New Baselines for Local Pseudorandom Number Generators by Field Extensions

Akın Ünal

Department of Computer Science
ETH Zurich
Zurich, Switzerland
akin.uenal@inf.ethz.ch

April 29, 2023

Abstract. We will revisit recent techniques and results on the cryptoanalysis of local pseudorandom number generators (PRGs). By doing so, we will achieve a new attack on PRGs whose time complexity only depends on the algebraic *degree* of the PRG.
Concretely, against PRGs $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ we will give an algebraic attack whose time complexity is bounded by

$$\exp(O(\log(n)^{\deg F/(\deg F-1)} \cdot n^{1-e/(\deg F-1)}))$$

and whose advantage is at least $1 - o(1)$ in the worst case.
To the best of the author's knowledge, this attack outperforms current attacks on the pseudorandomness of local random functions with guaranteed noticeable advantage and gives a new baseline algorithm for local PRGs. Furthermore, this is the first subexponential attack that is applicable to polynomial PRGs of constant *degree* over fields of *any* size with a guaranteed noticeable advantage.

Keywords: PRGs · $\mathbf{NC}^0$ · Local Random Functions · Polynomial Equation Systems · Algebraic Attacks · Subexponential · Lower Bounds

1 Introduction

Pseudorandom Number Generators. A pseudorandom number generator (PRG) is a deterministic algorithm $F : \{0,1\}^n \to \{0,1\}^m$ that stretches a seed consisting of n bits to a longer string of m bits. Ideally, a PRG guarantees that, if the seed has been sampled uniformly at random, then the bitstring output by the PRG is indistinguishable from a truly random bitstring of length m for a certain class of algorithms. PRGs are part of the foundations of modern theoretical cryptography [23,1]. In the real world, keystream generators (which can be seen as a more advanced form of PRGs) are a popular method to construct fast and reliable symmetric encryption between two clients.

Local PRGs. While it is widely accepted that PRGs do exist from a practical point of view, it is in theory not clear which computational complexity a PRG must have to be reasonably secure while maintaining a high (i.e. polynomial) output length. The most simple but plausibly secure PRGs are so-called *local* PRGs, which have been introduced by Goldreich [22] (also known as *random local functions*, *Goldreich's PRGs* and *PRGs in* $\mathbf{NC}^0$). For each of its output bits, a local PRG only needs to look up a constant number of its input bits. This makes local PRGs from a practical and a theoretical point of view highly efficient. In fact, local PRGs have several applications in theory:

1. Ishai, Kushilevitz, Ostrovsky & Sahai [24] showed that local poly-stretch PRGs (i.e. $m \geq n^{1+e}$ for $e > 0$ constant) together with oblivious transfer imply highly efficient secure communication protocols for two parties evaluating a circuit on private data. In the semi-honest model, where the users abide to the protocol but try to learn as much about the other party's data as possible, the authors could construct a protocol where the computational complexity for both parties is linear in the size of the circuit to be evaluated. In the malicious model, where users may deviate from the protocol, the authors constructed a protocol where the computational complexity of both parties is slightly superlinear in the size of the circuit.
   The first protocol has been extended to arithmetic circuits assuming local poly-stretch PRGs $F : k^n \to k^m$ over a field k [5].
2. Further, local PRGs are interesting for multiparty-computation (MPC) protocols and fully-homomorphic encryption (FHE) schemes. These primitives suffer strongly from computing circuits of large depths. While the complexity of MPC protocols rise with the depth of the to be evaluated circuit, the noise of FHE ciphers grows substantially with the multiplicative depth of the circuit that is to be evaluated on encrypted data. In the case of lattice-based FHE, this forces one to weaken the underlying learning with errors (LWE) assumption to support the evaluation of deeper circuits.
3. Another important application of local PRGs are indistinguishability obfuscation (iO) schemes. Jain, Lin & Sahai [25,26] gave recently new iO schemes whose security is reduced to a number of assumptions, including local PRGs of polynomial stretch.

Subexponential Security. In particular, the recent iO constructions of Jain, Lin & Sahai [25,26] need that the local PRGs they use have stronger security guarantees than usual: the advantage of each poly-time adversary of distinguishing the output of the PRG from uniform randomness must not only be negligible, but smaller than the inverse of some subexponential functional. They dub this *subexponential security*, and, in fact, require subexponential security of all assumptions they use.

This raises the interest in attack algorithms on local PRGs whose runtime are beyond poly-time and whose advantage is below negligible. While some efficient attacks on local PRGs have been known for quite some time, there has been a recent interest in subexponential attacks on PRGs [33,7,9,39,16]. While

efficient attacks are only applicable when a certain stretch of the PRG is given, subexponential attacks are applicable even for small polynomial stretches, and degrade gracefully with decreasing stretch of PRGs. Hence, when estimating the concrete security of Goldreich's PRGs, existing subexponential attacks must be taken into consideration [16].

Faster Baselines for Local PRGs. In [39], the author gave new algebraic attacks on local PRGs and PRGs of constant degree over large fields. In the initial version of [39], he mistakenly claimed that his attacks would give new baselines for local PRGs. This claim was erroneous as there are so-called *shrinking-set attacks* [38,6] that provably break local PRGs of poly-stretch in subexponential time and are by a log-factor in the exponent faster than the attacks given in [39].

However, as we will show in this paper, by improving the techniques of [39] we can improve substantially the time complexity of algebraic attacks on PRGs s.t. their runtimes only depend on the *degree* of PRGs rather than their *locality* and still retain a high advantage. Since the degree of a PRG must always be smaller than its locality, this yields a new algebraic attack on local PRGs that surpasses the shrinking-set attack.

1.1 Contribution

In this work, we will revisit and improve the techniques of [39]. On PRGs $F : \{0,1\}^n \to \{0,1\}^m$ of constant *degree* d over $\mathbb{Z}_2$ and poly-stretch $m \geq n^{1+e}$, our improvements will yield an attack algorithm of subexponential time complexity $2^{O(\log(n)^{d/(d-1)} \cdot n^{1-e/(d-1)})}$ and high advantage $1 - O\left((\log(n)/n^e)^{1/(d-1)}\right)$.

We give a generalized version of this attack that works on constant-degree PRGs $F : k^n \to k^m$ over *any* field k.

To the best of the author's knowledge, this is the first attack on binary constant-degree poly-stretch PRGs that is provably subexponential and has a provable non-negligible advantage, even in the worst case. Since the time complexity of this attack is independent of the locality of the PRG, this attack outperforms other subexponential attacks on local PRGs that have a provably non-negligible advantage in the worst case and gives us new faster baseline for distinguishing the output of random local functions from true randomness.

1.2 Technical Overview

For a PRG F, we will in the following denote by $\deg F$ its (algebraic) degree over its base field, and by $\operatorname{loc} F$ its locality.

Note that the attacks against (local) PRGs in [39] are based on two simple tricks: a basic algebraic attack and a hashing resp. reduction technique. The basic algebraic attack has a subexponential runtime and works well against PRGs $F : k^n \to k^m$ of constant degree over a large field k, however it struggles with PRGs over small fields, e.g. $k = \mathbb{Z}_2 = \{0,1\}$.

The hashing technique compensates this problem by turning a binary PRG $F : \{0,1\}^n \to \{0,1\}^m$ into a PRG $G : k^n \to k^{m'}$ with $m' \approx m$ s.t. the degree of G equals the locality of F.

We will revisit both techniques in more detail:

The Basic Algebraic Attack. This attack appears in [39] as well as in [38]. Assume we are given a PRG $F : k^n \to k^m$ of constant degree over a field k. Denote by $f_1, \ldots, f_m \in k[X] := k[X_1, \ldots, X_n]$ the polynomials that compute the output values of F.

Now assume that we were aware of an algebraic relationship h between the polynomials $f_1, \ldots, f_m$. I.e., a new polynomial $h \in k[Y] := k[Y_1, \ldots, Y_m]$ s.t. $h(Y_1, \ldots, Y_m)$ is not the zero polynomial, but $h(f_1(X), \ldots, f_m(X))$ is the zero polynomial in $k[X]$. The key observation is that we can use h to check if a point $y \in k^m$ lies in the image of F. Indeed, if y equals $F(x)$ for some $x \in k^n$, then we must have

$$h(F(x)) = (h \circ F)(x) = (h(f_1, \ldots, f_m))(x) = (0)(x) = 0. \tag{1}$$

On the other side, because of the famous Schwartz-Zippel Lemma we have for a uniformly random $y \leftarrow k^m$

$$\Pr_{y \leftarrow k^m}[h(y) \neq 0] \geq 1 - \frac{\deg h}{\#k}. \tag{2}$$

A distinguisher that uses h to play the pseudorandomness game for the PRG F would therefore have an advantage of $1 - \frac{\deg h}{\#k}$, which converges towards 1 if the size of k grows faster than the degree of h.

Hence, an algebraic relationship of low degree among the output values of F helps substantially in distinguishing the outputs of F from true randomness. However, how can we compute such a relationship and, more importantly, how can we bound the degree of h in the worst case?

In [39], it was shown that h will be sublinear if F is of poly-stretch and of constant degree. In fact, we have $\deg h \in O(n^{1-\frac{e}{d-1}})$ if $m \geq n^{1+e}$ and $d = \deg F$. This was shown by considering the morphism of rings

$$\phi : k[Y_1, \ldots, Y_m] \longrightarrow k[X_1, \ldots, X_n] \tag{3}$$

$$g(Y_1, \ldots, Y_m) \longmapsto g(f_1(X), \ldots, f_m(X)) \tag{4}$$

that substitutes each variable Y_i by the polynomial f_i. Now, h is a non-zero kernel element of ϕ of minimal degree. To estimate the degree of h, we restrict ϕ to the subspace of

$$k[Y]^{\leq D} := k[Y_1, \ldots, Y_m]^{\leq D} := \{g \in k[Y] \mid \deg g \leq D\} \tag{5}$$

of polynomials of degree $\leq D$. When restricted to $k[Y]^{\leq D}$, the image of ϕ is contained in

$$k[X]^{\leq dD} := k[X_1, \ldots, X_n]^{\leq dD} := \{g \in k[X] \mid \deg g \leq dD\}. \tag{6}$$

In fact, when we plug the degree-d polynomials in $g(Y_1, \ldots, Y_m)$ the degree of the resulting polynomial $\phi(g) = g(f_1(X), \ldots, f_m(X))$ will be stretched a factor of d at most. This means, by restricting ϕ to $k[Y]^{\leq D}$ we get a linear map

$$\phi_D : k[Y_1, \ldots, Y_m]^{\leq D} \longrightarrow k[X_1, \ldots, X_n]^{\leq dD} \tag{7}$$

of spaces of finite dimensions. According to the dimension formula for linear maps we now have

$$\begin{aligned} \dim \ker \phi_D \geq & \dim k[Y_1, \ldots, Y_m]^{\leq D} - \dim k[X_1, \ldots, X_n]^{\leq dD} && (8) \\ = & \binom{m+D}{D} - \binom{n+dD}{dD}. && (9) \end{aligned}$$

I.e., we can guarantee the existence of an algebraic relationship of degree $\leq D$ whenever $\binom{m+D}{D} > \binom{n+dD}{dD}$.

