a'x \oplus e') \in \{0,1\}^{qn} \times \{0,1\}^q$ recovers (x, e') with probability more than ϵ. Similarly, D' parses (a',b) as in (9), checks if m has full rank and we define $\bar{m}$, $b_{\bar{m}}$ and $\mathcal{E}_f$ same as the above reduction. Let $(s^*, e^*) \leftarrow \mathsf{D}(a\bar{m}^{-1}, (a\bar{m}^{-1}) \cdot b_{\bar{m}} \oplus b_a)$. As analyzed above, conditioned on $\mathcal{E}_f$ we have $(a\bar{m}^{-1}) \cdot b_{\bar{m}} \oplus b_a = (a\bar{m}^{-1})s \oplus e$ where $(a\bar{m}^{-1}, s, e)$ follows distribution (A,S,E) defined above, and hence $(s^*, e^*) = (s, e)$ with probability more than 2ϵ. Once D' got s^*, it computes $x^* = \bar{m}^{-1} \cdot (b_{\bar{m}} \oplus s^*)$ (see (10)), $e'^* = a'x^* \oplus b$ and outputs (x^*, e'^*).

$$\begin{aligned} & \Pr[\ \mathsf{D}'(A', A' \cdot X \oplus E') = (X, E')\ ] \\ \geq \quad & \Pr[\ \mathcal{E}_f\ ] \cdot \Pr_{A,S,E}[\ \mathsf{D}(A,\ A \cdot S \oplus E) = (S,E)\ ] \\ > \quad & (1 - 2^{-1}) 2\epsilon \ = \ \epsilon \end{aligned}$$

where $A' \sim U_{qn}$, $X \sim U_n$ and $E' \sim \mathsf{Ber}_\mu^q$.

□

Proof of Lemma 4.3. To prove this indistinguishability result we use Patarin's H-coefficient technique in its modern transcript-based incarnation [48, 18].

Without loss of generality the distinguisher D is deterministic and does not repeat queries. We refer to the case when the D's oracle is $F_{\vec{R},\vec{H}}$ as the *real world* and to the case where the D's oracle is R as the *ideal world*.

D transcript consists of a sequence $(X_1, Y_1), \ldots, (X_q, Y_q)$ of query-answer pairs to its oracle, plus (and following the "transcript stuffing" technique of [18]) the vector $\vec{H} = H_1, \ldots, H_\kappa$ of hash functions, appended to the transcript after the distinguisher has made its last query; in the ideal world, $\vec{H}$ consists of a "dummy" κ-tuple $H_1, \ldots, H_\kappa$ that can be sampled after the distinguisher's last query, and is similarly appended to the transcript.

The probability space underlying the real world is $\Omega_{\mathsf{real}} \stackrel{\text{def}}{=} \mathcal{H}^\kappa \times \mathcal{F}^\kappa_{\ell\to n}$ where $\mathcal{F}_{\ell\to n}$ is the set of all functions from ℓ bits to n bits, with uniform measure. The probability space underlying the ideal world is $\Omega_{\mathsf{ideal}} \stackrel{\text{def}}{=} \mathcal{H}^\kappa \times \mathcal{F}_{n\to n}$ where $\mathcal{F}_{n\to n}$ is the set of all functions from n bits to n bits, also with uniform measure.

We can identify elements of Ω_{real} and/or Ω_{ideal} as "oracles" for D to interact with. We write D^ω for the transcript obtained when D interacts with oracle ω, where $\omega \in \Omega_{\mathsf{real}}$ in the real world and $\omega \in \Omega_{\mathsf{ideal}}$ in the ideal world. Thus, the real-world transcripts are distributed according to $D^{W_{\mathsf{real}}}$ where W_{real} is uniformly distributed over Ω_{real}, while the ideal-world transcripts are distributed according to $D^{W_{\mathsf{ideal}}}$ where W_{ideal} is uniformly distributed over Ω_{ideal}.

