Worst-Case Subexponential Attacks on PRGs of Constant Degree or Constant Locality

Akın Ünal[0000−0002−8929−0221]

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

Abstract. In this work, we will give new attacks on the pseudorandomness of algebraic pseudorandom number generators (PRGs) of polynomial stretch. Our algorithms apply to a broad class of PRGs and are in the case of general local PRGs faster than currently known attacks. At the same time, in contrast to most algebraic attacks, subexponential time and space bounds will be proven for our attacks without making any assumptions of the PRGs or assuming any further conjectures. Therefore, we yield in this text the first subexponential distinguishing attacks on PRGs from constant-degree polynomials and close current gaps in the subexponential cryptoanalysis of lightweight PRGs.
Concretely, against PRGs $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ that are computed by polynomials of degree d over a field $\mathbb{Z}_q$ and have a stretch of $m = n^{1+e}$ we give an attack with space and time complexities $n^{O(n^{1-\frac{e}{d-1}})}$ and noticeable advantage $1 - O(n^{1-\frac{e}{d-1}}/q)$. If q lies in $O(n^{1-\frac{e}{d-1}})$, we give a second attack with the same space and time complexities whose advantage is at least $q^{-O(n^{1-\frac{e}{d-1}})}$. If F is of constant *locality* d and q is constant, we construct a third attack that has a space and time complexity of $\exp(O(n^{1-\frac{e'}{(q-1)d-1}}))$ and noticeable advantage $1 - O(n^{-\frac{e'}{(q-1)d-1}})$ for every constant $e' < e$.

1 Introduction

A pseudorandom number generator (PRG) is a deterministic algorithm $F : \{0,1\}^n \to \{0,1\}^m$ that stretches a given string of bits i.e. $m > n$. We expect a PRG to expand a uniformly drawn string to a longer string of bits that sufficiently simulates randomness. More formally, for a PRG F its output – when evaluated on a short uniformly random string – should be for a certain class of computational models indistinguishable from a longer uniformly random string, even if the algorithm F is publicly known.

PRGs are an important tool in the toolbox of cryptography besides one-way functions [1, 27], pseudorandom permutations and pseudorandom functions. Further, in complexity theory, the existence of PRGs implies the derandomization of certain complexity classes [38]. For example, it is known that the existence of

so-called *high-end* PRGs implies that **P** equals **BPP** [29]. Additionally, PRGs have the real world task of simulating cryptographic pseudorandomness in deterministic software applications.

Of particular interest are PRGs that can be efficiently evaluated. Very prominent examples are *local* PRGs [26]. Each output bit of a local PRG depends on only a constant number of input bits. Besides their simplicity, local PRGs are an important building block in advanced cryptographic constructions, e.g. two-party protocols for computing circuits with constant overhead [30] or indistinguishability obfuscation [31, 32]. Assuming additionally the pseudorandomness of arithmetic PRGs $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ where each output value is computed by a polynomial of constant degree over $\mathbb{Z}_q$ leads to arithmetization of such primitives, like e.g. arithmetic two-party protocols [4].

Since PRGs play such a crucial role in cryptography, cryptoanalysis of PRGs is of general importance. In particular, local PRGs $F : \{0,1\}^n \to \{0,1\}^m$ of poly-stretch, i.e. $m \geq n^{1+e}$ for some constant $e > 0$, have been the subject of various attacks, and it could be shown that such PRGs can be distinguished by subexponential-size [1] circuits, or even poly-size circuits if $e > 0.5$ [3, 5, 11, 18, 39, 41].

PRGs of constant degree, i.e. PRGs that can be computed by polynomials of constant degree over some finite field, can be seen as a generalization of local PRGs. However, constant-degree PRGs have received much less attention in cryptoanalytic literature than local PRGs. While there is a huge collection of algebraic attacks on refuting and inverting constant-degree PRGs like F4/F5 and the XL-algorithms [12, 16, 17, 23, 24, 36, 44], we do not know of any attacks whose time-complexity for poly-stretch constant-degree PRGs is guaranteed to be subexponential even in the worst case. We intend to close this gap by introducing a new algebraic attack that is provably subexponential against poly-stretch PRGs of constant degree.

1.1 Contribution

In this text, we will introduce new algebraic attacks on PRGs and prove upper bounds for their complexities and lower bounds for their advantage in the worst case. Let $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ with $m \geq n^{1+e}$. Then, we give the following attacks on the pseudorandomness of F:

- If F is of degree d over $\mathbb{Z}_q$, we have an attack with subexponential space and time complexities $n^{O(n^{1-\frac{e}{d-1}})}$. The advantage of this attack is $1 - O(n^{1-\frac{e}{d-1}}/q)$, which is noticeable if q is large enough.
- If q should be small (e.g. $q \in O(n^{1-\frac{e}{d-1}})$), then we give a second attack in the above case with the same space and time complexities for which we can guarantee a subexponentially small advantage of $q^{-O(n^{1-\frac{e}{d-1}})}$.

[1] The notion of *subexponentiality* is ambiguous in literature. Here, we denote by subexponential a function that is contained in $\bigcup_{c<1} 2^{O(n^c)}$.

– If q is constant and F is of *locality* d, we give for each constant $e' \in [0, e)$ a third attack with subexponential space and time complexities $2^{O(n^{1-\frac{e'}{(q-1)d-1}})}$. For this attack, we will prove a noticeable advantage of $1 - O(n^{-\frac{e'}{(q-1)d-1}})$.

To the best of our knowledge, we give the first distinguishing algorithms on constant-degree PRGs that are provably subexponential in the worst case for sufficiently large moduli. Additionally, our second and third attack algorithms are faster than the attacks of Bogdanov & Qiao [11]. Hence, these attacks give new baselines for the cryptoanalysis of local PRGs.

1.2 Technical Overview

We want to motivate and explain here the ideas behind our new attacks. Let q be a prime, $\mathbb{Z}_q$ be the finite field of size q and $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ be a PRG of degree d. I.e., the i-th output value of F is computed by a polynomial $f_i \in \mathbb{Z}_q[X] := \mathbb{Z}_q[X_1, \ldots, X_n]$ of total degree $\leq d$. Now, assume we would know a non-zero polynomial $h \in \mathbb{Z}_q[Y] := \mathbb{Z}_q[Y_1, \ldots, Y_m]$ that vanishes on the image of F i.e.

$$h(F(x)) = 0 \tag{1}$$

for all $x \in k^n$. Let D be the total degree of h. Since h is not the zero polynomial, we have according to the famous Schwartz-Zippel lemma [40]

$$\Pr_{y \leftarrow \mathbb{Z}_q^m}[h(y) = 0] \leq \frac{D}{q}. \tag{2}$$

I.e., while h will always be zero on the image of F, the probability that h vanishes on a random point can be controlled by D/q. If D is sublinear and q is sufficiently large, $q \geq n$ for example, h gives us a strong indicator for distinguishing image points of F from random points of $\mathbb{Z}_q^m$. In fact, by using h we can distinguish the distribution $(F(x))_{x \leftarrow \mathbb{Z}_q^n}$ from $(y)_{y \leftarrow \mathbb{Z}_q^m}$ with advantage at least $1 - \frac{D}{q}$.

However, the following two questions remain:

1. For which degrees D can we guarantee the existence of a non-zero polynomial h of degree D that vanishes on the image of F?
2. Even if we know that such a polynomial h must exist, how can we algorithmically compute it?

Finding Algebraic Relations. The set of polynomials h that vanish on each $F(x)$ has a specific algebraic structure. To explore this structure, we consider the following morphism of $\mathbb{Z}_q$-algebras:

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

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

ϕ maps polynomials in $\mathbb{Z}_q[Y]$ to polynomials in $\mathbb{Z}_q[X]$ by substituting each variable Y_i by the polynomial $f_i(X)$. Denote by $\ker \phi$ the kernel of ϕ, i.e.

$$\ker \phi = \{g \in \mathbb{Z}_q[Y] \mid \phi(g) = 0\}. \tag{5}$$

If g lies in $\ker \phi$, we have $\phi(g) = g(f_1(X), \ldots, f_m(X)) = 0$. In particular, we have for each $x \in \mathbb{Z}_q^n$ then

$$g(f_1(x), \ldots, f_m(x)) = \phi(g)(x_1, \ldots, x_m) = 0. \tag{6}$$

This means, the kernel of ϕ contains polynomials h that are of interest for us.

Therefore, we can restate our questions as follows:

1. For what D can we guarantee the existence of a non-zero element of $\ker \phi$?
2. How can we compute all elements of $\ker \phi$ up to degree D?

To answer the first question, we define the following $\mathbb{Z}_q$-vector spaces for $\ell \in \mathbb{N}$:

$$\mathbb{Z}_q[X]^{\leq \ell} := \{g \in \mathbb{Z}_q[X] \mid \deg g \leq \ell\}, \tag{7}$$

$$\mathbb{Z}_q[Y]^{\leq \ell} := \{g \in \mathbb{Z}_q[Y] \mid \deg g \leq \ell\}. \tag{8}$$

The vector spaces $\mathbb{Z}_q[X]^{\leq \ell}$ and $\mathbb{Z}_q[Y]^{\leq \ell}$ contain all elements of $\mathbb{Z}_q[X]$ resp. $\mathbb{Z}_q[Y]$ of total degree $\leq \ell$. They are spanned by all monomials in the X- resp. Y-variables of degree $\leq \ell$. Therefore, we have

$$\dim_{\mathbb{Z}_q} \mathbb{Z}_q[X]^{\leq \ell} = \binom{n+\ell}{\ell} \quad \text{and} \quad \dim_{\mathbb{Z}_q} \mathbb{Z}_q[Y]^{\leq \ell} = \binom{m+\ell}{\ell}. \tag{9}$$

Now, we want to restrict ϕ on $\mathbb{Z}_q[Y]^{\leq \ell}$. Remember that F is a PRG of degree d, i.e., each f_i is a polynomial of degree d. It is easy to see that ϕ stretches the degree of each polynomial by at most a factor of d. I.e., we have for each $g \in \mathbb{Z}_q[Y]$

$$\deg \phi(g) = \deg g(f_1(X), \ldots, f_m(X)) \leq d \cdot \deg g. \tag{10}$$

So, by restricting ϕ on $\mathbb{Z}_q[Y]^{\leq \ell}$, we get a linear map

$$\phi^\ell : \mathbb{Z}_q[Y]^{\leq \ell} \longrightarrow \mathbb{Z}_q[X]^{\leq d \cdot \ell} \tag{11}$$

for each ℓ. For linear maps, it is quite easy to guarantee the existence of non-trivial kernel elements. In fact, by dimension formulas, we have

$$\dim_{\mathbb{Z}_q} \ker \phi^\ell \geq \dim_{\mathbb{Z}_q}(\mathbb{Z}_q[Y]^{\leq \ell}) - \dim_{\mathbb{Z}_q}(\mathbb{Z}_q[X]^{\leq d \cdot \ell}) \tag{12}$$

$$= \binom{m+\ell}{\ell} - \binom{n+d\cdot\ell}{d\cdot\ell}. \tag{13}$$

Therefore, it suffices to find the smallest D s.t.

$$\binom{m+D}{D} > \binom{n+d\cdot D}{d\cdot D}. \tag{14}$$

As we already stated, we are interested here in PRGs of poly-stretch, so let $e > 0$ be constant s.t. $m \geq n^{1+e}$. We claim that inequality Eq. (14) holds for $D \in \Omega(n^{1-\frac{e}{d-1}})$. To see this, note that we have

$$\binom{m+D}{D} > \binom{n+d\cdot D}{d\cdot D} \tag{15}$$

$$\iff \frac{(m+D)\cdots(m+1)}{D\cdots 1} > \frac{(n+dD)\cdots(n+1)}{(dD)\cdots 1} \tag{16}$$

$$\iff (m+D)\cdots(m+1)\cdot(dD)\cdots(D+1) > (n+dD)\cdots(n+1). \tag{17}$$

To show Eq. (17), we lower bound the LHS terms $(dD)\cdots(D+1) > D^{(d-1)D}$ and $(m+D)\cdots(m+1) > m^D$. Further, for the simplicity of this exposition, we approximate $(n+dD)\cdots(n+1)$ by n^{dD}. We then get roughly

$$(m+D)\cdots(m+1)\cdot(dD)\cdots(D+1) \tag{18}$$

$$> m^D \cdot D^{(d-1)D} \tag{19}$$

$$\geq n^{(1+e)D} \cdot n^{(1-\frac{e}{d-1})\cdot(d-1)D} \tag{20}$$

$$= n^{(1+e)D+(d-1-e)D} \tag{21}$$

$$= n^{dD} \approx (n+dD)\cdots(n+1). \tag{22}$$

This shows that the degree $D \in \Omega(n^{1-\frac{e}{d-1}})$ is a plausible bound for non-trivial elements in $\ker\phi$. In Section 3, we will show that we can choose any $D \geq c{\cdot}n^{1-\frac{e}{d-1}}$ for a constant $c \in (2,4]$ that depends on d.