In [39], it was shown that Eq. (9) does hold for some sublinear D. In fact, it has been shown:

Lemma 1 ([39] Lemma 7). *Let $d \in \mathbb{N}, d \geq 2$. Let $m : \mathbb{N} \to \mathbb{N}$ be a function with $m(n) \geq 2^{2d-1} \cdot d^{d-1} \cdot n$.*

Then, we have for all integers $n \geq 2d$

$$\binom{m(n) + D(n)}{D(n)} > \binom{n + dD(n)}{dD(n)} \tag{10}$$

where $D(n) = \left\lceil \left(\frac{(2n)^d}{m} \right)^{\frac{1}{d-1}} \right\rceil$. For $m(n) \geq n^{1+e}$, we in particular have for n large enough

$$D(n) = \left\lceil 2^{\frac{d}{d-1}} \cdot n^{1 - \frac{e}{d-1}} \right\rceil \in O(n^{1-\frac{e}{d-1}}). \tag{11}$$

Now, how do we find $h \in \ker \phi$ of degree $\leq D \in O(n^{1-\frac{e}{d-1}})$? Since we know that h is a non-zero element of $\ker \phi_D$, it suffices to write down a matrix representation M of $\phi_D : k[Y]^{\leq D} \to k[X]^{\leq dD}$. M is a matrix of shape $\binom{m+D}{D} \times \binom{n+dD}{dD}$ over k. The algebraic relation h corresponds to non-zero kernel vector of M. Hence, we can use Gaussian elimination to find h. Eliminating the rows of M costs $O(\binom{m+D}{D}^2 \cdot \binom{n+dD}{dD})$ arithmetic operations. Therefore, the memory complexity of finding h is $\binom{m+D}{D} \cdot \binom{n+dD}{dD} \in n^{O(D)} = 2^{O(\log(n) \cdot n^{1-e/(d-1)})}$ and the time complexity lies in $O(\binom{m+D}{D}^2 \cdot \binom{n+dD}{dD}) \subset 2^{O(\log(n) \cdot n^{1-e/(d-1)})}$. Now, evaluating h on a point $y \in k^m$ costs $\deg h \cdot \binom{m+\deg h}{m} \in 2^{O(\log(n) \cdot n^{1-e/(d-1)})}$ arithmetic operations in k.

In total, we get an attack algorithm with space and time complexities in $2^{O(\log(n) \cdot n^{1-e/(d-1)})}$ and noticeable advantage for fields k of size $\geq n$. Formally, it was shown:

Theorem 1 ([39] Theorem 2). *Let $m \in \omega(n)$. Let $F : k^n \to k^m$ be a PRG of constant degree $\deg F$ over k.*

Then, there is an algebraic attack on F whose time and space complexities are bounded from above by $n^{O((n^{\deg F}/m)^{\frac{1}{\deg F-1}})}$. Further, the advantage of this attack in the pseudorandomness game of F (Definition 4) is lower bounded by

$$\mathsf{adv}_F(\mathcal{A}) \geq 1 - O\left(\left(n^{\deg F}/m\right)^{\frac{1}{\deg F-1}} \cdot \frac{1}{\#k}\right). \tag{12}$$

For $m \geq n^{1+e}$ and $\#k \geq n$, the attack is a subexponential algorithm with time and space complexities in $2^{O(\log(n)\cdot n^{1-e/\deg F-1})}$ and high advantage $\geq 1 - O(n^{-e/(\deg F-1)})$.

Hashing to Larger Fields. Note that the above attack does not fare well against PRGs of constant degree over small fields. In the reign of, let's say, $\mathbb{Z}_2 = \{0,1\}$, one can alter the above algorithm to find an algebraic relationship h that is reduced modulo the field equations $Y_1^2 - Y_1, \ldots, Y_m^2 - Y_m$ and prove that the probability of h vanishing on a random point $y \leftarrow \{0,1\}^m$ is bounded by

$$\Pr_{y \leftarrow \{0,1\}^m}[h(y) = 0] \leq 1 - 2^{-\deg h}. \tag{13}$$

However, this will only yield a distinguishing attack with advantage $\geq 2^{-\deg h}$. Since the degree of h will be sublinear in the worst case, we can only guarantee a subexponentially small advantage of this attack, which is very unsatisfactory.

To solve this problem for local PRGs, a simple hashing resp. reduction technique was introduced in [39]: if we are given a PRG $F : \{0,1\}^n \to \{0,1\}^m$ and a point $y \in \{0,1\}^m$, the idea is to convert F and y to a new PRG $G : \{0,1\}^n \to k^{m'}$ and a new point $y \in k^{m'}$ s.t. the following things hold:

1. The size of the field k is large enough i.e. $\#k \geq n$.
2. m' is only by a small factor smaller than m.
3. y' lies in the image of G if y lies in the image of F.
4. y' is close to being uniformly random over $k^{m'}$ if y is sampled uniformly random from $\{0,1\}^m$.

To achieve this, the field $k = \mathbb{Z}_p$ for a prime $p \geq n$ was considered in [39]. Set $m' := \lfloor m/(3\lceil \log p \rceil) \rfloor \in O(m/\log n)$ and note that the distribution of random matrices $A \leftarrow \mathbb{Z}_p^{m \times m'}$ gives us a universal family of hash functions of type $\{0,1\}^m \to \mathbb{Z}_p^{m'}$. According to the leftover hash lemma, the matrix-vector product $A \cdot y$ is statistically close to being uniform for $y \leftarrow \{0,1\}^m$. In fact, we can bound its statistical distance from $z \leftarrow \mathbb{Z}_p^{m'}$ by

$$\Delta\left((A, Ay), (A, z)\right) \leq 2^{-n}. \tag{14}$$

This justifies to set $y' := A \cdot y \in \mathbb{Z}_p^{m'}$.

Now, we define $G : \{0,1\}^n \to \mathbb{Z}_p^{m'}$ as the concatenation $G := A \circ F$. I.e., G first evaluates F normally on its seed and then applies the matrix A to the binary output of F (over $\mathbb{Z}_p$). It is clear that $y' = A \cdot y$ must lie in $G(\{0,1\}^n) = A \cdot F(\{0,1\}^n)$ if y lies in $F(\{0,1\}^n)$.

We are now in a situation where we can apply the algorithm of Theorem 1 on G and y' to decide if y is an image of F or uniformly random. However, to bound the advantage and runtime of this algorithm, we need to know the algebraic degree of G or, more formally, of a representation of G by constant-degree polynomials. It can be shown that G can be computed by polynomials of degree d when F is of *locality* d. I.e., the degree of G (which influences the performance of our algorithm) equals the locality of F, in general.

In total, we get a distinguishing algorithm for the PRG $F : \{0,1\}^n \to \{0,1\}^m$, $m \geq n^{1+e}$, with time and space complexities in $2^{O(\log(n)^{\operatorname{loc} F/(\operatorname{loc} F-1)} \cdot n^{1-e/(\operatorname{loc} F-1)})}$ and advantage $\geq 1 - O\left((\log(n)/n^e)^{1/(\operatorname{loc} F-1)}\right)$ where $\operatorname{loc} F$ is the *locality* of F (the additional $\log(n)^{1/(\operatorname{loc} F-1)}$ factor in the time complexity and the advantage of our attack stems from the fact that m' is by a logarithmic factor smaller than m).

A Bad Trade-Off. Note, that this attack has a high advantage, since $n^{e/(\operatorname{loc} F-1)}$ grows substantially faster than $\log(n)^{1/(\operatorname{loc} F-1)}$. However, the runtime of our attack worsened: while the time and space complexities of the attack in Theorem 1 only depend on the degree of the PRG, the complexities of the new attack depend on its locality. In particular, the attack that uses the old hashing technique in [39] is not applicable to PRGs that are computed by dense polynomials of constant degree over the binary numbers.

In the following, we will improve the hashing technique and, as a result, gain a new subexponential attack algorithm whose runtime is independent of the locality of the PRG.

The New Extension Technique. The problem when hashing from $\mathbb{Z}_2$ to $\mathbb{Z}_p$ is that the embedding $\mathbb{Z}_2 \hookrightarrow \mathbb{Z}_p$ that maps zero to zero and one to one is not homomorphic. While this inclusion preserves multiplication, it does not preserve addition. Indeed, we have $1 + 1 = 0$ in $\mathbb{Z}_2$ but not in $\mathbb{Z}_p$ if $p > 2$. To get a homomorphic inclusion we need to consider extension fields of $\mathbb{Z}_2$.

Set $N := 2^{\lceil \log n \rceil}$ and denote by $\mathbb{F}_N$ the Galois field that has N elements. Up to isomorphism, $\mathbb{F}_N$ is uniquely determined by the number of its elements and there exists a natural homomorphic ring homomorphism $\mathbb{Z}_2 \hookrightarrow \mathbb{F}_N$.

In algebra, it is well known that finite fields are *perfect*, i.e. each extension of finite fields is *separable*. The primitive element theorem postulates that each finite separable field extension $k \subset \overline{k}$ is generated by one element. I.e., there is one $\zeta \in \overline{k}$ s.t. $\overline{k}$ has the basis $1, \zeta, \ldots, \zeta^{r-1}$ as k-vector space (where $r = \dim_k \overline{k}$ is the dimension of $\overline{k}$ as k-vector space). In particular, we have

$$\overline{k} = k[\zeta] \cong k \oplus \zeta \cdot k \oplus \zeta^2 \cdot k \oplus \ldots \oplus \zeta^{r-1} \cdot k \cong k^r \tag{15}$$

where the second and third equalities are isomorphisms of vector spaces ($\oplus$ denotes the direct sum of vector fields).

Since the extension $\mathbb{Z}_2 \subset \mathbb{F}_N$ is separable, it is also generated by one element. Let's call this element ζ, too. Eq. (15) implies that the linear map

$$\psi : \mathbb{Z}_2^{\lceil \log n \rceil} \longrightarrow \mathbb{F}_N \tag{16}$$

$$(b_1, \ldots, b_{\lceil \log n \rceil}) \longmapsto b_1 + b_2 \cdot \zeta + \ldots + b_{\lceil \log n \rceil} \cdot \zeta^{\lceil \log n \rceil - 1} \tag{17}$$

is an isomorphism of vector spaces. This leads to the following two observations:

1. Since ψ is bijective, i.e. one-to-one and onto, it maps uniformly random vectors in $\mathbb{Z}_2^{\lceil \log n \rceil}$ to uniformly random elements in $\mathbb{F}_N$. I.e., if we sample $b \leftarrow \mathbb{Z}_2^{\lceil \log n \rceil}$ then $\psi(b)$ is identically distributed as $c \leftarrow \mathbb{F}_N$.
2. The natural inclusion $\mathbb{Z}_2 \hookrightarrow \mathbb{F}_N$ extends to a natural inclusion of polynomial rings

$$\mathbb{Z}_2[X_1, \ldots, X_n] \hookrightarrow \mathbb{F}_N[X_1, \ldots, X_n]. \tag{18}$$

In fact, we can consider $\mathbb{Z}_2[X_1, \ldots, X_n]$ to be a subring of $\mathbb{F}_N[X_1, \ldots, X_n]$. This inclusion preserves functionality and degree: if we have a polynomial $f \in \mathbb{Z}_2[X_1, \ldots, X_n]$ we can – via the inclusion $\mathbb{Z}_2[X_1, \ldots, X_n] \subset \mathbb{F}_N[X_1, \ldots, X_n]$ – consider f to be a polynomial in $\mathbb{F}_N[X_1, \ldots, X_n]$. Interpreted as a polynomial in $\mathbb{F}_N[X_1, \ldots, X_n]$, f will evaluate on $\{0,1\}^n$ to the same values as before. Further, f – interpreted as element in $\mathbb{F}_N[X_1, \ldots, X_n]$ – has the same degree as $f \in \mathbb{Z}_2[X_1, \ldots, X_n]$. This comes from the fact that – when we go from $\mathbb{Z}_2[X_1, \ldots, X_n]$ to $\mathbb{F}_N[X_1, \ldots, X_n]$ – we only "change" the coefficients of f.
ψ now extends to an isomorphism of vector spaces

$$\psi : (\mathbb{Z}_2[X_1, \ldots, X_n])^{\lceil \log n \rceil} \longrightarrow \mathbb{F}_N[X_1, \ldots, X_n] \tag{19}$$

$$(f_1, \ldots, f_{\lceil \log n \rceil}) \longmapsto f_1 + \zeta \cdot f_2 + \ldots + \zeta^{\lceil \log n \rceil - 1} \cdot f_{\lceil \log n \rceil}. \tag{20}$$

This isomorphism is degree-preserving: if $f_1, f_2, \ldots, f_{\lceil \log n \rceil}$ are all of degree $\leq d$, then so are their scaled versions $f_1, \zeta \cdot f_2, \ldots, \zeta^{\lceil \log n \rceil - 1} \cdot f_{\lceil \log n \rceil}$ and the linear combination $\psi(f_1, \ldots, f_{\lceil \log n \rceil}) = f_1 + \zeta \cdot f_2 + \ldots + \zeta^{\lceil \log n \rceil - 1} \cdot f_{\lceil \log n \rceil}$.