A transcript τ is *attainable* if there exists some $\omega \in \Omega_{\mathsf{ideal}}$ such that $D^\omega = \tau$. (Which transcripts are attainable depends on D, but we assume a fixed D.) A transcript $\tau = ((X_1, Y_1), \ldots, (X_q, Y_q), H_1, \ldots, H_\kappa)$ is *bad* if there exists some $i \in [q]$ such that

$$H_j(X_i) \in \{H_j(X_1), \ldots, H_j(X_{i-1})\}$$

for all $j \in \kappa$. We let T_{bad} be the set of bad attainable transcripts, T_{good} the set of non-bad attainable transcripts.

We will show that $\Pr[D^{W_{\mathsf{real}}} = \tau] = \Pr[D^{W_{\mathsf{ideal}}} = \tau]$ for all $\tau \in T_{\mathsf{good}}$. In this case, by Patarin's H-coefficient technique [18], D's distinguishing advantage is upper bounded by $\Pr[D^{W_{\mathsf{ideal}}} \in T_{\mathsf{bad}}]$. We commence by upper bounding the later quantity, and then move to the former claim.

Let $\mathcal{E}_{i,j}$, $(i,j) \in [q] \times [\kappa]$, be the event that

$$H_j(X_i) \in \{H_j(X_1), \ldots, H_j(X_{i-1})\}$$

and let

$$\mathcal{E}_i = \mathcal{E}_{i,1} \wedge \cdots \wedge \mathcal{E}_{i,\kappa}.$$

Since the values $X_1, \ldots, X_q$ and the hash functions $H_1, \ldots, H_\kappa$ are uniquely determined by any $\omega \in \Omega_{\mathsf{ideal}}$ or $\omega \in \Omega_{\mathsf{real}}$, we can write $\mathcal{E}_i(W_{\mathsf{ideal}})$ (in the ideal world) or $\mathcal{E}_i(W_{\mathsf{real}})$ (in the real world) to emphasize that $\mathcal{E}_i$ is a deterministic predicate of the uniformly distributed oracle, in either world. Then

$$(D^{W_{\mathsf{ideal}}} \in T_{\mathsf{bad}}) \iff (\mathcal{E}_1(W_{\mathsf{ideal}}) \vee \cdots \vee \mathcal{E}_q(W_{\mathsf{ideal}})). \tag{11}$$

Moreover,

$$\Pr[\mathcal{E}_{i,j}(W_{\mathsf{ideal}})] \le (i-1)\frac{1}{2^\ell} \le \frac{q}{2^\ell}$$

since the hash functions $H_1, \ldots, H_\kappa$ are chosen independently of everything in the ideal world, and by the universality of $\mathcal{H}$, and

$$\Pr[\mathcal{E}_i(W_{\mathsf{ideal}})] \le \left(\frac{q}{2^\ell}\right)^\kappa$$

since the events $\mathcal{E}_{i,1}, \ldots, \mathcal{E}_{i,\kappa}$ are independent in the ideal world; finally

$$\Pr[D^{W_{\mathsf{ideal}}} \in T_{\mathsf{bad}}] \le q\left(\frac{q}{2^\ell}\right)^\kappa = \frac{q^{\kappa+1}}{2^{\ell\kappa}}$$

by (11) and by a union bound.

To complete the proof, we must show that $\Pr[D^{W_{\mathsf{real}}} = \tau] = \Pr[D^{W_{\mathsf{ideal}}} = \tau]$ for all $\tau \in T_{\mathsf{good}}$. Clearly,

$$\Pr[D^{W_{\mathsf{ideal}}} = \tau] = \frac{1}{2^{nq}} \cdot \frac{1}{|\mathcal{H}|^\kappa}$$

for all attainable τ. Moreover, if

$$\tau = ((x_1, y_1), \ldots, (x_q, y_q), h_1, \ldots, h_\kappa)$$

then it is easy to see that

$$\Pr[D^{W_{\mathsf{real}}} = \tau \mid \vec{H}(W_{\mathsf{real}}) = (h_1, \ldots, h_\kappa)] = \frac{1}{2^{nq}}$$

by induction on the number of distinguisher queries, using $\tau \in T_{\mathsf{good}}$. (We write $\vec{H}(W_{\mathsf{real}})$ for the $\vec{H}$-coordinate of W_{real}.) Since

$$\Pr[\vec{H}(W_{\mathsf{real}}) = (h_1, \ldots, h_\kappa)] = \frac{1}{|\mathcal{H}|^\kappa}$$

this completes the proof. □

Proof of Lemma 5.3.