The above considerations also give us a straight-forward algorithm for computing a non-zero element $h \in \ker\phi$: For each $\ell = 1,\ldots,D$, we compute a matrix representation of the linear map

$$\phi^\ell : \mathbb{Z}_q[Y]^{\leq \ell} \longrightarrow \mathbb{Z}_q[X]^{\leq d\cdot\ell}. \tag{23}$$

By using Gaussian elimination, we can then check if this matrix has a non-trivial kernel vector. Such a non-trivial kernel vector corresponds to a non-trivial kernel element $h \in \ker\phi$ of degree ℓ. By our observations above, we know that for $\ell = D = c\cdot n^{1-\frac{e}{d-1}}$, this algorithm must eventually find a non-zero polynomial.

The space and time complexities of this algorithm is in each step dominated by computing the Gaussian elimination of a matrix of shape $M_\ell \times N_\ell$ where $M_\ell = \binom{m+\ell}{\ell} \in O(n^{(1+e)\ell})$ and $N_\ell = \binom{n+d\ell}{d\ell} \in O(n^{d\ell})$. Therefore, we need to store $M_D{\cdot}N_D \in n^{O(n^{1-\frac{e}{d-1}})}$ field elements and perform $D{\cdot}M_D{\cdot}N_D^2 \in n^{O(n^{1-\frac{e}{d-1}})}$ arithmetic operations in $\mathbb{Z}_q$.

Evaluating h on a point $y \in \mathbb{Z}_q^m$ costs $D\cdot M_D \in n^{O(n^{1-\frac{e}{d-1}})}$ field operations. The advantage of using h in distinguishing a random point from an image point of F is at least $1 - D/q$. Hence, for $q \in \omega(n^{1-\frac{e}{d-1}})$ and $m \geq n^{1+e}$, we have an attack algorithm with noticeable advantage, which is subexponential in the worst case.

We give a detailed description of the algorithms sketched here and formal proofs for their correctness in Section 3 and Section 4.

Handling Small Moduli. Note, that we cannot guarantee any advantage of the above algorithm if $q \leq D = cn^{1-\frac{e}{d-1}}$. In fact, it may be that the above algorithm will retrieve the kernel element $h(Y) = Y_1^q - Y_1 \in \ker\phi$ in this case, which is not helpful since the polynomial $Y_1^q - Y_1$ vanishes on each point $y \in \mathbb{Z}_q^m$.

We can prevent the appearance of trivial polynomials in the kernel of ϕ as follows: instead of $\mathbb{Z}_q[X]$ and $\mathbb{Z}_q[Y]$ we consider the rings

$$R_q[X] := \mathbb{Z}_q[X]/(X_1^q - X_1, \ldots, X_n^q - X_n) \tag{24}$$

$$R_q[Y] := \mathbb{Z}_q[Y]/(Y_1^q - Y_1, \ldots, Y_m^q - Y_m). \tag{25}$$

On these rings, we have a morphism

$$\phi_q : R_q[Y] \longrightarrow R_q[X] \tag{26}$$

$$g(Y_1, \ldots, Y_m) \longmapsto g(f_1(X), \ldots, f_m(X)) \tag{27}$$

that again maps each variable Y_i to the polynomial f_i. The non-zero elements of $\ker \phi_q$ are now exactly the polynomials h that vanish on the image of F, but not everywhere on $\mathbb{Z}_q^m$. If we restrict ϕ_q to $R_q[Y]^{\leq \ell}$ we get again a linear map

$$\phi_q^\ell : R_q[Y]^{\leq \ell} \longrightarrow R_q[X]^{\leq d\ell}, \tag{28}$$

and it can be shown again that we have for $D \geq c \cdot n^{1-\frac{e}{d-1}}$ for some constant c

$$\dim_{\mathbb{Z}_q} R_q[Y]^{\leq D} > \dim_{\mathbb{Z}_q} R_q[X]^{\leq dD}. \tag{29}$$

Therefore, $\ker \phi_q$ must contain a non-zero element h of sublinear degree D that is not contained in the ideal $(Y_1^q - Y_1, \ldots, Y_m^q - Y_m)$. For such an element, it can be shown that its probability to not vanish on a random point can be subexponentially bounded, even if $D > q$. I.e.

$$\Pr_{y \leftarrow \mathbb{Z}_q^m}[h(y) \neq 0] \geq q^{-D}. \tag{30}$$

This gives us an algorithm of subexponential complexity with a subexponentially small advantage in distinguishing between random points and images of the PRG F.

In a multi-challenge setting, where the adversary can query multiple values $y_1, \ldots, y_Q$ that either are all uniformly and independently random or are all values in the image of F, the above attack can be amplified to have a noticeable advantage for a subexponential number $Q \in q^{\Omega(n^{1-\frac{e}{d-1}})}$ of challenges.

Local PRGs of Constant Moduli. While the advantage of the above attack may be much higher in practice (since the probability that h vanishes on a random point may be higher than q^{-D}), from a theoretical point of view the postulated subexponential advantage is not satisfying.

Fortunately, in the case where the modulus q is constant and F is of constant locality, we can use a little trick to noticeably boost the advantage of our attack. For simplicity, we will assume here that q is 2, however the following approach works for each constant modulus:

Let $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ be of locality d. This means, the i-th bit of the output of F is computed by a function $f_i : \mathbb{Z}_2^n \to \mathbb{Z}_2$ that only depends on d of its inputs. Choose a prime number $p \in [n, 2n]$ and note that – due to the locality of F – for each f_i we can find a polynomial $f_i' \in \mathbb{Z}_p[X]$ of degree d that coincides[2] with f_i on

[2] Note, that in the case $q > 2$, the degree of the polynomials f_i' that coincide with f_i on the set $\{0, \ldots, q-1\}^n$ will be $(q-1)d$ instead of d.

$\{0,1\}^n$, i.e., we have $f'_i(x) = f_i(x)$ for each $x \in \{0,1\}^n$. So, instead of attacking the pseudorandomness of F, we can focus on the pseudorandomness of the map $F' : \mathbb{Z}_p^n \to \mathbb{Z}_p^m$ of degree d that consists of the polynomials $f'_1, \ldots, f'_m$. However, distinguishing a random point $y \leftarrow \mathbb{Z}_p^m$ from $F'(x) = F(x)$, for $x \leftarrow \{0,1\}^n$, is obviously simple, since the latter will always lie in $\{0,1\}^m$. To come up for that, we set $m' := \frac{m}{3 \log p}$ and draw a uniformly random matrix $A \leftarrow \mathbb{Z}_p^{m' \times m}$. According to the Leftover Hash Lemma, the distributions

$$(A, Ay)_{y \leftarrow \{0,1\}^m} \quad \text{and} \quad (A, y')_{y' \leftarrow \mathbb{Z}_p^{m'}} \tag{31}$$

are statistically very close. Therefore, if $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ is pseudorandom, then the map $G : \mathbb{Z}_p^n \to \mathbb{Z}_p^{m'}$ of degree d that maps x to $A \cdot F'(x)$ must be, too. However, we can apply our first attack against G. Since $m \geq n^{1+e}$, we have $m' \geq n^{1+e'}$ for any constant $e' < e$ and therefore an attack of time and space complexity $2^{O(n^{1-\frac{e'}{d-1}})}$ and noticeable advantage $1 - O(n^{1-\frac{e'}{d-1}})/p \geq 1 - O(n^{-\frac{e'}{d-1}})$. Going back to F, we get an algorithm of subexponential complexity that has a noticeable advantage in distinguishing images of F from random bit strings $y \leftarrow \{0,1\}^m$. We detail this attack in Section 5.

1.3 Related Work

We try to give here a short survey of the current cryptoanalytic literature on PRGs.

Linear Tests and Low-Degree Correlation. A *linear test* for a PRG $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ is a degree-1 polynomial $L \in \mathbb{Z}_q[Y]$ that has a noticeable advantage

$$\left| \Pr_{x \leftarrow \mathbb{Z}_q^n}[L(F(x)) = 0] - \Pr_{y \leftarrow \mathbb{Z}_q^m}[L(y) = 0] \right| \tag{32}$$

in distinguishing random points from image points of F. While linear tests form a very simple class of attacks against PRGs, it can be shown that they are a good sanity check in the case of local PRGs: a local random PRG that is secure against linear tests also fools other classes of distinguishers like e.g. $\mathbf{AC}^0$, l-wise tests and degree-2 threshold functions [2, Proposition 4.10]. Mossel *et al.* [37] shows that there exist PRGs of constant locality s.t. each linear test only has negligible advantage against those PRGs, even if the PRG is of polynomial stretch $m = n^{1+e}$. Their construction is based on the famous tri-sum-and predicate

$$X_1 X_2 + X_3 + X_4 + X_5 \tag{33}$$

that gets applied on random subsets of the input to compute the output bits of the PRG.

If we allow the degree of L to be greater than 1, we get a polynomial test of higher degree. Viola [43] showed that for each constant d a PRG can be constructed that cannot be distinguished by degree-d tests with noticeable advantage (his constructions allows non-constant values for d, however such d reduce the stretch of the PRG substantially).

Groebner Basis-Based Attacks. A huge class of attacks against PRGs of constant degree constitute of algebraic attacks [12, 16, 17, 23, 24, 36, 44]. These attacks aim to invert the potential image of a PRG by computing a Groebner basis or something similar in the case of XL-algorithms.

These algorithms work well in practice, and it has been suspected that they give subexponential attack algorithms against PRGs of polynomial stretch [9]. However, computing a Groebner basis can be a task of double exponential complexity in the worst case, and therefore those algorithms do not give us provable subexponential attacks.

In the full version [45] of this text, we will give a deeper comparison of our algorithms with Groebner basis-based algorithms, draw new insights for Groebner basis-based algorithms and construct a Macaulay matrix-based algorithm that is provable subexponential in the worst case when distinguishing the images of poly-stretch PRGs.