Now, given a PRG $F : \{0,1\}^n \to \{0,1\}^m$ of *degree* d and a point $y \in \{0,1\}^m$, we want to apply ψ on blocks of length $\lceil \log n \rceil$ of F and y. Assume for the simplicity of this exposition that m is a multiple of $\lceil \log n \rceil$ i.e. there is an $m' \in \mathbb{N}$ s.t. $m = \lceil \log n \rceil \cdot m'$. Further, consider the following matrix [1] of shape $m' \times m$ that applies the linear map ψ block-wise on its input:

$$A := I_{m'} \otimes \left(1 \; \zeta \; \ldots \zeta^{\lceil \log n \rceil - 1}\right) = \begin{pmatrix} 1 \; \zeta \; \ldots \zeta^{\lceil \log n \rceil - 1} & & \\ & \ddots & \\ & & 1 \; \zeta \; \ldots \zeta^{\lceil \log n \rceil - 1} \end{pmatrix} \tag{21}$$

where $I_{m'}$ denotes identity matrix of shape $m' \times m'$ and $\otimes$ the Kronecker product. Just as before, we can apply A via matrix-vector multiplication on F and y. I.e., we set

$$y' := A \cdot y \in \mathbb{F}_N^{m'}, \tag{22}$$

[1] It is easy to notice the resemblance of A and the gadget matrix of Micciancio & Peikert [30] in the setting of lattice-based cryptography.

$$G(x) := A \cdot F(x). \tag{23}$$

Then, y lies in the image of F iff y' lies in the image of G. Further, y is uniformly random from $\{0,1\}^m$ iff y' is uniformly distributed in $\mathbb{F}_N^{m'}$. Note that for this property we do not need to sample A uniformly random as before, and we do not need to invoke the leftover hash lemma. However, more importantly, the degree of the PRG $G : \{0,1\}^n \to \mathbb{F}_N^{m'}$, which computes $F(x)$ and then applies A, equals the *degree* of F over $\mathbb{Z}_2$. This comes from the fact that ψ preserves the algebraic degree of polynomials and A applies ψ block-wise to the polynomials computing F.

It follows that we can invoke the attack from Theorem 1 on G and y', if we want to decide if y lies in the image of F. If $m \geq n^{1+e}$, then $m' \geq \frac{n^{1+e}}{\lceil \log n \rceil}$ and the attack from Theorem 1 has time and space complexities in

$$2^{O(\log(n)^{\deg G/(\deg G-1)} \cdot n^{1-e/(\deg G-1)})} = 2^{O(\log(n)^{\deg F/(\deg F-1)} \cdot n^{1-e/(\deg F-1)})} \tag{24}$$

and an advantage of

$$\geq 1 - O\left(\frac{\log(n)^{1/(\deg G-1)} \cdot n^{1-e/(\deg G-1)}}{N}\right) \tag{25}$$

$$\geq 1 - O\left(\frac{\log(n)^{1/(\deg G-1)} \cdot n^{1-e/(\deg G-1)}}{n}\right) \tag{26}$$

$$= 1 - O\left(\frac{\log(n)^{1/(\deg G-1)}}{n^{e/(\deg G-1)}}\right) \tag{27}$$

$$= 1 - O\left(\frac{\log(n)^{1/(\deg F-1)}}{n^{e/(\deg F-1)}}\right). \tag{28}$$

This gives us a subexponential attack algorithm on poly-stretch PRGs $F : \{0,1\}^n \to \{0,1\}^m$ whose runtime is independent of the locality of F. I.e., this attack algorithm is even applicable on PRGs computed by dense polynomials of constant degree.

While we only handled the case of binary PRGs here, the extension technique can naturally be used for PRGs $F : k^n \to k^m$ that are polynomial over any small field k.

We can summarize our result as follows:

Theorem 2 (Main Result). *Let $F : k^n \to k^m$ be a PRG of constant degree $\deg F$ over some field k. Set $r := \max(\log(n)/\log(\#k), 1)$ and assume $m \in \omega(r \cdot n)$.*

Then, there is an algebraic attack on F whose time and space complexities are bounded from above by $n^{O((\frac{r \cdot n^{\deg F}}{m}) \frac{1}{\deg F - 1})}$. Further, the advantage of this attack in the pseudorandomness game of F (Definition 4) is lower bounded by

$$\mathsf{adv}_F(\mathcal{A}) \geq 1 - O\left(\frac{r \cdot n}{m}\right)^{1/(\deg F-1)} \geq 1 - o(1). \tag{29}$$

For $m \geq n^{1+e}$, the attack is a subexponential algorithm with time and space complexities in $2^{O(\log(n)\cdot r^{1/(\deg F-1)}\cdot n^{1-e/(\deg F-1)})}$ and noticeable advantage $\geq 1 - O((r/n^e)^{1/(\deg F-1)})$.

Remark 1 (Avoiding Field Extensions.). The hashing technique in [39] did indeed hash from $\{0,1\}^m$ to $\mathbb{Z}_p^{\lfloor m/(3\lceil \log p\rceil)\rfloor}$ via a random matrix which induced inevitable information loss. However, the new "extension technique" works via a bijection that maps $\{0,1\}^{\lceil \log n\rceil \cdot m'}$ to $\mathbb{F}_N^{m'}$ without any information loss. In fact, this technique is actually just a change in how we view the PRG $F : \{0,1\}^n \to \{0,1\}^m$ and is reversible: a PRG $G : \mathbb{F}_N^n \to \mathbb{F}_N^m$ of degree d over $\mathbb{F}$ can be seen as a PRG $G' : \{0,1\}^{\lceil \log(n)\rceil \cdot n} \to \{0,1\}^{\lceil \log(n)\rceil \cdot m}$ of *degree d over* $\mathbb{Z}_2$ and a PRG $H : \{0,1\}^n \to \mathbb{F}_N^m$ of degree d over $\mathbb{F}$ can be seen as a PRG $H' : \{0,1\}^n \to \{0,1\}^{\lceil \log(n)\rceil \cdot m}$ of *degree d over* $\mathbb{Z}_2$. We explain this further in Remark 3.

1.3 Related Work

To better understand our result we will present here existing attacks on PRGs of constant degree and constant locality, and finally compare their performance.

Attacks on PRGs of Constant Degree.

Relinearization Attacks. Each known attack on PRGs of constant degree over some field is of algebraic nature. A first approach is to understand the equation $F(X) = y$ as a polynomial equation system with n variables $X_1, \ldots, X_n$ and m polynomial equations $f_1(X) = y_1, \ldots, f_m(X) = y_m$. By relinearizing this equation system one can generate a linear equation system, on which one can apply Gaussian elimination. If we have enough equations, i.e. $m \geq \binom{n+\deg F}{\deg F}$, then with high probability this linear equation system can be solved for a possible seed x, or at least the satisfiability of the linear equation system can be checked. This leads to a basic attack on polynomial PRGs that is efficient and very reliable (its advantage is provably noticeable). This attack can already be improved: we don't need that m is greater than $\binom{n+\deg F}{\deg F}$, in fact, it suffices that $m \in \Omega(n^{\deg F})$. If m is smaller than $\binom{n+\deg F}{\deg F}$, but has the same asymptotic complexity then it suffices to populate the linear equation system with more polynomial equations that can be generated from $F(X) = y$ up to some constant degree.

Groebner Bases. Extending the idea of the relinearization-and-elimination algorithm above leads to Groebner basis-based attacks. Groebner bases together with a first algorithm for computing them have been introduced by Buchberger [10]. Faster algorithms have been given by Faugère [20,19], these algorithms are based on Macaulay matrices [28,27]. Additionally, the XL-algorithm with a lot of variations [15,14,17,31,37] have been introduced, which also aim to compute something that is similar to a Groebner basis. The core idea of those algorithms is to solve the polynomial equation system $F(X) = y$ by computing a Groebner

basis for the ideal $(f_1(X)-y_1, \ldots, f_m(X)-y_m) \subset k[X]$ for some monomial ordering. Most algorithms do this by computing a Macaulay matrix for an increasing degree and applying Gaussian elimination on it: the Macaulay matrix for degree D is the matrix where each row represents a polynomial $X^I \cdot (f_i(X) - y_i)$, for a multi-index I with $|I| \leq D - \deg f_i$, and where each column represents a monomial of $k[X]$ up to degree D. I.e., the rows of the Macaulay matrix are the coefficient vectors of polynomials $X^I \cdot (f_i(X) - y_i)$. The columns are ordered according to the monomial ordering. By applying Gaussian elimination to the Macaulay matrix of degree D one can extract a Groebner basis from it, if D is high enough. In most cases, the Groebner basis will be of the shape $\{X_1-x_1, \ldots, X_n-x_n\}$, which allows to directly read off the solution $X = x \in k^n$ of the polynomial equation system $F(X) = y$. Hence, Groebner basis-based attacks are usually inversion attacks that try to extract the seed $X = x$ from the PRG problem $F(X) = y$.

While Groebner basis-based algorithms perform well in reality, it is hard to give formal guarantees for them. In the worst case, the highest degree of polynomials of a reduced Groebner basis for an equation system $F(X) = y$ is doubly exponential [18]. However, the doubly exponential degree only occurs in extreme cases. On average, the maximum degree for which a Macaulay matrix must be computed is suspected to be upper-bounded by the degree of regularity (in the case of graded anti-lexicographic monomial orders [11,12]). The degree of regularity is a popular heuristic for Groebner basis-based algorithms, it has been shown to be smaller than $O(n^{1-e/(d-1)})$ for a system of $m \geq n^{1+e}$ equations of degree d [39]. This would yield an inversion attack of suspected time complexity $n^{O(n^{1-e/(\deg F-1)})}$.

In the case of refutation, better bounds can be given: a Groebner basis-based algorithm for refutation problems only checks if the equation $1 = 0$ can be deduced from a Macaulay matrix of sufficiently high degree (so the monomial ordering does not matter, those algorithms are actually just Macaulay matrix-based). If up to some degree the span of the rows of the Macaulay matrix does not contain a vector that corresponds to a constant non-zero polynomial, then the algorithm assumes that the system $F(X) = y$ is solvable and decides that y lies in the image of the PRG F. Otherwise, the algorithm could prove that $F(X) = y$ is unsatisfiable and refutes y. It has been shown [39] that such algorithms only need to compute the Macaulay matrix up to some degree in $O(n^{1-e/(\deg F-1)})$. If the base field k is large enough ($\#k \geq n$ e.g.), then this approach will provably have an advantage of $1-o(1)$. Otherwise, for small fields, the advantage can still be lower bounded by an inverse of a subexponential function. Our results here extend to this algebraic refutation approach: according to Remark 3, it suffices to compute the Macaulay matrix up to some degree in $O(\log(n)^{1/(\deg F-1)} \cdot n^{1-e/(\deg F-1)})$ for this algorithm to have a provable advantage of $1 - o(1)$.