$$\begin{aligned} & \Pr_{a \xleftarrow{\$} \mathcal{A}} [\ \exists y \in \mathcal{Y} :\ y' \neq y \wedge h_a(y') = h_a(y)\ ] \\ \leq & \sum_{y' \in \mathcal{Y} \setminus \{y\}} \Pr_{a \xleftarrow{\$} \mathcal{A}} [\ h_a(y') = h_a(y)\ ] \\ \leq & \ \ |\mathcal{Y}| \cdot 2^{-l_2} \ \leq \ 2^{-(l_2 - l_1)}, \end{aligned}$$

where the first inequality is a union bound and the second inequality follows by the universality of $\mathcal{H}$. □

Proof of Lemma 5.5. Assume WLOG that $\mu' m$ is integer and use shorthand $p_l \stackrel{\text{def}}{=} \Pr[\ |\mathsf{Ber}^m_{\mu'}| \ = \ l\ ]$ and thus

$$p_{\mu' m} = \binom{m}{\mu' m} \mu^{\mu' m} (1 - \mu')^{m - \mu' m}$$

For $1 \leq i \leq \mu' m$, we have

$$\begin{aligned} p_{\mu' m - i} &= \binom{m}{\mu' m - i} \mu'^{\mu' m - i} (1 - \mu')^{m - \mu' m + i} \\ &= \frac{m! \cdot \mu'^{\mu' m} (1 - \mu')^{m - \mu' m}}{(\mu' m - i)!(m - \mu' m + i)!} \\ &= p_{\mu' m} \frac{(\mu' m - i + 1)(\mu' m - i + 2) \ldots (\mu' m - i + i)}{(m - \mu' m + 1)(m - \mu' m + 2) \ldots (m - \mu' m + i)} \cdot (\frac{1 - \mu'}{\mu'})^i \\ &= p_{\mu' m} \frac{(1 - \frac{i-1}{\mu' m})(1 - \frac{i-2}{\mu' m}) \ldots (1 - \frac{0}{\mu' m})}{(1 + \frac{1}{m(1-\mu')})(1 + \frac{2}{m(1-\mu')}) \ldots (1 + \frac{i}{m(1-\mu')})}. \end{aligned}$$

Similarly, for $1 \leq i \leq (1 - \mu')m$ we can show that

$$p_{\mu' m + i} = p_{\mu' m} \frac{(1 - \frac{0}{m(1-\mu')})(1 - \frac{1}{m(1-\mu')}) \ldots (1 - \frac{i-1}{m(1-\mu')})}{(1 + \frac{1}{\mu' m})(1 + \frac{2}{\mu' m}) \ldots (1 + \frac{i}{\mu' m})}.$$

Therefore, we have $p_{\mu' m} = \max\{p_i \mid 0 \leq i \leq m\ \}$ and thus complete the proof with the following

$$\begin{aligned} (1 + 2\sqrt{m}) \cdot p_{\mu' m} \ &\geq \sum_{j = \mu' m - \min\{\sqrt{m}, \mu' m\}}^{\mu' m + \sqrt{m}} p_j \\ &\geq \ 1 \ - \ \Pr[\ \big|\ |\mathsf{Ber}^m_{\mu'}| - \mu' m \big| \geq \ \sqrt{m}\ ] \\ &\geq \ 1 \ - \ 2\exp^{-2} \ = \Omega(1) \end{aligned}$$

where the last inequality is a Hoeffding bound. □