Random Local Functions. A *random local function* is a PRG $F : \{0,1\}^n \to \{0,1\}^m$ where each output bit is computed by a fixed predicate $P : \{0,1\}^d \to \{0,1\}$ that is applied on a random subset of bits of the input string. The notion of random local functions has been put forth by Goldreich [26] and was the subject of a great body of cryptoanalytic literature. For exhaustive surveys and studies on the security of random local functions, we refer the reader to the works of Applebaum [2] and Couteau *et al.* [18]. We will only review here some attacks on random local functions, which we think are the most relevant for the context of this work:

1. It is known that F can be inverted in polynomial time and with high probability if $m \in \Omega(\log(n) \cdot n^{\frac{\lfloor 2d/3 \rfloor}{2}})$ [2]. First note, that F can be efficiently inverted by linearization of the corresponding polynomial equation system if it is of stretch $m \in \Omega(n^{\deg P})$, where $\deg P$ denotes the degree of P as a polynomial over $\mathbb{Z}_2$.
   This means, the degree of P must be greater than $d/3$ if we want to avoid the above attack for $m \geq n^{\frac{d}{3}}$. However, if $\deg P \geq d/3$, then P is correlated with the sum of $c \leq d - \frac{d}{3}$ of its variables [41]. I.e., P can be written as
   $$P(Z_1, \ldots, Z_d) = Z_1 + \ldots + Z_c + N(Z_1, \ldots, Z_d) \tag{34}$$
   where N is a biased predicate i.e. $\Pr_{z \leftarrow \{0,1\}^d}[N(z) = 0] \neq \frac{1}{2}$. When solving the system $F(x) = y$, one can see the N predicates as dependent noise added to linear equations. This constrained noisy linear equation system can be solved efficiently if $m \in \Omega(n^{c/2})$ [15, 25].
2. There is a subexponential inversion attack [2, 11] on $F(x)$ that utilizes approximations of the correct inverse and has a runtime complexity of $2^{O(n^{1-\frac{e}{2d}})}$ (if $m \geq n^{1+e}$). The idea is to assign random bits to the first $(1-2n^{-\frac{e}{2d}})$ bits of an approximate solution. By iterating over all possible $x' \in \{0,1\}^n$ with the given prefix, one will find an approximation that coincides with x on at least $(\frac{1}{2} + n^{-\frac{e}{2d}})n$ of its bits with probability at least $\frac{1}{2}$. This approximation

can now be used to find efficiently and with high probability the correct solution x.
Note, that the time complexity $2^{O(n^{1-\frac{e}{2d}})}$ of this algorithm is worse than the time complexity $2^{O(n^{1-\frac{e'}{d-1}})}$, for any constant $e' < e$, of the algorithm we sketched against d-local PRGs of stretch n^{1+e}.

3. Couteau *et al.* [18] constructed a guess-and-determine-style attack on PRGs $F : \{0,1\}^n \to \{0,1\}^{n^{1+e}}$ of constant locality. Their attack guesses – in an intelligent way – a portion of the bits of x and tries to extract a linear equation system from the system $F(x) = y$ for the unguessed input bits. If the predicate P for F is of the form

$$P(X_1, \ldots, X_r) = X_1X_2 + X_3 + \ldots + X_r, \tag{35}$$

they can prove that their attack will succeed in distinguishing random points from images of F and has a time complexity of $2^{O(n^{1-e})}$. Note, that r does not need to be constant.
They even generalize their attack to work with general predicates

$$P(X_1, \ldots, X_r) = M(X_1, \ldots, X_d) + X_{d+1} + \ldots + X_r, \tag{36}$$

for any predicate $M : \{0,1\}^d \to \{0,1\}$ and get an attack algorithm of time complexity $2^{O(n^{1-\frac{e}{d-1}})}$. However, to prove a high success probability of the generalization of their attack they need to assume a special conjecture that depends on M.
4. While there are a lot of efficient attacks against local PRGs of sufficient stretch, it is known that algebraic attacks against d-local PRGs of stretch n^{1+e} will have a time complexity of at least $n^{O(n^{1-32\frac{e}{d-2}})}$ in the worst case [2, Theorem 5.5]. This means, up to some constants in the exponent, the time complexities we achieve with our attacks are optimal for algebraic attacks.

Attacks Based on Sum-of-Squares. Sum-of-Squares attacks are a special class of SDP-based attacks. These attacks were discovered recently and used to refute several candidate light-weight PRGs of polynomial stretch for indistinguishability obfuscation schemes [7, 8]. While these attacks are efficient, they need to make special assumptions about the PRGs they attack, which limits the generality of those attacks. We will list below some PRGs for which a sum-of-squares attack can successfully distinguish PRG images from random points:

1. Let $F : \{0,1\}^{nb} \to \{0,1\}^m$ be *two block-local*, i.e., the input is partitioned into n blocks of size b and each output depends on two blocks. If $m \in \Omega(2^{2b} \cdot \log^2(n) \cdot n)$ is big enough, then there is an efficient attack on F [7].
2. Let $c > 0$ be a constant and let Y be a distribution over $\mathbb{R}$ s.t. we have $\Pr_{y \leftarrow Y}[y \notin [a, a+c]] \geq \frac{1}{10}$ for each $a \in \mathbb{R}$. Let $F : \{0,1\}^n \to \mathbb{R}^m$ be a PRG of degree d over the reals s.t. the polynomials in F have at most s monomials. If $m \in \Omega(\log^2(n) \cdot s \cdot n^{\lceil d/2 \rceil})$ is big enough and if we assume a special assumption for the polynomials $f_1, \ldots, f_m$, there is an efficient attack that can successfully distinguish images of F from points $y \leftarrow Y^m$ [7].

3. Let $t \in \text{poly}(n)$ and let Q be a distribution of quadratic polynomials in $\mathbb{R}[X]$ with some special properties. If $m \in \log(n)^{\Omega(1)} \cdot n$ is big enough, there is an efficient algorithm that can extract with high probability the input x from $(F, F(x))$ where we sample $x \leftarrow [-t, t]^n$ and $F \leftarrow Q^m$ [8].

1.4 Organization of this Text

In Section 2, we will introduce some algebraic and cryptographic preliminaries. In Section 3, we will give an algorithm that finds non-trivial polynomials that vanish on the images of PRGs $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ of constant degree d and prove that one can find such polynomials of sublinear degree if F is of polynomial stretch $m = n^{1+e}$. In Section 4, we will give a distinguishing attack on F of time and space complexity $n^{O(n^{1-\frac{e}{d-1}})}$ and prove that it has an advantage of at least $1-O(n^{1-\frac{e}{d-1}}/q)$. In Section 5, we will investigate the case of small constant moduli q. For simplicity, we will only treat the representative case $q = 2$, however all results shown for $q = 2$ can be generalized for any small or constant prime q. We will show in this section, that one can find a polynomial of sublinear degree that vanishes on the image of F, but does not vanish everywhere on $\mathbb{Z}_q^m$. This leads to a second attack on degree-d PRGs $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ of complexity $n^{O(n^{1-\frac{e}{d-1}})}$ and subexponential advantage $q^{-O(n^{1-\frac{e}{d-1}})}$. Additionally, we will in this section give an attack on d-local PRGs $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ that has a time and space complexity of $2^{O(n^{1-\frac{e'}{(q-1)d-1}})}$ and noticeable advantage $O(n^{-\frac{e'}{(q-1)d-1}})$ for every constant $e' < e$. In Section 6, we will derive some insights for the design of PRGs that shall be secure against subexponential adversaries.

In the full version of this text [45], we will give an exhaustive comparison between our algorithms and Groebner basis-based algorithms, derive some insights for Groebner basis-based algorithms, give a new attack algorithm against PRGs of constant degree and polynomial stretch, which is also Groebner basis- or rather Macaulay matrix-based, and prove that it is subexponential in the worst case. Additionally, we will formally prove some claims that were only sketched and give some algebraic background.

2 Preliminaries

2.1 Notation

Denote by $\mathbb{N} = \{1, 2, 3, \ldots\}$ the set of natural numbers and by $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$ the set of natural numbers plus zero.

For the rest of this text, by k we will always denote a field and by $k[X_1, \ldots, X_n]$ resp. $k[Y_1, \ldots, Y_m]$ the corresponding polynomial ring, for $n, m \in \mathbb{N}$. Since the numbers of X and Y variables will always be n resp. m, by abuse of notation, we will write $k[X]$ resp. $k[Y]$ instead of $k[X_1, \ldots, X_n]$ resp. $k[Y_1, \ldots, Y_m]$.

Let $f \in k[X]$. When we speak of f's *degree* we always mean its *total degree* that is the minimum number $d \in \mathbb{N}_0$ s.t. f can be written as a k-linear combination of monomials that are the product of $\leq d$ variables.

If S is a finite set, we denote by $x \leftarrow S$ the fact that the random variable x is drawn uniformly and independently at random from S.

For a number $q \in \mathbb{N}$, we define the finite ring $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$.

We will denote by n the security parameter in this text. The parameter $m = m(n)$ will in most cases be dependent on n. For this to be consistent, we assume in those cases that m is time-constructible.

We call a function $\epsilon : \mathbb{N} \to [0,1]$ negligible, if we have $\lim_{n\to\infty} \epsilon(n) \cdot n^d =$ for each $d \in \mathbb{N}$. By $\mathrm{poly}(n) := \left\{ f : \mathbb{N} \to \mathbb{N} \mid \exists c, d \in \mathbb{N} : f(n) \leq n^d + c \right\}$ we denote the set self-maps of the natural numbers that are upper-bounded by constant-degree polynomials.

Given two discrete distributions $\mathcal{X}$ and $\mathcal{Y}$, we define their statistical distance as $\Delta(\mathcal{X}, \mathcal{Y}) := \frac{1}{2} \sum_x |\mathcal{X}(x) - \mathcal{Y}(x)|$.

Given two vector spaces V, W over the same field, we denote by $V \oplus W$ their direct sum.

2.2 Mathematical Preliminaries

We will introduce now some basic facts and notions for the polynomial ring $k[X]$:

Remark 1. Let $n \in \mathbb{N}$. Let k be any field and consider the polynomial ring $k[X] = k[X_1, \ldots, X_n]$. The ring $k[X]$ is graded and can be written as

$$k[X] = \bigoplus_{\ell=0}^{\infty} k[X]^\ell \tag{37}$$

where $k[X]^\ell$ is the finite-dimensional k-vector space generated by all monomials of total degree $= \ell$, i.e.

$$k[X]^\ell = \mathrm{span}_k \left\{ X_1^{a_1} \cdots X_n^{a_n} \mid a_1, \ldots, a_n \in \mathbb{N}_0, a_1 + \ldots + a_n = \ell \right\}. \tag{38}$$

By $k[X]^{\leq \ell}$ we denote the space generated by all monomials of degree $\leq \ell$, i.e.

$$k[X]^{\leq \ell} := \bigoplus_{i=0}^{\ell} k[X]^i. \tag{39}$$

The dimensions of $k[X]^\ell$ and $k[X]^{\leq \ell}$ are given by

$$\dim_k k[X]^\ell = \binom{n+\ell-1}{\ell} \quad \text{and} \quad \dim_k k[X]^{\leq \ell} = \binom{n+\ell}{\ell}. \tag{40}$$

Sometimes, we will use the notion $X^{\alpha_1}, X^{\alpha_2}, \ldots$ to denote monomials

$$X_1^{a_{1,1}} \cdots X_n^{a_{1,n}}, X_1^{a_{2,1}} \cdots X_n^{a_{2,n}}, \ldots. \tag{41}$$

In those cases, the $\alpha_1, \alpha_2, \ldots \in \mathbb{N}_0^n$ are multi-indices given by

$$\alpha_i = (a_{i,1}, \ldots, a_{i,n}). \tag{42}$$

In the case of $k = \mathbb{Z}_q$ for a small prime q, it might be necessary to include the field equations $X_1^q - X_1, \ldots, X_n^q - X_n$ when considering $k[X]$. In this text, we will only treat the case $q = 2$, which is representative for all cases of constant moduli q:

Remark 2. Let $k = \mathbb{Z}_2$ and denote by $I \subset \mathbb{Z}_2[X]$ the ideal generated by the field equations of $\mathbb{Z}_2$, i.e.

$$I := (X_1^2 - X_1, \ldots, X_n^2 - X_n). \tag{43}$$

The ring $\mathbb{Z}_2[X]/I$ is not graded any more, since I is not a homogenous ideal. However, it is still filtrated where the filtration steps are given by the vector spaces

$$\mathbb{Z}_2[X]^{\leq \ell}/(I \cap \mathbb{Z}_2[X]^{\leq \ell}). \tag{44}$$

A basis for $\mathbb{Z}_2[X]^{\leq \ell}/(I \cap \mathbb{Z}_2[X]^{\leq \ell})$ is given by the set of all monomials of degree $\leq \ell$ where each variable occurs at most once. Therefore, we have for $\ell \leq n$

$$\dim_{\mathbb{Z}_2}(\mathbb{Z}_2[X]^{\leq \ell}/(I \cap \mathbb{Z}_2[X]^{\leq \ell})) = \sum_{i=0}^{\ell} \binom{n}{i}. \tag{45}$$

In particular, we have for $\ell \leq n$

$$\sum_{i=0}^{\ell} \binom{n}{i} = \dim_{\mathbb{Z}_2}(\mathbb{Z}_2[X]^{\leq \ell}/(I \cap \mathbb{Z}_2[X]^{\leq \ell})) \leq \dim_{\mathbb{Z}_2}(\mathbb{Z}_2[X]^{\leq \ell}) = \binom{n+\ell}{\ell}. \tag{46}$$

Definition 1 (Dual Morphisms). *Let k be any field and $k[X] = k[X_1, \ldots, X_n]$. Let $f_1, \ldots, f_m \in k[X]$ and $k[Y] = k[Y_1, \ldots, Y_m]$. The function*

$$F : k^n \longrightarrow k^m \tag{47}$$

$$x \longrightarrow (f_1(x), \ldots, f_m(x)) \tag{48}$$

gives us a geometrical map that is continuous in the Zariski topology. It has a ***dual morphism*** *of k-algebras*

$$\phi : k[Y] \longrightarrow k[X] \tag{49}$$

$$Y_i \longrightarrow f_i(X) \tag{50}$$

that maps each polynomial $h \in k[Y]$ to a polynomial $h(f_1(X), \ldots, f_m(X))$ in $k[X]$ by substituting each appearance of Y_i in h by f_i for each $i \in [m]$.