However, we want to point out that extending a PRG $F : k^n \to k^m$ to $G : k^n \to \overline{k}^{m'}$, $m' \approx m/(\log(n)/\log(\#k))$, will not improve the performance of a Groebner basis resp. Macaulay matrix-based algorithm (i.e. solving $G(x) = y'$ will not be simpler than solving $F(x) = y$ for such algorithms). The reason is that

– simply put – the field extension technique here will already be implicitly used when computing Macaulay matrices, even when all computations happen over the small base field k (as Remark 3 explains one polynomial over $\overline{k}$ in m' variables corresponds to many polynomials over k in m variables of the same degree). Therefore, to use the results here on Groebner basis resp. Macaulay matrix-based algorithms it suffices to increase the degree up to which the Macaulay matrix is computed by a small factor of $\log(n)^{1/(\deg F-1)}$.

New Algebraic Attacks. In his master thesis, Zichron [38], gave a new algebraic attack on polynomial PRGs of constant degree over any field. The new idea is to find an algebraic relationship [2] among the polynomials computing the output values of the PRG. The author proved that the time complexity of this algorithm is subexponential and gave lower bounds for the advantage of this algorithm [39]. In the case of small base fields (like $k = \mathbb{Z}_2$), the author gave a hashing technique that improves the advantage substantially by hashing from $\mathbb{Z}_2$ values to $\mathbb{Z}_p$ values.

On a PRG $F : k^n \to k^{n^{1+e}}$, the algorithm of Zichron and the author has a time complexity of $n^{O(n^{1-e/(\deg F-1)})}$ and a noticeable advantage if k is large. If k is small, the advantage of this algorithm can only be lower-bounded by a subexponentially small function.

Using the hashing technique of [39] on PRGs $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ yields an algorithm of time complexity $n^{O(\log(n)^{1/(\operatorname{loc} F-1)} \cdot n^{1-e/(\operatorname{loc} F-1)})}$ and advantage $1 - o(1)$. However, note that in this case the complexity depends on the locality of the PRG, which makes this algorithm with hashing technique only applicable to binary PRGs of constant *locality*.

Barrier of Applebaum and Lovett. Unfortunately, the time complexity of algebraic algorithms must be subexponential in general. In fact, Applebaum & Lovett proved that the time complexity of an algebraic algorithm deciding if $y \in \{0,1\}^{n^{1+e}}$ lies in the image of a random local function $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ is lower-bounded by $n^{O(n^{1-16 \cdot e/(d-1)})}$ where d is the *rational degree* [3] of the predicate of F [8, Theorem 5.4].

It is an interesting open problem to construct new algebraic algorithms that perform provably faster than the barrier of Applebaum and Lovett and have non-negligible advantage. Note that such new algorithms must avoid computing Macaulay matrices.

[2] Note that such a relationship $h(Y)$ corresponds to a non-trivial element of the elimination ideal $(f_1(X) - Y_1, \ldots, f_m(X) - Y_m) \cap k[Y]$. Since a generating set of this elimination ideal can be computed by a Groebner basis (via an elimination order) we see that algebraic relationships and Macaulay matrices are related. See [39] for a detailed discussion of parallels.

[3] The *rational degree* of a predicate $P : \{0,1\}^{\operatorname{loc} F} \to \{0,1\}$ is defined as the smallest number e s.t. there exist polynomials $Q, R \in \mathbb{Z}_2[X_1, \ldots, X_{\operatorname{loc} F}]$ of degree e s.t. we have $P(X) \cdot Q(X) = R(X) \mod (X_1^2 - X_1, \ldots, X_{\operatorname{loc} F}^2 - X_2)$ and $Q \neq 0$. In other words, P can be written as the rational function $P(X) = \frac{R(X)}{Q(X)}$ of degree e whenever Q does not evaluate to zero.

Attacks on PRGs of Constant Locality. Local PRGs [22] have been subject to cryptoanalysis for a long time. Good surveys on some attacks and on their security can be found In [2,16].

We will try to give here an overview on the most important attacks on local PRGs $F : \{0,1\}^n \to \{0,1\}^m$ of polynomial stretch $m \geq n^{1+e}$ over $\mathbb{Z}_2$. For simplicity, we will assume that F is a random local function, i.e. for computing each output bit it evaluates the same predicate $P : \{0,1\}^{\operatorname{loc} F} \to \{0,1\}$ on a constant set of input bits. The set of input bits depends on the corresponding output bit. The security of random local functions has been extensively discussed by Applebaum [2].

Linear Tests as Sanity Checks. A linear test is a degree-1 polynomial $L \in \mathbb{Z}_2[Y_1, \ldots, Y_m]$ that gets evaluated on the potential output y of a PRG. If $y \leftarrow \{0,1\}^m$ is uniformly random, then the output $L(y)$ is balanced. However, for an image $y = F(x)$, the output $L(y)$ may be biased towards zero or one. In fact, the bias of a PRG $F : \{0,1\}^n \to \{0,1\}^m$ is defined to be the maximum of distinguishing advantages of linear tests of F, and one strives to construct PRGs of negligible bias.

It has been shown that the bias of F depends mainly on the predicate P that is used by F [4]. Further, for random local functions linear tests are a good first check to probe their security: if a random local function has a low bias, it is also secure against a large corpus of other attacks [4,2]. Further, we know of random local functions that have a provably negligible bias: in fact, [32] constructed a predicate of locality 5 and a local PRG $F : \{0,1\}^n \to \{0,1\}^{cn}$ that uses this predicate s.t. the bias of F is provably $2^{-n/O(c^4)}$ (for small choices of $c > 1$).

Further, Viola [36] showed that one can generically create PRGs that are secure against low-degree tests by using PRGs of low bias.

Correlation-Based Attacks. A predicate $P : \{0,1\}^{\operatorname{loc} F} \to \{0,1\}$ is called *c-correlated* if there are c different input variables $X_{i_1}, \ldots, X_{i_c}$ s.t. $P(X) \oplus X_{i_1} \oplus \ldots \oplus X_{i_c}$ is unbalanced (i.e. its probability to evaluate to zero on a random input $x \leftarrow \{0,1\}^{\operatorname{loc} F}$ is not $1/2$). A predicate that has a high correlation is also called *resilient* (when we say "high correlation" we mean it is only c-correlated for large values of c).

Local random functions of low correlation can be efficiently inverted if they have sufficient stretch [2,3,32,9]. In fact, if $F : \{0,1\}^n \to \{0,1\}^m$ stems from a predicate of correlation c, and we have $m \in \Omega(n^{c/2}) + \omega(n)$ then we can deduce from $\Omega(n^{c/2})$ equations of the system $F(X) = y$ a new system of $\Omega(n)$ noisy equations of the shape $X_i \oplus X_j = y'_l$. By using an SDP algorithm [13,21] on this noisy system of locality 2, we can extract an approximation of x. By using this approximation and $\omega(n)$ fresh equations of $F(X) = y$ we can efficiently deduce the correct solution x [9].

Siegenthaler showed that a predicate can either have a high algebraic degree or a high correlation, but not both [35]. In fact, a balanced predicate $P : \{0,1\}^{\operatorname{loc} F} \to \{0,1\}$ must be c-correlated for $c \leq \operatorname{loc} F - \deg F - 1$. This leads to the following important attack: Given a PRG $F : \{0,1\}^n \to \{0,1\}^m$

of locality $d = \operatorname{loc} F$ and stretch $m \in \omega(n^{\lfloor 2d/3 \rfloor/2})$ that stems from a predicate $P : \{0,1\}^d \to \{0,1\}$, we can distinguish two cases: First case, we have $\deg F \leq \lfloor d/3 \rfloor \leq \lfloor 2d/3 \rfloor /2$. In this case, we can invert F by using relinearization (with high probability), since we have $m \in \omega(n^{\lfloor 2d/3 \rfloor/2})$ many equations. Second case, we have $\deg F > \lfloor d/3 \rfloor$. In this case, we have for the correlation $c \leq d - \lfloor d/3 \rfloor - 1$ and, hence, $c \leq \lfloor 2d/3 \rfloor$. In this case, the correlation-based attack can be applied to efficiently invert $F(x)$. It follows that a PRG $F : \{0,1\}^n \to \{0,1\}^m$ of stretch $m \in \omega(n^{\lfloor 2d/3 \rfloor/2})$ and locality d cannot be secure, at all.

O'Donnell and Witmer [33] showed that we really need a stretch of $m \in \omega(n^{\lfloor 2d/3 \rfloor/2})$ and that the correlation-based attack does not degrade gracefully for smaller stretches m. I.e., if the stretch m lies in $o(n^{\lfloor 2d/3 \rfloor/2})$ the above attack does not yield a subexponential attack, since the SDP algorithms have a minimum number of equations they need to use.

Approximation-Based Attacks. Bogdanov and Qiao [9] showed that sufficiently close approximations x' of the seed x can help to invert the function $F(x)$ efficiently. This leads to a subexponential inversion attack on local PRGs $F : \{0,1\}^n \to \{0,1\}^{n^{1+\epsilon}}$. This attack iterates over a set of $2^{O(n^{-\frac{\epsilon}{2 \operatorname{loc} F}})}$ bitstrings x' of length n. With good probability one of those bitstrings x' will be very close to the correct solution x. By using this approximate solution x' one can then invert $F(x) = y$ efficiently and check if the yielded solution is correct.

This leads to an inversion attack with time complexity $2^{O(n^{1-\frac{\epsilon}{2 \operatorname{loc} F}})}$.

Guess-and-Determine Attacks. Couteau, Dupin, Méaux, Rossi & Rotella [16] gave multiple new attacks on PRGs $F : \{0,1\}^n \to \{0,1\}^{n^{1+\epsilon}}$ that they call *guesss-and-determine* attacks. Their attacks are subexponential inversion attacks.

The PRG they attack does not need to be local, but it needs to use a predicate $P : \{0,1\}^{d+\ell} \to \{0,1\}$ of the following form

$$P(X) = M(X_1, \ldots, X_d) \oplus X_{d+1} \oplus \ldots \oplus X_{d+\ell} \tag{30}$$

for some unbalanced predicate $M : \{0,1\}^d \to \{0,1\}$. Note that we have $\deg F \leq d < \operatorname{loc} F$ if F is not to be trivially broken. This is important, since d determines the time complexity of their attack.

Their first attack proceeds as follows: In a so-called *selection phase*, their attack chooses (intelligent and greedily) $n^{1-e/(d-1)}$ input bits s.t. sufficiently many equations of $F(X) = y$ will become linear once these input bits are fixed to constant values. In a second phase, the algorithm iterates over all possible values for these input bits. For each assignment, the algorithm gets a linear equation system in the remaining variables with at least n equations. At this point, the algorithm distinguish two cases: if the resulting matrix has full rank, the algorithm solves the linear equation system and receives a possible assignment for the remaining inputs. Given such a candidate seed x', the algorithm can check if x' is a correct solution for $F(X) = y$. Otherwise, if the yield matrix does

not have full rank, the algorithm can deduce multiple linear equations that does have to hold over $y = F(x)$. If all of these equations do hold over y, then the algorithm knows that with noticeable probability y must lie in the image of F.