Definition 2 (Algebraic Independence). *In the situation of Definition 1, we call $f_1, \ldots, f_m$* ***algebraically independent*** *if ϕ is injective.*

If ϕ is not injective, we call a non-zero element $h \in \ker \phi$ of its kernel an ***algebraic relation*** *of the elements $f_1, \ldots, f_m$.*

When working with polynomials over $k = \mathbb{Z}_q$ for q sufficiently large, the Schwartz-Zippel Lemma is a helpful tool to lower bound the probability that a fixed polynomial vanishes on a random point of $\mathbb{Z}_q^m$.

Lemma 1 (Schwartz-Zippel [40]). *Let $q \in \mathbb{N}$ be a prime and let $m, d \in \mathbb{N}$. Let $h \in k[Y]$ be a polynomial of degree d. Then, we can bound the probability of h vanishing on a random point of $\mathbb{Z}_q^m$ by*

$$\Pr_{y \leftarrow \mathbb{Z}_q^m}[h(y) = 0] \leq d/q. \tag{51}$$

2.3 Cryptographic Preliminaries

In this subsection, we will introduce the notion of pseudorandom number generators, and define a simple security game for them.

Definition 3 (Pseudorandom Number Generators). *Let $m : \mathbb{N} \to \mathbb{N}$ be a time-constructible function and let k be any field. A **pseudorandom number generator** (PRG) is a family of functions $F = (F_n)_{n \in \mathbb{N}}$ s.t. each F_n is a deterministic function*

$$F_n : k^n \longrightarrow k^m. \tag{52}$$

*We call m the **stretch** of the PRG. If there is a constant $e > 0$ s.t. $m \geq n^{1+e}$, we say that $(F_n)_{n \in \mathbb{N}}$ is a **poly-stretch** PRG.*

Remark 3. If $F = (F_n)_{n \in \mathbb{N}}$ is a PRG, we will, by abuse of notation, just write

$$F : k^n \to k^m. \tag{53}$$

For a given n, we will further write F when we actually mean F_n.

The adversaries in this text are always given a description of F_n (which we will simply denote by F) that allows the adversary to efficiently evaluate F_n on points of k^n. We assume that this description of F_n always contains binary representations of the numbers n, m and a description of the field k that allows the adversary to perform arithmetic operations over k. Additionally, if F is of locality or degree $d \in \mathbb{N}$ (in the sense of Definition 4), we expect the description of F to contain a binary representation of d.

Definition 4 (Locality and Degree of PRGs). *Let $F = (F_n)_n$ be a PRG of stretch m over k. Let $d \in \mathbb{N}$. For $n \in \mathbb{N}$ and $i \in [m]$, we denote by $f_{n,i} : k^n \to k$ the function of the i-th output of F_n. I.e., $f_{n,1}, \ldots f_{n,m}$ are uniquely determined by*

$$F(x) = (f_{n,1}(x), \ldots, f_{n,m}(x)) \tag{54}$$

for all $x \in k^n$.

1. *We say that F is d-**local** if each of its output values depends on only d input values. I.e. for each $n \in \mathbb{N}$ and $i \in [m]$ there is a function $g : k^d \to k$ and indices $l_1, \ldots, l_d \in [n]$ s.t. we have for each $x \in k^n$*

$$f_{n,i}(x_1, \ldots, x_n) = g(x_{l_1}, \ldots, x_{l_d}).$$

2. *We say that F is of **degree** d if each $f_{n,i}$ can be computed by a polynomial of degree d. I.e., for each $n \in \mathbb{N}$ and $i \in [m]$ the function $f_{n,i} : k^n \to k$ coincides with a polynomial in $k[X]$ of degree $\leq d$. In this case, by abuse of notation, we will directly interpret $f_{n,i}$ as an element of $k[X]^{\leq d}$.*

For a given n, we will simply write $f_1, \ldots, f_m$ instead of $f_{n,1}, \ldots, f_{n,m}$ to denote the partial functions of F. We will usually say in those cases that F is made up of or consists of $f_1, \ldots, f_m$.

Definition 5 (Security Game for PRGs). *Let k be finite now and let $F : k^n \to k^m$ be a PRG. We describe here a non-interactive security game between a probabilistic challenger $\mathcal{C}$ and a (potentially probabilistic) adversary $\mathcal{A}$. The game is parametrized by n and proceeds in the following steps:*

1. *$\mathcal{C}$ 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 $\mathcal{C}$. It makes some computations on its own without interacting with $\mathcal{C}$ and finally sends a bit b' to $\mathcal{C}$.*

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

$$\mathsf{adv}_F(\mathcal{A}) := 2\Pr[\mathcal{A} \text{ wins}] - 1 = \Pr_{x \leftarrow k^n}[\mathcal{A}(F, F(x)) = 0] + \Pr_{y \leftarrow k^m}[\mathcal{A}(F, y) = 1] - 1 \tag{55}$$

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

We define $\mathcal{A}$'s space complexity to be the number of bits and elements of k it stores simultaneously in step 2, and we define its time complexity by the number of bit-operations and arithmetical operations over k it performs in step 2.

Definition 6. *We say that an algorithm is **subexponential** if there is a constant $e \in [0, 1)$ s.t. its time and space complexities lie in $2^{O(n^e)}$.*

Lemma 2 (Leftover Hash Lemma (Matrix Version) [21]). *Let $q \in \mathbb{N}$ be a prime and let $m, m' \in \mathbb{N}$ be natural numbers.*

If we draw $A_1, A_2 \leftarrow \mathbb{Z}_q^{m \times m'}, y_1 \leftarrow \{0, 1\}^m, y_2 \leftarrow \mathbb{Z}_q^{m'}$, we have

$$\Delta((A_1, A_1 y_1), (A_2, y_2)) \leq \frac{1}{2}\sqrt{2^{m' \cdot \log q - m}}. \tag{56}$$

3 Finding Algebraic Relations

In this section, we introduce an algorithm $\mathcal{B}1$ that – given a set of polynomials – finds an algebraic relation among these polynomials. Further, we will prove upper bounds for the degree of this relation and for the complexity of the algorithm.

Now, let $n, m, d \in \mathbb{N}$ and let k be any field in this section. Let $F : k^n \to k^m$ be a polynomial mapping of degree $\leq d$ that is given by polynomials $f_1, \ldots, f_m \in k[X]$ of degree $\leq d$.

Denote by $\phi : k[Y] \to k[X]$ the dual morphism to F. Note, that ϕ expands the degrees of its inputs by a factor of at most d, i.e., we have for each $\ell \in \mathbb{N}_0$

$$\phi(k[Y]^{\leq \ell}) \subseteq k[X]^{\leq d \cdot \ell}. \tag{57}$$

Let $\ker \phi = \{h \in k[Y] \mid \phi(h) = 0\}$ be the kernel of ϕ. Our aim is to compute a non-trivial element of $\ker \phi$.

We will propose a straight-forward approach for this task: For $\ell = 1, 2, \ldots$, the algorithm $\mathcal{B}1$ will compute a monomial basis for $k[Y]^{\leq \ell}$ and check – by linear algebra – if the vector space $k[Y]^{\ell} \cap \ker \phi$ is non-trivial. If $k[Y]^{\ell} \cap \ker \phi$ contains a non-trivial element eventually, $\mathcal{B}1$ will output it and terminate. Formally, $\mathcal{B}1$ is given by:

Algorithm 1. The algorithm $\mathcal{B}1$ gets as input numbers $n, m, d \in \mathbb{N}$, a description of k and a description of a polynomial map $F : k^n \to k^m$. It has to output a non-zero element of $\ker \phi$.

$\mathcal{B}1$ sets an iteration variable $\ell := 1$ and proceeds in the following steps:

1. $\mathcal{B}1$ computes $N := \binom{n+d\ell}{d\ell}$ and $M := \binom{m+\ell}{\ell}$
2. $\mathcal{B}1$ computes a finite list $(Y_1^{a_1} \cdots Y_m^{a_m} \mid a_1, \ldots, a_m \in \mathbb{N}_0, a_1 + \ldots + a_m \leq \ell) = (Y^{\alpha_1}, \ldots, Y^{\alpha_M})$ of all monomials in $k[Y]$ of degree $\leq \ell$.
3. $\mathcal{B}1$ applies ϕ to each Y^{α_i} and computes a second list $(\phi(Y^{\alpha_1}), \ldots, \phi(Y^{\alpha_M}))$ of polynomials in $k[X]$ of degree $\leq d\ell$.
4. Let $X^{\beta_1}, \ldots, X^{\beta_N}$ be the set of all monomials in $k[X]$ of degree $\leq d\ell$. Then, $X^{\beta_1}, \ldots, X^{\beta_N}$ is a basis of $k[X]^{\leq d\ell}$ and for each $\phi(Y^{\alpha_i})$ there is a unique column-vector $w_i = (w_{i,1}, \ldots, w_{i,N}) \in k^N$ s.t.

$$\phi(Y^{\alpha_i}) = \sum_{j=1}^{N} w_{i,j} \cdot X^{\beta_j}. \tag{58}$$

   $\mathcal{B}1$ computes for each Y_{α_i} the corresponding vector w_i and writes down the matrix

$$W_\ell := \left(w_1 | \ldots | w_M\right) \in k^{N \times M}. \tag{59}$$

5. $\mathcal{B}1$ uses Gaussian elimination to compute a basis for the vector space

$$K_\ell := \left\{r \in k^M \mid W_\ell \cdot r = 0\right\}. \tag{60}$$

6. If K_ℓ is the trivial null-space, $\mathcal{B}1$ increases ℓ by one and goes back to step 2.

7. Otherwise, $\mathcal{B}1$ chooses an arbitrary non-zero vector $r \in K_\ell$, computes the polynomial

$$h := r_1 \cdot Y^{\alpha_1} + \ldots + r_M \cdot Y^{\alpha_M} \in k[Y] \tag{61}$$

of total degree $\leq \ell$ and outputs it.

We will show the following properties for $\mathcal{B}1$:

Lemma 3. *Let $n, m, d \in \mathbb{N}$ s.t. $m > n$. Let $F : k^n \to k^m$ be a polynomial map of degree $\leq d$. Let $y \in k^m$. We have the following:*

1. *On input n, m, d and F, $\mathcal{B}1$ will always terminate after a finite number of steps and output a polynomial h.*
2. *The polynomial h outputted by $\mathcal{B}1$ will always lie in $\ker \phi$ and be non-zero.*

Proof. 1. Note that $m > n$. The first claim of the lemma is equivalent to stating that m elements of $k[X]$ must be algebraically dependent and $\phi : k[Y] \to k[X]$ cannot be injective. This is a well-known fact in algebra and is easy to prove, however writing down a formally correct proof will make the notions of transcendency bases and function fields necessary. A proof of this statement can be found in the full version [45] of this text.

2. Assume that $\mathcal{B}1$ stops after D iterations and outputs h. Then, h is a polynomial in $k[Y]$ of degree D and can be written as

$$h := r_1 \cdot Y^{\alpha_1} + \ldots + r_M \cdot Y^{\alpha_M} \in k[Y] \tag{62}$$

where $M = \binom{m+D}{D}$ and r is a non-zero kernel element of R_D. I.e., we have

$$\sum_{i=1}^{M} r_i \cdot w_i = 0. \tag{63}$$

Since the entries of w_i are exactly the coefficients of $\phi(Y^{\alpha_i})$, we have

$$\phi(h) = \phi\left(\sum_{i=1}^{M} r_i \cdot Y^{\alpha_i}\right) = \sum_{i=1}^{M} r_i \cdot \phi\left(Y^{\alpha_i}\right) = 0. \tag{64}$$

Ergo, $h \in \ker \phi$.

Lemma 4. *Assume that $\mathcal{B}1$ terminates after D iterations. Then, its space complexity can be bounded by $O(NM)$ and its time complexity can be bounded by $O(DN^2M)$ for $N = \binom{n+d \cdot D}{d \cdot D}$ and $M = \binom{m+D}{D}$.*

Proof. In each iteration step, $\mathcal{B}1$ computes a matrix of shape at most $N \times M$ over k. Therefore, the number of bits and elements of k it needs to store simultaneously can be bounded by $O(NM)$.

We can bound the time complexity of each iteration step from above by the time complexity of the D-th iteration step. In this step, $\mathcal{B}1$ performs Gaussian elimination on an $N \times M$-matrix which needs $O(N^2M)$ arithmetical operations over k. Therefore, the number of bit-operations and arithmetical operations $\mathcal{B}1$ needs to do in each step can be bounded by $O(N^2M)$, and $\mathcal{B}1$'s total time complexity can be bounded by $O(DN^2M)$.

Note, that $\mathcal{B}1$ starts at $\ell = 1$ and increases ℓ by one subsequently. Since $\mathcal{B}1$ terminates only if it finds a non-trivial element in $k[Y]^\ell \cap \ker \phi$, this means that the number D of iterations $\mathcal{B}1$ has to perform is exactly the lowest total degree of non-zero elements of $\ker \phi$.

Lemma 5. *$\mathcal{B}1$ terminates after D iterations iff $D = \min\{\deg h \mid h \in \ker \phi, h \neq 0\}$.*

3.1 Bounding D for Poly-Stretch PRGs

We have seen in the last subsection that the time and space complexity of $\mathcal{B}1$ is substantially influenced by D. Since D is the minimal degree of a non-trivial element of $\ker \phi$, our aim in this subsection is to bound the degree of algebraic relations for all sets of polynomials $f_1, \ldots, f_m$ of degree $\leq d$.

Since we are interested in the case of poly-stretch PRGs, we introduce an additional constant $e > 0$ and assume that m is always larger than n^{1+e}.

Let ϕ_ℓ be the restriction of ϕ on $k[Y]^{\leq \ell}$. Then, each ϕ_ℓ is a linear map of type $k[Y]^{\leq \ell} \to k[X]^{\leq d \cdot \ell}$. We can guarantee that ϕ_ℓ has a non-trivial kernel, if the dimension of $k[Y]^{\leq \ell}$ exceeds the dimension of $k[X]^{\leq d \cdot \ell}$. Now, the dimensions of $k[Y]^{\leq \ell}$ and $k[X]^{\leq d \cdot \ell}$ are given by

$$\dim_k(k[Y]^{\leq \ell}) = \binom{m+\ell}{\ell} \quad \text{and} \quad \dim_k(k[X]^{\leq d \cdot \ell}) = \binom{n + d \cdot \ell}{d \cdot \ell}. \tag{65}$$

Therefore, we get for algorithm $\mathcal{B}1$:

Lemma 6. *Let D be the number of iterations of $\mathcal{B}1$. Then, we have*

$$D \leq \min\left\{\ell \in \mathbb{N} \mid \binom{m+\ell}{\ell} > \binom{n + d \cdot \ell}{d \cdot \ell}\right\}. \tag{66}$$

Inequality Eq. (66) gives us a tool to compute a worst-case bound for $\mathcal{B}1$'s complexity for each possible case of polynomials $f_1, \ldots, f_m$. In the next lemma, we will show that we can bound D by $O(n^{1-\frac{e}{d-1}})$. While $n^{1-\frac{e}{d-1}}$ is non-constant for $e < d-1$, it implies that we can bound the complexity of $\mathcal{B}1$ subexponentially by $n^{O(n^{1-\frac{e}{d-1}})}$.

Lemma 7 (Main Inequalities). *Let $d \in \mathbb{N}, d \geq 2$ and $e \in (0, d-1]$. Let $m : \mathbb{N} \to \mathbb{N}$ be a function with $m(n) \geq n^{1+e}$ and set $c = 2^{\frac{d}{d-1}}$. Then, we have for all integers $n \geq (2dc)^{\frac{d-1}{e}}$*

$$\binom{m(n) + L(n)}{L(n)} > \binom{n + dL(n)}{dL(n)} \tag{67}$$

where $L(n) = \left\lceil cn^{1-\frac{e}{d-1}} \right\rceil$.

Proof. In the proof, by abuse of notation, we write $m = m(n)$ and $L = L(n)$.

We first prove the inequality $dL \leq n$. In fact, we have

$$dL = d\left\lceil cn^{1-\frac{e}{d-1}} \right\rceil \leq d(c \cdot n^{1-\frac{e}{d-1}} + 1) = dc \cdot n^{1-\frac{e}{d-1}} + d. \tag{68}$$

Since $n \geq (2dc)^{\frac{d-1}{e}}$, d must be smaller than $n/2$. For $dc \cdot n^{1-\frac{e}{d-1}}$, we have the equivalent inequalities

$$dc \cdot n^{1-\frac{e}{d-1}} \leq \frac{1}{2}n \iff dc \leq \frac{1}{2}n^{\frac{e}{d-1}} \tag{69}$$

$$\iff 2dc \leq n^{\frac{e}{d-1}} \tag{70}$$

$$\iff (2dc)^{\frac{d-1}{e}} \leq n \tag{71}$$

where the last inequality is exactly the requirement in our lemma for n. Therefore, we get

$$dL \leq dcn^{1-\frac{e}{d-1}} + d \leq \frac{n}{2} + \frac{n}{2} = n. \tag{72}$$

Now, for the claimed inequality of the lemma, we have the following chain of equivalent inequalities

$$\binom{m+L}{L} > \binom{n+dL}{dL} \tag{73}$$

$$\iff \frac{(m+L)\cdots(m+1)}{L!} > \frac{(n+dL)\cdots(n+1)}{(dL)!} \tag{74}$$

$$\iff (m+L)\cdots(m+1)\cdot(dL)\cdots(L+1) > (n+dL)\cdots(n+1). \tag{75}$$

Note, that we have for all n the inequalities

$$(m+L)\cdots(m+1) > m^L, \tag{76}$$

$$(dL)\cdots(L+1) > L^{(d-1)L}. \tag{77}$$

For the right-hand side, we have

$$(n+dL)\cdots(n+1) \leq (n+dL)^{dL} \leq (2n)^{dL} = n^{dL}\cdot 2^{dL}. \tag{78}$$

By using the inequalities Eqs. (76) to (78), we see that Eq. (75) is implied by the inequality

$$m^L \cdot L^{(d-1)L} \geq n^{dL} \cdot 2^{dL}. \tag{79}$$

By reducing Eq. (79) to the L-th root, we get the equivalent inequality

$$m \cdot L^{(d-1)} \geq n^d \cdot 2^d. \tag{80}$$

Now, it is easy to show that this inequality holds:

$$m \cdot L^{(d-1)} \geq n^{1+e} \cdot \left(c \cdot n^{1-\frac{e}{d-1}}\right)^{(d-1)} = n^{1+e+(d-1)-e} \cdot c^{d-1} = n^d \cdot 2^d. \tag{81}$$

This completes the proof.

Lemmas 3 to 7 now implies the following theorem:

Theorem 1. *Let $d \in \mathbb{N}$, $e > 0$ be constants and $m \geq n^{1+e}$. Let $f_1, \ldots, f_m \in k[X]$ be polynomials of degree $\leq d$.*

Then, $\mathcal{B}1$ in Algorithm 1 outputs a non-trivial element of $\ker \phi$ of degree $O(n^{1-\frac{e}{d-1}})$. Its space and time complexities lie in $n^{O(n^{1-\frac{e}{d-1}})}$.

4 Attacks on Constant-Degree PRGs over Large Moduli

In this section, we will focus on the case $k = \mathbb{Z}_q$ for a prime q that is sufficiently high (e.g. $q \in \Omega(n)$). We claim that in this case $\mathcal{B}1$ from Algorithm 1 gives us a subexponential attack on each PRG of constant degree over $\mathbb{Z}_q$ and poly-stretch. In this section, we will prove:

Theorem 2. *Let $d \in \mathbb{N}, e > 0$ be constants. Let $m \geq n^{1+e}$ and let $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ be a PRG of degree d over $\mathbb{Z}_q$.*

Then, there is an attack algorithm $\mathcal{A}1$ whose time and space complexities are bounded from above by $n^{O(n^{1-\frac{e}{d-1}})}$. Further, there exists a constant $c > 0$ s.t. $\mathcal{A}1$'s advantage in the security game Definition 5 is lower bounded by

$$\mathsf{adv}_F(\mathcal{A}1) \geq 1 - c \cdot n^{1-\frac{e}{d-1}}/q. \tag{82}$$

The attack $\mathcal{A}1$ on F is defined as follows:

Algorithm 2. $\mathcal{A}1$ receives as input a description of F that includes the numbers $n, m, q, d \in \mathbb{N}$ and an element $y^* \in \mathbb{Z}_q^m$. The goal of $\mathcal{A}1$ is to output 0, if y^* lies in the image of F, and 1, otherwise.

$\mathcal{A}1$ proceeds in two simple steps:

1. $\mathcal{A}1$ executes the algorithm $\mathcal{B}1$ from Algorithm 1 on the input n, m, d, q, F and receives a non-zero polynomial $h \in \mathbb{Z}_q[Y]$ as output.
2. $\mathcal{A}1$ outputs 0 if $h(y^*) = 0$. Otherwise, $\mathcal{A}1$ outputs 1.

The bound on the time and space complexities of $\mathcal{A}1$ follows now from Theorem 1. The advantage of $\mathcal{A}1$ can be bounded as follows:

If $b = 0$ in the security game of Definition 5, then the challenger $\mathcal{C}$ samples $x \leftarrow \mathbb{Z}_q^n$ and gives the pseudorandom image $y^* = F(x)$ to $\mathcal{A}1$. The polynomial h outputted by $\mathcal{B}1(F)$ lies in the kernel of ϕ, i.e., we have the equality $h(F(X)) = 0$ of polynomials in $\mathbb{Z}_q[X]$. In particular, we have $h(F(x)) = 0$ for each $x \in \mathbb{Z}_q^n$. Therefore, $\mathcal{A}1$ always outputs 0 if $b = 0$.

If $b = 1$ in the security game in Definition 5, then the challenger $\mathcal{C}$ samples a uniformly random $y \leftarrow \mathbb{Z}_q^m$ and gives $y^* = y$ to $\mathcal{A}1$. Since h is non-zero and of degree $O(n^{1-\frac{e}{d-1}})$, the probability that h vanishes on y can be bounded by

$$\Pr_{y \leftarrow \mathbb{Z}_q^n}[h(y) = 0] \leq O(n^{1-\frac{e}{d-1}})/q \tag{83}$$

according to Lemma 1. Therefore, $\mathcal{A}1$ will output 1 in this case with probability at least $1 - O(n^{1-\frac{e}{d-1}})/q$.