The runtime of this algorithm is $2^{O(n^{1-e/(d-1)})}$. Couteau, Dupin, Méaux, Rossi & Rotella could show that this attack has a noticeable advantage at distinguishing the output of F from uniform randomness for predicates $P = M \oplus \bigoplus_{i=1}^{\ell} X_{2+i}$ with $M = X_1 X_2$. For other choices of $M : \{0,1\}^d \to \{0,1\}$, they can only prove a high advantage of their algorithm assuming a specially tailored assumptions which depends on $\operatorname{loc} F$.

Further, they use the notion of *bit fixing algebraic immunity* [29] and show that the selection phase of their algorithm can be used to exploit low bit fixing algebraic immunity of predicates. In fact, if the algebraic immunity of a predicate M deteriorates fast, then instead of trying to extract a linear system the algorithm tries to fix less input bits to extract a polynomial equation system of low degree. This polynomial equation system can then be subexponentially solved by a Groebner basis-based algorithm.

Shrinking-Set Attacks. A *shrinking set* for a local PRG $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ is a subset $T_x \subset [n]$ of input bit positions s.t. the set

$$T_y = \{j \in [m] \mid f_j \text{ only depends on input bits in } T_x\} \tag{31}$$

of output bits that only depend on the input bits specified by T_x is truly larger than T_x, i.e. $\#T_y > \#T_x$. Zichron [38] showed that F must have a shrinking set T_x of size $\#T_x \in \Theta(n^{1-e/(\operatorname{loc} F-1)})$. It can be easily shown that, when we sample $T_x \subset [n]$ uniformly random of size $\Theta(n^{1-e/(\operatorname{loc} F-1)})$, T_y will have more elements than T_x with non-negligible probability.

This leads to the following distinguishing attack [6] on F: sample $T_x \subset [n]$ uniformly at random of size $\Theta(n^{1-e/(\operatorname{loc} F-1)})$. If T_y is larger than T_x, we basically have a smaller PRG

$$F' : \{0,1\}^{\#T_x} \longrightarrow \{0,1\}^{\#T_y} \tag{32}$$

that maps the input bits chosen by T_x to the output bits specified by T_y. Given a string $y \in \{0,1\}^{n^{1+e}}$ and the task to decide if y lies in the image of F, we can instead consider the substring $y' := (y_j)_{j \in T_y}$ and check if y' lies in the image of F'. Indeed, if $y = F(x)$ then $y' = F'(x')$ (for a substring x' of x). Otherwise, if y is uniformly random, then y' will not lie in image of F' with probability at least $1/2$ (since $\#T_y > \#T_x$). Therefore, it suffices to check if y' lies in the image of F'. This can be done by brute-force: we iterate over all possible assignments for the input bits chosen by T_x and see if y' is a possible output of F'.

Since the size of T_x lies in $O(n^{1-e/(\operatorname{loc} F-1)})$, this yields a distinguishing attack of time complexity $O(2^{n^{1-e/(\operatorname{loc} F-1)}})$.

Comparing Runtimes. We will now consider the problem of distinguishing the outputs of a local PRG $F : \{0,1\}^n \to \{0,1\}^m$ of poly-stretch $m \geq n^{1+e}$ from

true randomness. Remember that the attack that we present in this work has a time complexity upper-bounded by

$$n^{O(\log(n)^{1/(\deg F-1)}\cdot n^{1-e/(\deg F-1)})} = 2^{O(\log(n)^{\deg F/(\deg F-1)}\cdot n^{1-e/(\deg F-1)})} \tag{33}$$

and an advantage lower-bounded by

$$1 - O(\log(n)^{1/\deg F}/n^{e/(\deg F-1)}) \geq 1 - o(1). \tag{34}$$

From all attacks listed in this overview, only three have a (potentially) faster runtime for small values of e than this attack, since the runtimes all other attacks depend on the locality of d. Those attacks are the basic algebraic attack that is based on algebraic relationships [39,38], the Groebner basis-based resp. Macaualay matrix-based attacks [20,19,15,14,17,31,37] and the guess-and-determine attacks of Couteau, Dupin, Méaux, Rossi & Rotella [16]. We will compare each of those attacks individually with our attack:

1. The basic algebraic attack [39,38] from Theorem 1 works by finding an algebraic relationship among the polynomials $f_1, \ldots, f_m \in \mathbb{Z}_2[X]$ (without going from $\mathbb{Z}_2$ to a larger field). The runtime of this algorithm is upper-bounded by $n^{O(n^{1-e/(\deg F-1)})} = 2^{O(\log(n)\cdot n^{1-e/(\deg F-1)})}$ which is by the factor $\log(n)^{1/(\deg F-1)}$ in the exponent smaller than the algorithm of Theorem 2. However, in the case of binary PRGs, the advantage of this algorithm is only known to be lower-bounded by a subexponentially small function (note, that for this case the algorithm must be slightly adapted s.t. it retrieves an algebraic relationship that is reduced modulo the field equations of $\mathbb{Z}_2$, the details are described in [39]).
   Hence, while the basic algebraic algorithm is faster than the algorithm of Theorem 2, it only gives an unsatisfactory advantage for breaking the pseudorandomness of binary PRGs.
2. When trying to solve the equation system $F(X) = y$, the algorithms of Faugére [20,19] and the XL-algorithms [15,14,17,31,37] will aim to compute a Groebner basis for the ideal $(f_1(X) - y_1, \ldots, f_m(X) - y_m)$. While this may work well in praxis, in the theoretical very worst case, computing a Groebner basis may have a doubly exponential time complexity.
   In the simpler case of only checking the system of $F(X) = y$ for satisfiability, it suffices to consider algorithms that compute the Macaulay matrix up to some degree D and inspect if they can deduce a contradiction from it. Our results here show that there is some $D \in O(\log(n)^{1/(\deg F-1)} \cdot n^{1-e/(\deg F-1)})$ s.t. it suffices to check the Macaulay matrix up to degree D to have a high distinguishing advantage. It follows that for the Macaulay matrix-based algorithm and the algorithm from Theorem 2, we can give the same bound $2^{O(\log(n)^{\deg F/(\deg F-1)}\cdot n^{1-e/(\deg F-1)})}$ on the time complexity and the same bound $1 - o(1)$ on their advantages. However, following the discussion in [39], while both algorithms have the same asymptotic bounds, it is known that the hidden constants of the algorithm from Theorem 2 are better in the non-uniform model: in fact, if the PRG F is known ahead, the

algebraic relationship h can be computed in a *preprocessing phase*. In the actual security game, the non-uniform adversary then gets the relationship h as a hint and can solve the fresh challenge y by simply evaluating h on it. This is faster than computing the Macaulay matrix of $F(X) = y$ up to degree $D = \deg F \cdot \deg h$.

3. The guess-and-determine attacks of Couteau, Dupin, Méaux, Rossi & Rotella [16] have a time complexity of $2^{O(n^{1-e/(d-1)})}$ for an integer $d \in [\deg F, \operatorname{loc} F - 1]$ that depends on the predicate used by F. In the best cases, we have $d = \deg F$ and the algorithm performs by a factor of $\log(n)^{\deg F/(\deg F-1)}$ in the exponent faster than the algorithm from Theorem 2. However, in the general case, d will be larger than the degree of the predicate of F (for example the predicate $P(X_1, \ldots, X_\ell)$ could be replaced by $P(X_1, \ldots, X_\ell) \oplus P(X_{\ell+1}, \ldots, X_{2\ell})$). Hence, in most cases the guess-and-determine attacks will perform worse than the algorithm from Theorem 2.

Acknowledgements. Before writing this paper I presented its results in a talk at the crypto seminar at NYU. I want to thank the audience for interesting discussions we had while I gave my talk.

2 Preliminaries

2.1 Notation

Denote by $\mathbb{N} = \{1, 2, 3, \ldots\}$ the set of natural numbers.

For a field k, we denote by $k[X_1, \ldots, X_n]$ the ring of polynomials in n variables over k. Sometimes, by abuse of notation, we will just write $k[X] := k[X_1, \ldots, X_n]$. For $f \in k[X]$, we denote by $\deg f$ its degree, that is its *total degree*.

For a prime $p \in \mathbb{N}$, we denote by $\mathbb{Z}_p := \mathbb{Z}/p\mathbb{Z}$ the field with p elements. Further, for $e \in \mathbb{N}$, we denote by $\mathbb{F}_{p^e}$ the Galois field with p^e elements. Up to isomorphism, $\mathbb{F}_{p^e}$ is uniquely determined by its size.

In this text, n will always denote the *security parameter*. An additional parameter is given by the stretch $m = m(n)$ that depends on n. We will always tacitly assume that m is time-constructible.

Let V be some vector space. Given subspaces $V_1, \ldots, V_r \subset V$ that fulfil

$$V_i \cap \left(\sum_{j \neq i} V_j \right) = 0 \tag{35}$$

for each $i \in [r]$, we will denote by $V_1 \oplus \ldots \oplus V_r$ the smallest subspace of V that contains each V_i. I.e., $V_1 \oplus \ldots \oplus V_r$ equals $V_1 + \ldots + V_r$, but by using the $\oplus$ symbol we emphasize that the enumeration of basis vectors of $V_1, \ldots, V_r$ stays linearly independent.

2.2 Mathematical Preliminaries

Definition 1. *Let $k, \overline{k}$ be fields. If there exists a homomorphism of rings $\iota : k \to \overline{k}$, we will call the pair $k, \overline{k}$ a **field extension**. Note that each ring homomorphism must send 1 to 1, therefore each ring homomorphism must be injective on fields. In particular, $\iota : k \to \overline{k}$ is one-to-one and – without loss of generality – we can assume that k is a subset of $\overline{k}$. By abuse of notation, we will denote field extensions always as subset-relationships $k \subset \overline{k}$.*

We will repeat here a well-known fact from algebra and the theory of field extensions.

Lemma 2. *Let $k \subset \overline{k}$ be an extension of finite fields. Then, $k \subset \overline{k}$ is* simple*, i.e., there exists an element $\zeta \in \overline{k}$ s.t. $\overline{k} = k[\zeta]$. I.e., each element of $\overline{k}$ can be written as $f(\zeta)$ where $f \in k[Z]$ is a univariate polynomial.*

Proof (Sketch). First note that the unit group $\overline{k}^\times = \overline{k} \setminus \{0\}$ must be cyclic. Otherwise, there would be a proper divisor $d|(\#\overline{k} - 1)$ s.t. we have $x^d = 1$ for each $x \in \overline{k}^\times$. However, the polynomial $X^d - 1$ can have at most $d < \#(\overline{k}^\times)$ roots in $\overline{k}$. Ergo, there must exist at least one element in $\overline{k}^\times$ of proper order $\#\overline{k} - 1$.

Let $\zeta \in \overline{k}$ be a generator of $\overline{k}^\times$. Then, we have $k[\zeta] = \overline{k}$. In fact, besides zero each element of $\overline{k}$ can be written as a power of ζ.

Note that each simple and finite field extension $k \subset \overline{k}$ that is generated by one element ζ can be written as

$$k[\zeta] = \overline{k} \cong k \oplus \zeta \cdot k \oplus \ldots \oplus \zeta^{r-1} \cdot k \cong k^r \tag{36}$$

where $r = [\overline{k} : k] := \dim_k \overline{k}$ is the *degree* of the extension $k \subset \overline{k}$. I.e., as a k-vector space $\overline{k}$ has the basis $1, \ldots, \zeta^{r-1}$ and each element $c \in \overline{k}$ has a unique representation

$$c = b_1 + b_2 \cdot \zeta + \ldots + b_r \cdot \zeta^{r-1} \tag{37}$$

for $b_1, \ldots, b_r \in k$.