For the overall advantage of $\mathcal{A}1$, we get

$$\begin{aligned}\mathsf{adv}_F(\mathcal{A}1) = &\Pr_{x \leftarrow \mathbb{Z}_q^n}[\mathcal{A}1(F, F_n(x)) = 0] + \Pr_{y \leftarrow \mathbb{Z}_q^m}[\mathcal{A}1(F, y) = 1] - 1 &(84)\\ \geq &1 + 1 - O(n^{1-\frac{e}{d-1}})/q - 1 = 1 - O(n^{1-\frac{e}{d-1}})/q. &(85)\end{aligned}$$

Remark 4. Algorithm $\mathcal{A}1$ proceeds in two steps: in its first step, it uses $\mathcal{B}1$ to compute an algebraic relation h of F, and in its second step, it uses h to decide if the given image $y^* \in \mathbb{Z}_q^m$ is truly random.

However, since the PRG F is fixed and publicly known, the attack $\mathcal{A}1$ can be interpreted as an attack with preprocessing: In a first phase, the so-called *preprocessing* or *offline* phase, $\mathcal{A}1$ uses $\mathcal{B}1$ to compute an algebraic relation h of F of degree D (without seeing the value $y^* \in \mathbb{Z}_q^m$).

In a second phase, the so-called *online* phase, $\mathcal{A}1$ receives $y^* \in \mathbb{Z}_q^m$ and only needs to evaluate h on y^*.

If $m \geq n^{1+e}$, then the degree of h is bounded by $D \leq cn^{1-\frac{e}{d-1}}$ for some constant c. The evaluation of h requires $(D+1) \cdot \binom{m+D}{D}$ arithmetic operations over $\mathbb{Z}_q$ which will be much less than the time $\mathcal{B}1$ needs (since $\mathcal{B}1$ needs to reduce a matrix of shape $\binom{m+D}{D} \times \binom{n+d \cdot D}{d \cdot D}$).

Therefore, from a practical point of view, it makes more sense to interpret $\mathcal{A}1$ as an attack with preprocessing, where we invest a big one-time cost to find a relation h of F in the preprocessing phase, and then a smaller, but still subexponential, cost of $(D+1) \cdot \binom{m+D}{D}$ to decide challenges of F.

5 Attacks on Binary PRGs

We want to focus on the case $q = 2$ in this section. Note, that Theorem 2 does not give us a meaningful attack for small values of q like 2. In fact, if we were to use naively algorithm $\mathcal{B}1$ from Algorithm 1 on m polynomials over $\mathbb{Z}_2$, $\mathcal{B}1$ may return a field equation $Y_i^2 - Y_i$ for some $i \in [m]$. This field equation will not help us in distinguishing pseudo-random images from random images, since it will vanish on each $y \in \mathbb{Z}_2^m$.

To avoid trivial relations over $\mathbb{Z}_2$, we will present here a modified version of $\mathcal{B}1$ – that we will call $\mathcal{B}2$ – that will always find a non-trivial algebraic relation of polynomials over $\mathbb{Z}_2$. For this sake, we set by abuse of notation

$$R_2[X] := \mathbb{Z}_2[X_1, \ldots, X_n]/(X_1^2 - X_1, \ldots, X_n^2 - X_n), \tag{86}$$

$$R_2[Y] := \mathbb{Z}_2[Y_1, \ldots, Y_m]/(Y_1^2 - Y_1, \ldots, Y_m^2 - Y_m). \tag{87}$$

As explained in Remark 2, the rings $R_2[X]$ and $R_2[Y]$ are filtrated. For $\ell \in \mathbb{N}$, we have

$$R_2[X]^{\leq \ell} = \mathbb{Z}_2[X]^{\leq \ell} / \left(\mathbb{Z}_2[X]^{\leq \ell} \cap (X_1^2 - X_1, \ldots, X_n^2 - X_n)\right), \tag{88}$$

$$R_2[Y]^{\leq \ell} = \mathbb{Z}_2[Y]^{\leq \ell} / \left(\mathbb{Z}_2[Y]^{\leq \ell} \cap (Y_1^2 - Y_1, \ldots, Y_m^2 - Y_m)\right). \tag{89}$$

Now let F be a PRG of degree d over $\mathbb{Z}_2$ and let $f_1, \ldots, f_m \in \mathbb{Z}_2[X]$ be the polynomials that make up F. Without loss of generality, we can assume that $f_1, \ldots, f_m$ are reduced modulo the field equations $X_1^2 - X_1, \ldots, X_n^2 - X_n$. Therefore, by abuse of notation, we interpret $f_1, \ldots, f_m$ as elements of $R_2[X]$. Now, the dual map $\phi : \mathbb{Z}_2[Y] \to \mathbb{Z}_2[X]$ descends well-defined to a ring homomorphism

$$\phi_2 : R_2[Y] \longrightarrow R_2[X] \tag{90}$$

$$Y_i \longmapsto f_i(X). \tag{91}$$

For the kernel of ϕ_2, we have

$$\ker \phi_2 = (\ker \phi + (Y_1^2 - Y_1, \ldots, Y_m^2 - Y_m))/(Y_1^2 - Y_1, \ldots, Y_m^2 - Y_m). \tag{92}$$

I.e., $\ker \phi_2$ contains all algebraic relations of $\ker \phi$ modulo the trivial ones from the field equations of $\mathbb{Z}_2$. In particular, a non-zero element of $\ker \phi_2$ is now guaranteed to not vanish everywhere on $\mathbb{Z}_2^m$.

To find a non-zero element of $\ker \phi_2$, the algorithm $\mathcal{B}2$ will proceed similarly as $\mathcal{B}1$: For increasing $\ell = 1, \ldots, m$, the algorithm $\mathcal{B}2$ computes a basis of the $\mathbb{Z}_2$-vector space $\ker \phi_2 \cap R_2[Y]^{\leq \ell}$. If $\ker \phi_2 \cap R_2[Y]^{\leq \ell}$ is non-zero, $\mathcal{B}2$ returns a non-zero element of it and terminates. Otherwise, $\mathcal{B}2$ increments ℓ and repeats these computations. Formally, $\mathcal{B}2$ is given by:

Algorithm 3. The algorithm $\mathcal{B}2$ gets as input numbers $n, m, d \in \mathbb{N}$, and a description of a polynomial map $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$. It has to output a non-zero element of $\ker \phi_2$.

For $\ell = 1, \ldots, m$, $\mathcal{B}2$ does the following:

1. $\mathcal{B}2$ computes $N := \dim_{\mathbb{Z}_2} \left(R_2[X]^{\leq d\ell}\right) = \sum_{i=0}^{\min(d\ell, n)} \binom{n}{i}$ and $M := \dim_{\mathbb{Z}_2} \left(R_2[Y]^{\leq \ell}\right) = \sum_{i=0}^{\ell} \binom{m}{i}$.
2. $\mathcal{B}2$ computes a finite list $(Y_1^{a_1} \cdots Y_m^{a_m} \mid a_1, \ldots, a_m \in \{0,1\}, a_1 + \ldots + a_m \leq \ell) = (Y^{\alpha_1}, \ldots, Y^{\alpha_M})$ of all monomials in $R_2[Y]$ of degree $\leq \ell$.
3. $\mathcal{B}2$ applies ϕ_2 to each Y^{α_i} and computes a second list $(\phi_2(Y^{\alpha_1}), \ldots, \phi_2(Y^{\alpha_M}))$ of polynomials in $R_2[X]$ of degree $\leq d\ell$.
4. Let $X^{\beta_1}, \ldots, X^{\beta_N}$ be the set of all monomials in $R_2[X]$ of degree $\leq d\ell$ where each variable appears at most once. For each $\phi_2(Y^{\alpha_i})$ let $w_i = (w_{i,1}, \ldots, w_{i,N}) \in \mathbb{Z}_2^N$ be the unique column-vector s.t.

$$\phi_2(Y^{\alpha_i}) = \sum_{j=1}^{N} w_{i,j} \cdot X^{\beta_j}. \tag{93}$$

   These vectors give us the matrix

$$W_\ell := \left(w_1 | \ldots | w_M\right) \in \mathbb{Z}_2^{N \times M}. \tag{94}$$

5. $\mathcal{B}2$ uses Gaussian elimination over $\mathbb{Z}_2$ to compute the kernel of W_ℓ

$$K_\ell := \left\{ r \in \mathbb{Z}_2^M \mid W_\ell \cdot r = 0 \right\}. \tag{95}$$

6. If K_ℓ is the trivial null-space, $\mathcal{B}2$ increases ℓ by one. If $\ell \leq m$, $\mathcal{B}2$ goes back to step 1. Otherwise, if $\ell = m + 1$, $\mathcal{B}2$ has exhausted the whole vector space $R_2[Y] = R_2[Y]^{\leq m}$. In this case, $\mathcal{B}2$ aborts, since now ϕ_2 must be injective.
7. If K_ℓ is not the null-space, $\mathcal{B}2$ chooses an arbitrary non-zero vector $r \in K_\ell$, computes the polynomial

$$h := r_1 \cdot Y^{\alpha_1} + \ldots + r_M \cdot Y^{\alpha_M} \in R_2[Y] \tag{96}$$

   of total degree $\leq \ell$ and outputs it.

We have for $\mathcal{B}2$ similar time and space bounds as for $\mathcal{B}1$:

Lemma 8. *Assume that $\mathcal{B}2$ terminates after D iterations. Then, its space complexity can be bounded by $O(NM)$ and its time complexity can be bounded by $O(DN^2M)$ for $N = \sum_{i=0}^{\min(dD,n)} \binom{n}{i}$ and $M = \sum_{i=0}^{D} \binom{m}{i}$.*

Similarly, as in Section 3, one can show that $\mathcal{B}2$ will return an algebraic relation of minimal degree, if such a relation exists:

Lemma 9. *Let $n, m, d \in \mathbb{N}$. Let $f_1, \ldots, f_m \in R_2[X]$ be polynomials of degree $\leq d$. Assume that the corresponding morphism*

$$\phi_2 : R_2[Y] \longrightarrow R_2[X] \tag{97}$$

$$Y_i \longmapsto f_i \tag{98}$$

is not injective and set $D := \min\{\deg h \mid h \in \ker \phi_2, h \neq 0\}$. Then, $\mathcal{B}2$ terminates after D iterations and outputs a non-zero element of $\ker \phi_2$ of degree D.

Now, let $e > 0$ and $d \in \mathbb{N}$ be constants and assume $m \geq n^{1+e}$. The inequality in Lemma 7 has a pendant that states that for almost all n we have

$$\sum_{i=0}^{L} \binom{m}{i} > \sum_{i=0}^{dL} \binom{n}{i} \tag{99}$$

where $L = \left\lceil c \cdot n^{1-\frac{e}{d-1}} \right\rceil$. We give a formal proof of this inequality in the full version [45] of this text. However, its proof is very similar to the proof of Lemma 7. It follows that $\mathcal{B}2$'s complexity is subexponential for $m \geq n^{1+e}$ polynomials $f_1, \ldots, f_m$:

Theorem 3. *Let $d \in \mathbb{N}$, $e > 0$ be constants and $m \geq n^{1+e}$. Let $f_1, \ldots, f_m \in R_2[X]$ be polynomials of degree $\leq d$.*

Then, $\mathcal{B}2$ in Algorithm 3 outputs a non-trivial element of $\ker \phi_2$ of degree $O(n^{1-\frac{e}{d-1}})$. Its space and time complexities lie in $n^{O(n^{1-\frac{e}{d-1}})}$.

5.1 Binary PRGs of Constant Degree

$\mathcal{B}2$ gives rise to the following attacker $\mathcal{A}2$ on degree-d PRGs over $\mathbb{Z}_2$:

Algorithm 4. The algorithm $\mathcal{A}2$ receives as input a description of a PRG $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ of degree d, which includes the numbers $n, m, d \in \mathbb{N}$, and an element $y^* \in \mathbb{Z}_2^m$. The goal of $\mathcal{A}2$ is to output 0, if y^* lies in the image of F, and 1, otherwise.