We cite here the general Schwartz-Zippel lemma, which will be implicitly used by Theorem 3.

Lemma 3 (Schwartz-Zippel [34]). *Let k be any field and let $S \subset k$ be a finite set. Let $h \in k[Y_1, \ldots, Y_m]$ be a polynomial in m variables. We have*

$$\Pr_{y \leftarrow S^m}[h(y) = 0] \leq \frac{\deg h}{\#S}. \tag{38}$$

2.3 Cryptographic Preliminaries

In the following, we will revisit the definitions and results from [39] for pseudorandom number generators. For compactness, the definitions here will be less detailed than in [39].

Definition 2 (Pseudorandom Number Generators). *Let k be a field. A **pseudorandom number generator** (PRG) is a deterministic algorithm*

$$F : k^n \longrightarrow k^m \tag{39}$$

that is parametrized by n.

*We call $m = m(n)$ the **stretch** of F. If $m \geq n^{1+e}$ for some constant $e > 0$, we call F a **poly-stretch** PRG.*

Definition 3 (Locality and Degree of PRGs). *Let $F : k^n \to k^m$ be a PRG. Let $f_1, \ldots, f_m : k^n \to k$ be the functions that compute the corresponding output values of F. Note, that each f_i can be computed by a polynomial in $k[X_1, \ldots, X_n]$. In this text, we will always assume that each f_i is in fact a polynomial in $k[X_1, \ldots, X_n]$ that is reduced modulo the field equations of k.*

1. *We define the (algebraic) **degree** of F as the maximum of all degrees of the polynomials computing its output values. I.e.*

$$\deg F := \max_{i \in [m]} (\deg f_i) \tag{40}$$

2. *We define the **locality** of a polynomial $f \in k[X_1, \ldots, X_n]$ as the number of variables that occur non-trivially in f. I.e.*

$$\operatorname{loc} f := \min \{ \#S \mid S \subseteq [n], f \in k[X_i \mid i \in S] \} . \tag{41}$$

*We define the **locality** of F as the maximum of all localities of the polynomials computing its output values. I.e.*

$$\operatorname{loc} F := \max_{i \in [m]} (\operatorname{loc} f_i) \tag{42}$$

Definition 4 (Security Game for PRGs). *Let $F : k^n \to k^m$ be a PRG over a finite field k. The **security-game** for F with an adversary $\mathcal{A}$ is given by:*

1. *A challenger draws a bit $b \leftarrow \{0, 1\}$. If $b = 0$, it samples a preimage $x \leftarrow k^n$ uniformly at random, computes $F(x)$ and sends $(F, F(x))$ to $\mathcal{A}$. If $b = 1$, it samples $y \leftarrow k^m$ and sends (F, y) to $\mathcal{A}$.*
2. *$\mathcal{A}$ receives (F, y^*) for some $y^* \in k^m$ and must decide which bit b has been drawn by the challenger. $\mathcal{A}$ makes some computations on its own and finally sends a bit b' to the challenger.*

$\mathcal{A}$ wins an instance of this game iff $b = b'$ holds at the end. We define $\mathcal{A}$'s advantage against F by

$$\mathsf{adv}_F(\mathcal{A}) := 2 \cdot \Pr[\mathcal{A} \text{ wins}] - 1 \tag{43}$$

$$= \Pr_{x \leftarrow k^n} [\mathcal{A}(F, F(x)) = 0] + \Pr_{y \leftarrow k^m} [\mathcal{A}(F, y) = 1] - 1 \tag{44}$$

where we take the probability over the randomness of $\mathcal{A}$ and the challenger.

Definition 5. *We say that an algorithm is* ***subexponential*** *if there is a constant $e \in [0,1)$ s.t. its time complexity lies in $2^{O(n^e)}$.*

Theorem 3 ([39] Theorem 2). *Let $m \in \omega(n)$. Let $F : k^n \to k^m$ be a PRG of constant degree $\deg F$ over k.*

Then, there is an algebraic attack on F whose time and space complexities are bounded from above by $n^{O((n^{\deg F}/m)^{\frac{1}{\deg F - 1}})}$. Further, the advantage of this attack in the pseudorandomness game of F (Definition 4) is lower bounded by

$$\geq 1 - O\left(\left(n^{\deg F}/m\right)^{\frac{1}{\deg F - 1}} \cdot \frac{1}{\#k}\right). \tag{45}$$

Remark 2. In [39], Theorem 3 has only been stated for fields $k = \mathbb{Z}_p$, however, it is clear that the attack and the proof of Theorem 3 work for any field k (in fact, the algorithm $\mathcal{B}$ that is used as a subroutine of this attack is stated for any field and the Schwartz-Zippel lemma works for any field, too).

While in Definition 4 we only define the security of PRGs with regard to the uniform distribution over k^n, the attack of Theorem 3 works for any distribution of seeds in k^n. The reason is that the attack will always correctly recognize image points $F(x)$ independent of the seed x. However, a random point $y \leftarrow k^m$ will only be refuted by the attack with probability $\geq 1 - O(\left(n^{\deg F}/m\right)^{\frac{1}{\deg F - 1}} \cdot \frac{1}{\#k})$.

3 The New Attack

We will prove here our main result:

Theorem 4. *Let $F : k^n \to k^m$ be a PRG of constant degree $\deg F$ over some field k. Set $r := \max\left(\left\lceil \frac{\log n}{\log \#k} \right\rceil, 1\right)$ and assume $m \in \omega(r \cdot n)$.*

Then, there is an algebraic attack on F whose time and space complexities are bounded from above by $n^{O((\frac{r \cdot n^{\deg F}}{m})^{\frac{1}{\deg F - 1}})}$. Further, the advantage of this attack in the game of Definition 4 against F is lower bounded by

$$\geq 1 - O\left(\left(\frac{r \cdot n}{m}\right)^{1/(\deg F - 1)}\right) \geq 1 - o(1). \tag{46}$$

We will need the following lemma:

Lemma 4. *Let $k \subset \overline{k}$ be a field extension generated by an element $\zeta \in \overline{k}$. Let $r = [\overline{k} : k] = \dim_k \overline{k}$ be the degree of this extension.*

Then, the $m' \times (rm')$ matrix

$$A := I_{m'} \otimes \left(1\ \zeta \ldots \zeta^{r-1}\right) = \begin{pmatrix} 1\ \zeta \ldots \zeta^{r-1} & & \\ & \ddots & \\ & & 1\ \zeta \ldots \zeta^{r-1} \end{pmatrix} \tag{47}$$

gives an isomorphism of k-vector spaces

$$A : k^{rm'} \longrightarrow \overline{k}^{m'} \tag{48}$$

and a degree-preserving isomorphism of k-vector spaces

$$A : (k[X_1, \ldots, X_n])^{rm'} \longrightarrow (\overline{k}[X_1, \ldots, X_n])^{m'} \tag{49}$$

by left-multiplication.

Proof. Note, that $1, \zeta, \ldots, \zeta^{r-1}$ is a basis of $\overline{k}$ as a k-vector space. Indeed, since ζ generates $\overline{k}$ over k, each element of $\overline{k}$ can be written as a polynomial $g(\zeta)$ evaluated on ζ. Since the algebraic degree of ζ over k is r, there must exist a non-zero polynomial $h \in k[Z]$ of degree exactly r that has ζ as root. Therefore, the polynomial representation $g \in k[Z]$ of any element of $\overline{k} = k[\zeta]$ can be reduced modulo h to a polynomial of degree $< r$.

Since each element of $c \in \overline{k}$ has a unique representation as a linear combination

$$c = b_1 + b_2 \cdot \zeta + \ldots + b_r \cdot \zeta^{r-1} \tag{50}$$

for $b_1, \ldots, b_r$, A gives us a linear bijective map from $k^{rm'}$ to $\overline{k}^{m'}$.

Now, if we are given polynomials $f_1, \ldots, f_r \in k[X_1, \ldots, X_n]$ of degree $\leq d$, then $f_1 + f_2 \cdot \zeta + \ldots + f_r \cdot \zeta^{r-1}$ is an element of $\overline{k}[X_1, \ldots, X_n]$. Since scaling and adding polynomials does not increase their degree, the degree of $f_1 + f_2 \cdot \zeta + \ldots + f_r \cdot \zeta^{r-1}$ is at most d (in fact, it equals the maximum of degrees of $f_1, \ldots, f_r$). It follows, that for each $d \in \mathbb{N}$, A gives us a bijective linear map

$$A : (k[X_1, \ldots, X_n]^{\leq d})^{rm'} \longrightarrow (\overline{k}[X_1, \ldots, X_n]^{\leq d})^{m'}. \tag{51}$$

Proof (Theorem 4). Denote by $\mathcal{A}$ the algorithm from Theorem 3. $\mathcal{A}$ is an algorithm that – when given a PRG $G : \overline{k}^n \to \overline{k}^{m'}$ of degree d and a point $y' \in \overline{k}^{m'}$ – will always output 0 if $y' \in G(\overline{k}^n)$ and will output 1 with probability $\geq 1 - O\big((n^d/m')^{\frac{1}{d-1}} \cdot \frac{1}{\#\overline{k}}\big)$ for $y' \leftarrow \overline{k}^{m'}$. The runtime of $\mathcal{A}$ is bounded by $\leq n^{O((n^d/m')^{1/(d-1)})}$.

Let k be a field $F : k^n \to k^m$ be a PRG of degree $d := \deg F$ over k. We can assume that $\#k < n$, since otherwise the claim of the theorem follows directly from Theorem 3. Set $r := \left\lceil \frac{\log n}{\log \#k} \right\rceil = \lceil \log_{\#k} n \rceil$ and $m' := \lfloor \frac{m}{r} \rfloor$.

We will attack the pseudorandomness of F by giving a reduction $\mathcal{R}$ that transforms F to a PRG whose pseudorandomness can be broken by $\mathcal{A}$ with noticeable advantage. On input $F : k^n \to k^m$ and a point $y \in k^m$, $\mathcal{R}$ has to decide if y lies in the image of F or has been sampled uniformly at random. We assumed that the size of k is less than n, so k is finite. $\mathcal{R}$ now constructs an extension field $\overline{k}$ of k s.t. $\overline{k}$ has at least $(\#k)^r \geq n$ elements. In particular, the

algebraic degree of the extension $k \subset \overline{k}$ will be r. According to Lemma 2, there is an element $\zeta \in \overline{k}$ that generates $\overline{k}$ over k, i.e.

$$\overline{k} = k[\zeta]. \tag{52}$$

$\mathcal{R}$ can find an extension field $\overline{k}$ together with a generator ζ by searching for an irreducible polynomial $g \in k[Z]$ of degree r, which can be done in polytime. When given g, $\overline{k}$ is isomorphic to $k[Z]/(g(Z))$ and the residue class of Z corresponds to the generator ζ. Alternatively, if $\mathcal{R}$ already knows a suitable extension field $\overline{k}$ of k, $\mathcal{R}$ can find a generator for this extension by searching for an element $\zeta \in \overline{k}^{\times}$ whose multiplicative order is exactly $\#\overline{k} - 1$. The time complexity of searching for ζ is at most polynomial in the size of $\overline{k}$.

Given $k \subset \overline{k}$ and ζ, $\mathcal{R}$ computes the matrix $A \in \overline{k}^{m' \times (r \cdot m')}$ from Lemma 4. Note that $r \cdot m' \leq m < (r+1) \cdot m'$. Since rm' may be smaller than m, $\mathcal{R}$ constructs a new matrix $B \in \overline{k}^{m' \times m}$

$$B = \begin{pmatrix} A & 0_{m' \times (m - rm')} \end{pmatrix} = \begin{pmatrix} 1\ \zeta \ldots \zeta^{\lceil \log n \rceil - 1} & & 0 \ldots 0 \\ & \ddots & \vdots \quad \vdots \\ & & 1\ \zeta \ldots \zeta^{\lceil \log n \rceil - 1}\ 0 \ldots 0 \end{pmatrix} \tag{53}$$

which consists of A and $m - rm' < r$ columns of zero. Multiplying A with F and y is equivalent to truncating F and y to their first rm' functions resp. outputs and multiplying those with A. Lemma 4 therefore yields that the map

$$G := B \cdot F : k^n \longrightarrow \overline{k}^{m'} \tag{54}$$

$$x \longmapsto B \cdot F(x) \tag{55}$$

is polynomial of degree $d = \deg F$ over $\overline{k}$, and that $y' := B \cdot y$ is uniformly distributed in $\overline{k}^{m'}$ if $y \leftarrow k^m$.

$\mathcal{R}$ therefore computes $y' = B \cdot y$ and polynomials representing $G = B \cdot F$ and submits both to $\mathcal{A}$. We can now consider two cases:

1. If y equals $F(x)$ for some $x \in \{0,1\}^n$, then $y' = B \cdot y$ equals $G(x) = B \cdot F(x)$. In this case, $\mathcal{A}$ will always output 0.
2. If $y \leftarrow k^m$ has been sampled uniformly at random, then $y' = B \cdot y$ is distributed uniformly in $\overline{k}^{m'}$. In this case, $\mathcal{A}$ will output 1 with probability

$$\geq 1 - O(\left(n^d/m'\right)^{\frac{1}{d-1}} \cdot \frac{1}{\#\overline{k}}) \geq 1 - O(\left(r \cdot n^d/m\right)^{\frac{1}{d-1}} \cdot \frac{1}{n}) \tag{56}$$

$$\geq 1 - O((r \cdot n/m)^{\frac{1}{d-1}}) \tag{57}$$

It follows that the advantage of $\mathcal{R}$ is lower-bounded by

$$\mathsf{adv}_F(\mathcal{R}) \geq 1 - O((r \cdot n/m)^{\frac{1}{d-1}}) \tag{58}$$

and hence noticeable.

Further, the time complexity of $\mathcal{A}$ is upper-bounded by

$$\leq n^{O((n^d/m')^{1/(d-1)})} \leq n^{O((r\cdot n^d/m)^{1/(d-1)})}. \tag{59}$$

Hence, $\mathcal{R}$'s time complexity is bounded by $\leq n^{O((r\cdot n^d/m)^{1/(d-1)})}$, too.

Remark 3. If one does not want to use extension fields $\overline{k}$, one can modify the algorithm from Theorem 3 s.t. one receives an algebraic algorithm over the base field k that has the same runtime and advantage bounds as in Theorem 2.

Given a binary PRG $F : \{0,1\}^n \to \{0,1\}^m$, $m \geq n^{1+e}$, the idea is – when running the algorithm from Theorem 3 – to compute a basis for $\ker \phi_D$ where

$$\phi_D : \mathbb{Z}_2[Y_1, \ldots, Y_m] \longrightarrow \mathbb{Z}_2[X_1, \ldots, X_n] \tag{60}$$

$$Y_i \longmapsto f_i \tag{61}$$

for $D \in O((\log n)^{1/(\deg F - 1)} \cdot n^{1-e/(\deg F-1)})$ large enough.

When one needs to decide if a point $y \in \{0,1\}^m$ lies in the image of F, one evaluates *all* basis polynomials of $\ker \phi_D$ on y: if one polynomial in $\ker \phi_D$ does not vanish on y, then we know that y cannot lie in the image of F. Otherwise, if the whole space $\ker \phi_D$ vanishes on y, then we know that with high probability y must lie in the image of F.

The reason for this is that the algebraic relationship $h \in \mathbb{F}_N[Y'_1, \ldots, Y'_{m'}]$, $N = 2^{\lceil \log n \rceil}$, that is yield in the algorithm of Theorem 3 after we went from $F : \{0,1\}^n \to \{0,1\}^m$ to $G : \{0,1\}^n \to \mathbb{F}_N^{m'}$ can be interpreted as $\lceil \log n \rceil$ polynomials in $\mathbb{Z}_2[Y_1, \ldots, Y_m]$ of degree $O((\log n)^{1/(\deg F-1)} \cdot n^{1-e/(\deg F-1)})$.

In fact, the matrix A from Lemma 4 maps the variables $Y_1, \ldots, Y_m$ (which represents binary values) to the variables $Y'_1, \ldots, Y'_{m'}$, $m' = \lfloor m/\lceil \log n \rceil \rfloor$, (which represent extension field values) by

$$Y'_i = Y_{(i-1)\cdot\lceil \log n\rceil+1} + \zeta \cdot Y_{(i-1)\cdot\lceil \log n\rceil+2} + \ldots + \zeta^{\lceil \log n\rceil-1} \cdot Y_{i\cdot\lceil \log n\rceil}. \tag{62}$$

Further, $h \in \mathbb{F}_N[Y'_1, \ldots, Y'_{m'}]$ can be written as

$$h(Y') = \sum_{I \subset \mathbb{N}_0^{m'}, |I| \leq D} (c_{I,1} + \zeta \cdot c_{I,2} + \ldots + \zeta^{\lceil \log n\rceil-1} \cdot c_{I,\lceil \log n\rceil}) \cdot {Y'}^I \tag{63}$$

for coefficients $c_{I,i} \in \mathbb{Z}_2$. By substituting the Y'-variables in h by Y-variables according to Eq. (62) and by sorting the terms in h by powers of ζ, we see that h can be written as

$$h(Y') = u_1(Y) + \zeta \cdot u_2(Y) + \ldots + \zeta^{\lceil \log n\rceil-1} \cdot u_{\lceil \log n\rceil}(Y) \tag{64}$$

for polynomials $u_1, \ldots, u_{\lceil \log n\rceil} \in \mathbb{Z}_2[Y_1, \ldots, Y_m]$. The degree of the polynomials $u_1, \ldots, u_{\lceil \log n\rceil}$ is bounded by the degree of h.

It follows that checking one polynomial over the extension field $\mathbb{F}_N$ is equivalent to checking $\lceil \log n \rceil$ polynomials over the base field $\mathbb{Z}_2$. Hence, if $y \leftarrow \{0,1\}^m$ is truly random, with probability $1 - o(1)$ one polynomial in $\ker \phi_D$ will not vanish on y for D large enough.

4 On the Security of Polynomial and Local PRGs

We try to derive here some insights on parameter choices for PRGs of constant locality and constant degree. By the attacks of this work, we know that the number of *security bits* [4] of a PRG $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ provably lies in

$$O(\log(n)^{\deg F/(\deg F-1)} \cdot n^{1-e/(\deg F-1)}). \tag{65}$$

For simplicity, we will pretend that the number of security bits of F is upper-bounded by

$$n^{1-e/(\deg F-1)}. \tag{66}$$

From a theoretical point of view, this is a very pessimistic view, since we are ignoring the factor $\log(n)^{\deg F/(\deg F-1)}$ and all potentially hidden constants in Eq. (65). However, if we compare the simple theoretical security estimations here with the experimental estimations of Couteau, Dupin, Méaux, Rossi & Rotella [16], we see that their results are far more pessimistic. Hence, we think that the term in Eq. (66) is a justified upper-bound for the number of security bits of F.

While n denotes the number of bits of the seed of F, we will use λ to denote the desired number of security bits. In this view, n gives the number of bits we are ready to invest, while λ denotes the number of bits we are going to get in the end. Optimally, n and λ would be close, however, in our case, it is more suitable to relate them polynomially: by $\delta > 1$ we will denote the *security leverage* and relate λ and n by

$$\lambda^\delta = n. \tag{67}$$

I.e., to achieve a security of λ we will need to invest polynomially more bits. Now, we upper-bound λ by the term in Eq. (66). This leads to the formula

$$1 \leq \delta \cdot \left(1 - \frac{e}{\deg F - 1}\right). \tag{68}$$

By rearranging everything, we get the following lower bound for the degree of F

$$\deg F \geq \frac{\delta}{\delta - 1} \cdot e + 1. \tag{69}$$

Eq. (69) tells us how high we need to set the algebraic degree of F if we want to maintain a security leverage of δ and a stretch of e. Note that for $\delta \to \infty$, the RHS of Eq. (69) converges against $e+1$. In fact, $\deg F > e+1$ is the trivial bound that we always have for $\deg F$ and e, since for $\deg F \leq e+1$ the relinearization attack on F becomes efficient. In other words, the higher our security leverage δ becomes, the less important subexponential attacks on F get. Efficient attacks

[4] By "number of security bits" we mean the logarithm of the average of the concrete number of bit operations a "strong" computational machine needs to run the fastest attacks on PRGs of constant degree.

like relinearization and correlation attacks, however, will always stay relevant and give hard bounds for degree and locality of F.

If F is additionally supposed to have constant locality, we need to take the correlation-based attack from Section 1.3 into account. If c denotes the correlation of F, then it must hold $e < c/2$, since otherwise the correlation-based attack becomes applicable. Since we have $c \leq \operatorname{loc} F - \deg F - 1$, we get $e < (\operatorname{loc} F - \deg F - 1)/2$ and have therefore the minimum bound

$$\operatorname{loc} F > \deg F + 2e + 1 \geq \left(\frac{\delta}{\delta - 1} + 2\right) \cdot e + 2 \tag{70}$$

for the locality of F.

Eq. (69) and Eq. (70) now relate the degree, the locality, the stretch and the security leverage of F. In particular, they tell us lower bounds for the degree and locality of F for given δ and e. For concrete values of δ, we can now list the concrete bounds on $\deg F$ and $\operatorname{loc} F$ in Table 1 [5].

δ	$\deg F$	$\operatorname{loc} F$
1.1	$\geq 11 \cdot e + 1$	$> 13 \cdot e + 2$
1.2	$\geq 6 \cdot e + 1$	$> 8 \cdot e + 2$
1.3	$\geq 13/3 \cdot e + 1$	$> 19/3 \cdot e + 2$
1.4	$\geq 7/2 \cdot e + 1$	$> 11/2 \cdot e + 2$
1.5	$\geq 3 \cdot e + 1$	$> 5 \cdot e + 2$
1.6	$\geq 8/3 \cdot e + 1$	$> 14/3 \cdot e + 2$
1.7	$\geq 17/7 \cdot e + 1$	$> 31/7 \cdot e + 2$
1.8	$\geq 9/4 \cdot e + 1$	$> 17/4 \cdot e + 2$
1.9	$\geq 19/9 \cdot e + 1$	$> 37/9 \cdot e + 2$
2.0	$\geq 2 \cdot e + 1$	$> 4 \cdot e + 2$
2.1	$\geq 21/11 \cdot e + 1$	$> 43/11 \cdot e + 2$
2.2	$\geq 11/6 \cdot e + 1$	$> 23/6 \cdot e + 2$
2.3	$\geq 23/13 \cdot e + 1$	$> 49/13 \cdot e + 2$
2.4	$\geq 12/7 \cdot e + 1$	$> 26/7 \cdot e + 2$
2.6	$\geq 13/8 \cdot e + 1$	$> 29/8 \cdot e + 2$
2.8	$\geq 14/9 \cdot e + 1$	$> 32/9 \cdot e + 2$
3.0	$\geq 3/2 \cdot e + 1$	$> 7/2 \cdot e + 2$
∞	$> e + 1$	$> 3 \cdot e + 2$

Table 1. This table shows for some values of the security leverage $\delta = \log(n)/\log(\lambda)$ corresponding lower bounds for the degree and locality of the PRG $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$. Each row can be read as: "if we want a security of λ bits while our seeds contain $n = \lambda^\delta$ bits, then the degree of the PRG needs to be at least $\frac{\delta}{\delta-1} \cdot e + 1$ and the locality of F needs to be larger than $\left(\frac{\delta}{\delta-1} + 2\right) \cdot e + 2$."