$\mathcal{A}2$ proceeds in two simple steps:

1. $\mathcal{A}2$ executes $\mathcal{B}2$ from Algorithm 3 on the input n, m, d, F and receives a non-zero polynomial $h \in R_2[Y]$ as output.
2. $\mathcal{A}2$ outputs 0 if $h(y^*) = 0$. Otherwise, $\mathcal{A}2$ outputs 1.

It is clear that $\mathcal{A}2$'s space and time complexities are comparable to the space and time complexities of $\mathcal{B}2$. However, since the degree D of h will be much higher than the cardinality of $\mathbb{Z}_2$, we cannot apply the Schwartz-Zippel Lemma any more. Since h is not zero in $R_2[Y]$, we can only guarantee that h vanishes on at most $2^m - 2^{m-D}$ points of $\mathbb{Z}_2^m$ (we show this in the full version [45]). This gives us the following theorem:

Theorem 4. *Let $d \in \mathbb{N}, e > 0$ be constants. Let $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ be a PRG of degree d and stretch $m \geq n^{1+e}$.*

Then, there is an attack algorithm $\mathcal{A}2$ whose time and space complexities are bounded from above by $n^{O(n^{1-\frac{e}{d-1}})}$. Further, there exists a constant $c > 0$ s.t. $\mathcal{A}2$'s advantage in the security game in Definition 5 against F is lower bounded by

$$\mathsf{adv}_F(\mathcal{A}2) \geq 2^{-cn^{1-\frac{e}{d-1}}}. \tag{100}$$

Theorem 4 is unsatisfying, since $\mathcal{A}2$'s advantage can only be guaranteed to be at least subexponential. One solution for this problem is to look at a multi-challenge security game for the PRG F where the adversary receives Q challenges $y_1^*, \ldots, y_Q^* \in \mathbb{Z}_2^m$ and has to guess if all $y_1^*, \ldots, y_Q^*$ have been drawn uniformly and independently at random from $\mathbb{Z}_2^m$ or if all $y_1^*, \ldots, y_Q^*$ lie in the image of F.

If the number of challenges is $Q \in 2^{\Omega(n^{1-\frac{e}{d-1}})}$, then the advantage of $\mathcal{A}2$ can be amplified to a positive constant. We give here an informal theorem for this observation and flesh out the details in the full version [45] of this text:

Theorem 5 (Multi-Challenge Attack (Informal)). *Let $d \in \mathbb{N}, e > 0$ be constants and let $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ be a PRG of degree d and poly-stretch $m \geq n^{1+e}$.*

Then, there is an attack algorithm whose time and space complexities are bounded from above by $n^{O(n^{1-\frac{e}{d-1}})}$ and whose advantage in breaking the pseudo-randomness of F when given $Q \in 2^{\Omega(n^{1-\frac{e}{d-1}})}$ challenges is a constant greater than zero.

5.2 Binary PRGs of Constant Locality

Now, let $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ be a poly-stretch PRG of constant locality $d \in \mathbb{N}$, i.e., each output bit of F is determined by at most d input bits. In case of a PRG of constant locality we can perform a subexponential attack (for a single challenge value) where we can guarantee a much better advantage than for $\mathcal{A}2$ in Theorem 4.

Theorem 6. *Let $d \in \mathbb{N}$ and $e > 0$ be constants. Let $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ be a PRG of locality d with poly-stretch $m \geq n^{1+e}$.*

There is an attack $\mathcal{A}3$ on F and a constant $c > 0$ s.t. $\mathcal{A}3$'s space and time complexities are bounded by $2^{O(n^{1-\frac{e'}{d-1}})}$ for each constant $e' \in (0, e)$ and whose advantage in the security game of Definition 5 is at least

$$\mathsf{adv}_F(\mathcal{A}3) \geq 1 - cn^{-\frac{e'}{d-1}}. \tag{101}$$

The idea of $\mathcal{A}3$ is to convert F to a PRG $G : \mathbb{Z}_q^n \to \mathbb{Z}_q^{m'}$ of degree d over $\mathbb{Z}_q$ with stretch $m' = \lfloor m/(3\log(q)) \rfloor$ for a prime $q \geq n$.

Let $f_1, \ldots, f_m \in R_2[X]$ be the polynomials that make up F. Since each f_i is d-local, there are polynomials $f_1', \ldots, f_m' \in \mathbb{Z}_q[X]$ of degree $\leq d$ that coincide with $f_1, \ldots, f_m$ on $\{0,1\}^n$. In fact, for $i \in [m]$, let $j_1, \ldots, j_d \in [n]$ and $u_i : \{0,1\}^d \to \{0,1\}$ s.t. for all $x \in \{0,1\}^n$

$$f_i(x) = u_i(x_{j_1}, \ldots, x_{j_d}). \tag{102}$$

Then, the polynomial $f_i' \in \mathbb{Z}_q[X]$ is given by

$$f_i'(X) := \sum_{z \in \{0,1\}^d} u_i(z) \cdot (1 - z_1 - X_{j_1} + 2z_1 X_{j_1}) \cdots (1 - z_d - X_{j_d} + 2z_d X_{j_d}). \tag{103}$$

However, the image of the $f_1', \ldots, f_m'$ does not look random over $\mathbb{Z}_q$, since it is contained in $\{0,1\}^m$ (if the input is chosen from $\{0,1\}^n$). To compensate for that, we use the Leftover Hash Lemma. Let $F' = (f_1', \ldots, f_m') : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ be the collection of all f_i'. $\mathcal{A}3$ samples now a random matrix $A = (a_{i,j})_{i,j} \leftarrow \mathbb{Z}_q^{m' \times m}$ and defines a PRG $G : \mathbb{Z}_q^n \to \mathbb{Z}_q^{m'}$ by

$$G(X) := A \cdot F'(X). \tag{104}$$

I.e., if G consists of the polynomials $g_1, \ldots, g_{m'}$, each g_i is given by

$$g_i = \sum_{j=1}^{m} a_{i,j} \cdot f_j'. \tag{105}$$

Now, G is a degree-d PRG over $\mathbb{Z}_q$. According to Lemma 2, the image of G will *look random* (relative to $\mathbb{Z}_q^{m'}$) if the image of F looks random (relative to $\{0,1\}^m$). Finally, $\mathcal{A}3$ can use $\mathcal{A}1$ from Theorem 2 to break the pseudorandomness of G (and break therefore the pseudorandomness of F).

We will now formally define how $\mathcal{A}3$ proceeds:

Algorithm 5. Let $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ be a PRG of locality d consisting of polynomials $f_1, \ldots, f_m \in R_2[X]$. The algorithm $\mathcal{A}3$ receives as input a description of F, which includes the numbers $n, m, d \in \mathbb{N}$, and an element $y^* \in \mathbb{Z}_2^m$. The goal of $\mathcal{A}3$ is to output 0, if y^* lies in the image of F, and 1, otherwise.

$\mathcal{A}3$ proceeds in the following steps:

1. $\mathcal{A}3$ searches for a prime number $q \in \{n, n+1, \ldots, 2n\}$. Because of Bertrand's postulate we know that such a prime must exist.
2. $\mathcal{A}3$ sets $m' := \lfloor m/\left(3\log q\right) \rfloor$
3. $\mathcal{A}3$ computes polynomials $f_1', \ldots, f_m' \in \mathbb{Z}_q[X]$ that coincide with $f_1, \ldots, f_m$ on $\{0,1\}^n$.
4. $\mathcal{A}3$ draws a random matrix $A \leftarrow \mathbb{Z}_q^{m' \times m}$ and sets

$$G(X) := A \cdot F'(X). \tag{106}$$

Now, $G : \mathbb{Z}_q^n \to \mathbb{Z}_q^{m'}$ is a polynomial map of degree d.

5. $\mathcal{A}3$ interprets y^* as a binary vector in $\{0,1\}^m \subseteq \mathbb{Z}_q^m$ and computes

$$y'^* := A \cdot y^* \in \mathbb{Z}_q^m. \tag{107}$$

6. $\mathcal{A}3$ runs algorithm $\mathcal{A}1$ on (G, y'^*) and returns the output of $\mathcal{A}1$.

Now, let e', e'' be constants s.t. $0 < e' < e'' < e$ and assume that $\mathcal{A}3$ found the prime number q. Since $q \leq 2n$, we have $m' \geq \frac{m}{3\log q} - 1 \geq \frac{n^{1+e}}{3\log(n)+3} - 1$. The term on the right-hand side becomes greater than $n^{1+e''}$ for n big enough. Ergo, we have $m' \geq n^{1+e''}$ for almost all n. It is easy to see that the time and space complexities of $\mathcal{A}3$ are dominated by the complexities of $\mathcal{A}1$, which are upper-bounded by $n^{O(n^{1-\frac{e''}{d-1}})}$. For n large enough, $\mathcal{A}3$ will therefore have complexities upper-bounded by $2^{O(n^{1-\frac{e'}{d-1}})}$

To bound the advantage of $\mathcal{A}3$, we first distinguish two cases:

1. If $y^* = F(x)$ for some $x \in \mathbb{Z}_2^n$, then y'^* will be of the form

$$y'^* = Ay^* = AF'(x) = G(x). \tag{108}$$

   In those cases, $\mathcal{A}1$ will always output zero.
2. If y^* is a random element of $\{0,1\}^m$, then Lemma 2 states that the statistical distance of the distributions

$$(A, y'^*) \text{ and } (A, r) \tag{109}$$

   for $r \leftarrow \mathbb{Z}_q^{m'}$ is less than $\frac{1}{2}\sqrt{2^{m'\log(q)-m}} \leq \frac{1}{2}q^{-m'}$. Therefore, the probability that $\mathcal{A}1$ outputs 1 in this case can be lower bounded by

$$1 - \frac{O(n^{1-\frac{e'}{d-1}})}{q} - \frac{1}{2}q^{-m'} \geq 1 - \frac{O(n^{1-\frac{e'}{d-1}})}{n} - \frac{1}{2}n^{-m'} \tag{110}$$

$$\geq 1 - O(n^{-\frac{e'}{d-1}}). \tag{111}$$

Now, we can bound the advantage of $\mathcal{A}3$ in the security-game of Definition 5 as follows:

$$\mathsf{adv}_F(\mathcal{A}3) \geq \Pr_{y\leftarrow\{0,1\}^m}[\mathcal{A}3(F,y)=1] + \Pr_{x\leftarrow\{0,1\}^n}[\mathcal{A}3(F,F(x))=0] - 1 \tag{112}$$

$$\geq 1 - O(n^{-\frac{e'}{d-1}}) + 1 - 1 \geq 1 - O(n^{-\frac{e'}{d-1}}). \tag{113}$$

6 Avoiding Subexponential Attacks

Finally, we want to discuss three counter-measures in the design of PRGs that help to avoid the attacks presented in this paper:

Rational Functions. In case of a large modulus $q \geq n$, the algorithm $\mathcal{A}1$ in Theorem 2 gives a subexponential attack on constant-degree PRGs with non-negligible advantage.

To avoid $\mathcal{A}1$, one can consider PRGs $F : \mathbb{Z}_q^n \to \mathbb{Z}_q^m$ that incorporate rational functions of constant degree i.e. where each output value is computed by $f_i(X) := \frac{g_1(X)}{h_1(X)} + \ldots + \frac{g_\ell(X)}{h_\ell(X)}$ for polynomials $g_1, \ldots, g_\ell, h_1, \ldots, h_\ell \in \mathbb{Z}_q[X]$ of constant degree d. The functions $f_1, \ldots, f_m$ are still algebraically dependent, since $m > n$. However, we cannot bound the degree of the relation outputted by $\mathcal{B}1$, since the set $\left\{ \frac{f}{g} \mid f, g \in \mathbb{Z}_q[X]^{\leq d}, g \neq 0 \right\}$ is not contained in a finite-dimensional vector space any more.

We conjecture that if ℓ grows polynomially with n, then this kind of PRGs could even be resistant against Groebner basis-based attacks like F4/F5 and XL, since these algorithms need to multiply the equality $f_i(X) = y_i$ with $h_1 \cdots h_\ell$ to get a polynomial equality of non-constant degree $d \cdot \ell$.