[5] In the initial version of this text, Table 1 contained wrong values (the correct values are a bit more pessimistic). The reason is that I initially computed the values in Table 1 by hand. This time, I used a python script to generate the table. You can find it here: `https://github.com/Semigroup/algebraic-prg-attack-scripts`

If we want, for example, a PRG $F : \{0,1\}^n \to \{0,1\}^{n^2}$ of quadratic stretch, i.e. $e = 1$, then it makes sense to pick a security leverage that is by a little bit larger than 2. If $\delta > 2$, the degree bound $\deg F \geq \frac{\delta}{\delta-1} \cdot e + 1$ is already implied by the trivial bound $\deg F > e + 1 = 2$ that stems from relinearization, since $\frac{\delta}{\delta-1} \cdot e + 1 < 2 \cdot e + 1 = 3$ and $\deg F$ needs to be an integer. On the other hand, for the locality bound $\operatorname{loc} F > \left(\frac{\delta}{\delta-1} + 2\right) \cdot e + 2$ we have $\left(\frac{\delta}{\delta-1} + 2\right) \cdot e + 2 < 4 \cdot e + 2 = 6$. I.e., we can choose $\operatorname{loc} F = 6$, and indeed this bound is also already implied by the trivial bound $\operatorname{loc} F > 3 \cdot e + 2 = 5$.

However, note that a typical security of $\lambda = 128$ security bits would – under a security leverage of $\delta > 2$ – imply a seed of more than 128^2 bits, i.e. 2 kilobytes, and an output size of F of > 4 megabytes.

References

1. Applebaum, B.: Pseudorandom generators with long stretch and low locality from random local one-way functions. In: Karloff, H.J., Pitassi, T. (eds.) 44th ACM STOC. pp. 805–816. ACM Press (May 2012). https://doi.org/10.1145/2213977.2214050
2. Applebaum, B.: Cryptographic hardness of random local functions-survey. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, p. 599. Springer, Heidelberg (Mar 2013). https://doi.org/10.1007/978-3-642-36594-2_33
3. Applebaum, B., Barak, B., Wigderson, A.: Public-key cryptography from different assumptions. In: Schulman, L.J. (ed.) 42nd ACM STOC. pp. 171–180. ACM Press (Jun 2010). https://doi.org/10.1145/1806689.1806714
4. Applebaum, B., Bogdanov, A., Rosen, A.: A dichotomy for local small-bias generators. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 600–617. Springer, Heidelberg (Mar 2012). https://doi.org/10.1007/978-3-642-28914-9_34
5. Applebaum, B., Damgård, I., Ishai, Y., Nielsen, M., Zichron, L.: Secure arithmetic computation with constant computational overhead. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part I. LNCS, vol. 10401, pp. 223–254. Springer, Heidelberg (Aug 2017). https://doi.org/10.1007/978-3-319-63688-7_8
6. Applebaum, B., Ishai, Y., Kushilevitz, E.: On pseudorandom generators with linear stretch in nc0. Comput. Complex. **17**(1), 38–69 (apr 2008). https://doi.org/10.1007/s00037-007-0237-6, https://doi.org/10.1007/s00037-007-0237-6
7. Applebaum, B., Lovett, S.: Algebraic attacks against random local functions and their countermeasures. In: Wichs, D., Mansour, Y. (eds.) 48th ACM STOC. pp. 1087–1100. ACM Press (Jun 2016). https://doi.org/10.1145/2897518.2897554
8. Applebaum, B., Lovett, S.: Algebraic attacks against random local functions and their countermeasures. SIAM Journal on Computing **47**(1), 52–79 (2018). https://doi.org/10.1137/16M1085942, https://doi.org/10.1137/16M1085942
9. Bogdanov, A., Qiao, Y.: On the security of goldreich's one-way function. In: Dinur, I., Jansen, K., Naor, J., Rolim, J. (eds.) Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques. pp. 392–405. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)
10. Buchberger, B.: A theoretical basis for the reduction of polynomials to canonical forms. SIGSAM Bull. **10**(3), 19–29 (aug 1976). https://doi.org/10.1145/1088216.1088219, https://doi.org/10.1145/1088216.1088219

11. Caminata, A., Gorla, E.: Solving multivariate polynomial systems and an invariant from commutative algebra. In: Bajard, J.C., Topuzoğlu, A. (eds.) Arithmetic of Finite Fields. pp. 3–36. Springer International Publishing, Cham (2021)
12. Caminata, A., Gorla, E.: Solving degree, last fall degree, and related invariants. J. Symb. Comput. **114**(C), 322–335 (jan 2023). https://doi.org/10.1016/j.jsc.2022.05.001, https://doi.org/10.1016/j.jsc.2022.05.001
13. Charikar, M., Wirth, A.: Maximizing quadratic programs: Extending Grothendieck's inequality. In: 45th FOCS. pp. 54–60. IEEE Computer Society Press (Oct 2004). https://doi.org/10.1109/FOCS.2004.39
14. Cheng, C.M., Chou, T., Niederhagen, R., Yang, B.Y.: Solving quadratic equations with XL on parallel architectures. In: Prouff, E., Schaumont, P. (eds.) CHES 2012. LNCS, vol. 7428, pp. 356–373. Springer, Heidelberg (Sep 2012). https://doi.org/10.1007/978-3-642-33027-8_21
15. Courtois, N., Klimov, A., Patarin, J., Shamir, A.: Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 392–407. Springer, Heidelberg (May 2000). https://doi.org/10.1007/3-540-45539-6_27
16. Couteau, G., Dupin, A., Méaux, P., Rossi, M., Rotella, Y.: On the concrete security of Goldreich's pseudorandom generator. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 96–124. Springer, Heidelberg (Dec 2018). https://doi.org/10.1007/978-3-030-03329-3_4
17. Ding, J., Buchmann, J., Mohamed, M., Moahmed, W., Weinmann, R.: Mutantxl. SCC pp. 16–22 (01 2008)
18. Dubé, T.W.: The structure of polynomial ideals and gröbner bases. SIAM Journal on Computing **19**(4), 750–773 (1990). https://doi.org/10.1137/0219053, https://doi.org/10.1137/0219053
19. Faugère, J.C.: A new efficient algorithm for computing gröbner bases without reduction to zero (f5). In: Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation. p. 75–83. ISSAC '02, Association for Computing Machinery, New York, NY, USA (2002). https://doi.org/10.1145/780506.780516, https://doi.org/10.1145/780506.780516
20. Faugére, J.C.: A new efficient algorithm for computing gröbner bases (f4). Journal of Pure and Applied Algebra **139**(1), 61–88 (1999). https://doi.org/https://doi.org/10.1016/S0022-4049(99)00005-5, https://www.sciencedirect.com/science/article/pii/S0022404999000055
21. Goemans, M.X., Williamson, D.P.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. J. ACM **42**(6), 1115–1145 (nov 1995). https://doi.org/10.1145/227683.227684, https://doi.org/10.1145/227683.227684
22. Goldreich, O.: Candidate One-Way Functions Based on Expander Graphs, pp. 76–87. Springer Berlin Heidelberg, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22670-0_10, https://doi.org/10.1007/978-3-642-22670-0_10
23. Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM Journal on Computing **28**(4), 1364–1396 (1999)
24. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Cryptography with constant computational overhead. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC. pp. 433–442. ACM Press (May 2008). https://doi.org/10.1145/1374376.1374438
25. Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from well-founded assumptions. In: Proceedings of the 53rd Annual ACM SIGACT Symposium on

Theory of Computing. p. 60–73. STOC 2021, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3406325.3451093, https://doi.org/10.1145/3406325.3451093
26. Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from lpn over fp, dlin, and prgs in nc0. In: Dunkelman, O., Dziembowski, S. (eds.) Advances in Cryptology – EUROCRYPT 2022. pp. 670–699. Springer International Publishing, Cham (2022)
27. Lazard, D.: Gröbner bases, gaussian elimination and resolution of systems of algebraic equations. In: van Hulzen, J.A. (ed.) Computer Algebra. pp. 146–156. Springer Berlin Heidelberg, Berlin, Heidelberg (1983)
28. Macaulay, F.: The algebraic theory of modular systems. Cambridge Mathematical Library **xxxi** (1916)
29. Méaux, P., Journault, A., Standaert, F.X., Carlet, C.: Towards stream ciphers for efficient FHE with low-noise ciphertexts. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part I. LNCS, vol. 9665, pp. 311–343. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49890-3_13
30. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (Apr 2012). https://doi.org/10.1007/978-3-642-29011-4_41
31. Mohamed, M.S.E., Mohamed, W.S.A.E., Ding, J., Buchmann, J.A.: MXL2: Solving polynomial equations over GF(2) using an improved mutant strategy. In: Buchmann, J., Ding, J. (eds.) Post-quantum cryptography, second international workshop, PQCRYPTO 2008. pp. 203–215. Springer, Heidelberg (Oct 2008). https://doi.org/10.1007/978-3-540-88403-3_14
32. Mossel, E., Shpilka, A., Trevisan, L.: On e-biased generators in NC0. In: 44th FOCS. pp. 136–145. IEEE Computer Society Press (Oct 2003). https://doi.org/10.1109/SFCS.2003.1238188
33. O'Donnell, R., Witmer, D.: Goldreich's prg: Evidence for near-optimal polynomial stretch. pp. 1–12 (06 2014). https://doi.org/10.1109/CCC.2014.9
34. Schwartz, J.T.: Fast probabilistic algorithms for verification of polynomial identities. J. ACM **27**(4), 701–717 (oct 1980). https://doi.org/10.1145/322217.322225, https://doi.org/10.1145/322217.322225
35. Siegenthaler, T.: Correlation-immunity of nonlinear combining functions for cryptographic applications (corresp.). IEEE Transactions on Information Theory **30**(5), 776–780 (1984). https://doi.org/10.1109/TIT.1984.1056949
36. Viola, E.: The sum of d small-bias generators fools polynomials of degree d. In: 2008 23rd Annual IEEE Conference on Computational Complexity. pp. 124–127 (2008). https://doi.org/10.1109/CCC.2008.16
37. Yang, B.Y., Chen, J.M.: All in the xl family: Theory and practice. In: Park, C.s., Chee, S. (eds.) Information Security and Cryptology – ICISC 2004. pp. 67–86. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)
38. Zichron, L.: Locally computable arithmetic pseudorandom generators (2017), https://www.bennyapplebaum.sites.tau.ac.il/_files/ugd/f706bf_501515c9cd7744c498935684bd1648a2.pdf
39. Ünal, A.: Worst-case subexponential attacks on prgs of constant degree or constant locality. Cryptology ePrint Archive, Paper 2023/119 (2023), https://eprint.iacr.org/2023/119, https://eprint.iacr.org/2023/119