As a concrete challenge, we propose a PRG – parametrized by n – where q is the smallest prime in $[2^n, 2^{n+1}]$, ℓ equals n, each g_i is one and each h_i is a random sum of two variables of $X_1, \ldots, X_n$. For $m \in \Omega(n^2)$, there is a trivial attack on this PRG. However, for smaller m, let's say $m = n^{1.9}$, we don't know a subexponential attack on this PRG with provably non-trivial advantage.

Non-Constant Locality. In the case of binary poly-stretch PRGs of constant degree, we gave two attacks $\mathcal{A}2$ and $\mathcal{A}3$. For $\mathcal{A}2$, we can only guarantee a subexponentially small advantage. However, this is only a pessimistic lower-bound and does not exclude that $\mathcal{A}2$ may perform much better in praxis.

$\mathcal{A}3$ is guaranteed to have a high advantage, however it can only be applied on binary PRGs of constant locality. This means, that all PRG candidates in $\mathbf{NC}^0$ with poly-stretch are susceptible to subexponential attacks.

To avoid subexponential attacks for binary PRGs, the only option seems to be to design PRGs of non-constant locality.

Small Non-Constant Modulus. The attack $\mathcal{A}1$ needs that the modulus, over which the PRG is evaluated, is large enough, while the attack $\mathcal{A}3$ needs that the modulus is constant (since otherwise the PRG constructed by $\mathcal{A}3$ will not have constant degree).

One can try to avoid both attacks by setting q to a number between both extremes (for example $q = \Theta(\sqrt{n})$ for $e < 0.5$). For such moduli q, neither $\mathcal{A}1$ nor $\mathcal{A}3$ can be applied and the attack $\mathcal{A}2$ must be used. If $\mathcal{B}2$ uses the appropriate field equations $Y_i^q - Y_i$, it will find a non-trivial algebraic relation of sublinear degree, however it is hard to show in such cases that this relation will not vanish on a non-negligible portion of $\mathbb{Z}_q^m$, since the Schwartz-Zippel Lemma cannot be applied.

References

1. Applebaum, B. *Pseudorandom generators with long stretch and low locality from random local one-way functions* in *44th ACM STOC* (eds Karloff, H. J. & Pitassi, T.) (ACM Press, May 2012), 805–816.
2. Applebaum, B. *Cryptographic Hardness of Random Local Functions-Survey* in *TCC 2013* (ed Sahai, A.) **7785** (Springer, Heidelberg, Mar. 2013), 599.
3. Applebaum, B. *The Cryptographic Hardness of Random Local Functions – Survey* Cryptology ePrint Archive, Report 2015/165. `https://eprint.iacr.org/2015/165`. 2015.
4. Applebaum, B., Damgård, I., Ishai, Y., Nielsen, M. & Zichron, L. *Secure Arithmetic Computation with Constant Computational Overhead* in *CRYPTO 2017, Part I* (eds Katz, J. & Shacham, H.) **10401** (Springer, Heidelberg, Aug. 2017), 223–254.
5. Applebaum, B. & Lovett, S. *Algebraic attacks against random local functions and their countermeasures* in *48th ACM STOC* (eds Wichs, D. & Mansour, Y.) (ACM Press, June 2016), 1087–1100.
6. Ars, G., Faugère, J.-C., Imai, H., Kawazoe, M. & Sugita, M. *Comparison Between XL and Gröbner Basis Algorithms* in *Advances in Cryptology - ASIACRYPT 2004* (ed Lee, P. J.) (Springer Berlin Heidelberg, Berlin, Heidelberg, 2004), 338–353. ISBN: 978-3-540-30539-2. `https://doi.org/10.1007/978-3-540-30539-2_24`.
7. Barak, B., Brakerski, Z., Komargodski, I. & Kothari, P. K. *Limits on Low-Degree Pseudorandom Generators (Or: Sum-of-Squares Meets Program Obfuscation)* in *EUROCRYPT 2018, Part II* (eds Nielsen, J. B. & Rijmen, V.) **10821** (Springer, Heidelberg, 2018), 649–679.
8. Barak, B., Hopkins, S. B., Jain, A., Kothari, P. & Sahai, A. *Sum-of-Squares Meets Program Obfuscation, Revisited* in *EUROCRYPT 2019, Part I* (eds Ishai, Y. & Rijmen, V.) **11476** (Springer, Heidelberg, May 2019), 226–250.
9. Bardet, M., Faugère, J.-C. & Salvy, B. *Complexity of Gröbner basis computation for Semi-regular Overdetermined sequences over F_2 with solutions in F_2* Research Report RR-5049 (INRIA, 2003). `https://hal.inria.fr/inria-00071534`.
10. Bardet, M., Faugère, J.-C. & Salvy, B. On the complexity of Gröbner basis computation of semi-regular overdetermined algebraic equations. `https://doi.org/10.1016/j.jsc.2015.12.001` (2004).
11. Bogdanov, A. & Qiao, Y. *On the Security of Goldreich's One-Way Function* in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques* (eds Dinur, I., Jansen, K., Naor, J. & Rolim, J.) (Springer Berlin Heidelberg, Berlin, Heidelberg, 2009), 392–405. ISBN: 978-3-642-03685-9. `https://doi.org/10.1007/978-3-642-03685-9_30`.
12. Buchmann, J. A., Ding, J., Mohamed, M. S. E. & Mohamed, W. S. A. E. *MutantXL: Solving Multivariate Polynomial Equations for Cryptanalysis* in *Symmetric Cryptography* (eds Handschuh, H., Lucks, S., Preneel, B. & Rogaway, P.) **9031** (Schloss Dagstuhl – Leibniz-Zentrum für Informatik,

Dagstuhl, Germany, 2009), 1–7. https://drops.dagstuhl.de/opus/volltexte/2009/1945.

13. Caminata, A. & Gorla, E. *Solving Multivariate Polynomial Systems and an Invariant from Commutative Algebra* in *Arithmetic of Finite Fields* (eds Bajard, J. C. & Topuzoğlu, A.) (Springer International Publishing, Cham, 2021), 3–36. ISBN: 978-3-030-68869-1. https://doi.org/10.1007/978-3-030-68869-1_1.
14. Caminata, A. & Gorla, E. Solving Degree, Last Fall Degree, and Related Invariants. *J. Symb. Comput.* **114,** 322–335. ISSN: 0747-7171. https://doi.org/10.1016/j.jsc.2022.05.001 (2023).
15. Charikar, M. & Wirth, A. *Maximizing Quadratic Programs: Extending Grothendieck's Inequality* in *45th FOCS* (IEEE Computer Society Press, Oct. 2004), 54–60.
16. Cheng, C.-M., Chou, T., Niederhagen, R. & Yang, B.-Y. *Solving Quadratic Equations with XL on Parallel Architectures* in *CHES 2012* (eds Prouff, E. & Schaumont, P.) **7428** (Springer, Heidelberg, Sept. 2012), 356–373.
17. Courtois, N., Klimov, A., Patarin, J. & Shamir, A. *Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations* in *EUROCRYPT 2000* (ed Preneel, B.) **1807** (Springer, Heidelberg, May 2000), 392–407.
18. Couteau, G., Dupin, A., Méaux, P., Rossi, M. & Rotella, Y. *On the Concrete Security of Goldreich's Pseudorandom Generator* in *ASIACRYPT 2018, Part II* (eds Peyrin, T. & Galbraith, S.) **11273** (Springer, Heidelberg, Dec. 2018), 96–124.
19. Diem, C. *The XL-Algorithm and a Conjecture from Commutative Algebra* in *ASIACRYPT 2004* (ed Lee, P. J.) **3329** (Springer, Heidelberg, Dec. 2004), 323–337.
20. Ding, J. & Schmidt, D. in *Number Theory and Cryptography: Papers in Honor of Johannes Buchmann on the Occasion of His 60th Birthday* (eds Fischlin, M. & Katzenbeisser, S.) 34–49 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2013). ISBN: 978-3-642-42001-6. https://doi.org/10.1007/978-3-642-42001-6_4.
21. Dodis, Y., Reyzin, L. & Smith, A. *Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data* in *EUROCRYPT 2004* (eds Cachin, C. & Camenisch, J.) **3027** (Springer, Heidelberg, May 2004), 523–540.
22. Dubois, V. & Gama, N. *The Degree of Regularity of HFE Systems* in *ASIACRYPT 2010* (ed Abe, M.) **6477** (Springer, Heidelberg, Dec. 2010), 557–576.
23. 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* (Association for Computing Machinery, Lille, France, 2002), 75–83. ISBN: 1581134843. https://doi.org/10.1145/780506.780516.

24. Faugére, J.-C. A new efficient algorithm for computing Gröbner bases (F4). *Journal of Pure and Applied Algebra* **139,** 61–88. ISSN: 0022-4049. https://www.sciencedirect.com/science/article/pii/S0022404999000055 (1999).
25. Goemans, M. X. & Williamson, D. P. Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming. *J. ACM* **42,** 1115–1145. ISSN: 0004-5411. https://doi.org/10.1145/227683.227684 (1995).
26. Goldreich, O. in *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation: In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan, Salil Vadhan, Avi Wigderson, David Zuckerman* (ed Goldreich, O.) 76–87 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011). ISBN: 978-3-642-22670-0. https://doi.org/10.1007/978-3-642-22670-0_10.
27. Håstad, J., Impagliazzo, R., Levin, L. A. & Luby, M. A Pseudorandom Generator from any One-way Function. *SIAM Journal on Computing* **28,** 1364–1396 (1999).
28. Huang, M.-D. A., Kosters, M. & Yeo, S. L. *Last Fall Degree, HFE, and Weil Descent Attacks on ECDLP* in *CRYPTO 2015, Part I* (eds Gennaro, R. & Robshaw, M. J. B.) **9215** (Springer, Heidelberg, Aug. 2015), 581–600.
29. Impagliazzo, R. & Wigderson, A. *P = BPP if E Requires Exponential Circuits: Derandomizing the XOR Lemma* in *29th ACM STOC* (ACM Press, May 1997), 220–229.
30. Ishai, Y., Kushilevitz, E., Ostrovsky, R. & Sahai, A. *Cryptography with constant computational overhead* in *40th ACM STOC* (eds Ladner, R. E. & Dwork, C.) (ACM Press, May 2008), 433–442.
31. 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* (Association for Computing Machinery, Virtual, Italy, 2021), 60–73. ISBN: 9781450380539. https://doi.org/10.1145/3406325.3451093.
32. Jain, A., Lin, H. & Sahai, A. *Indistinguishability Obfuscation from LPN over Fp, DLIN, and PRGs in NC0* in *Advances in Cryptology – EUROCRYPT 2022* (eds Dunkelman, O. & Dziembowski, S.) (Springer International Publishing, Cham, 2022), 670–699. ISBN: 978-3-031-06944-4. https://doi.org/10.1007/978-3-031-06944-4_23.
33. Lang, S. *Algebra* ISBN: 9781461300410 146130041X. https://doi.org/10.1007/978-1-4613-0041-0 (Springer, New York, NY, 2002).
34. Lazard, D. *Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations* in *Computer Algebra* (ed van Hulzen, J. A.) (Springer Berlin Heidelberg, Berlin, Heidelberg, 1983), 146–156. ISBN: 978-3-540-38756-5. https://doi.org/10.1007/3-540-12868-9_99.

35. Macaulay, F. The algebraic theory of modular systems. *Cambridge Mathematical Library* **xxxi.** https://doi.org/10.3792/chmm/1263317740 (1916).
36. 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 *Post-quantum cryptography, second international workshop, PQCRYPTO 2008* (eds Buchmann, J. & Ding, J.) (Springer, Heidelberg, Oct. 2008), 203–215.
37. Mossel, E., Shpilka, A. & Trevisan, L. *On e-Biased Generators in NC0* in *44th FOCS* (IEEE Computer Society Press, Oct. 2003), 136–145.
38. Nisan, N. & Wigderson, A. *Hardness vs. Randomness (Extended Abstract)* in *29th FOCS* (IEEE Computer Society Press, Oct. 1988), 2–11.
39. ODonnell, R. & Witmer, D. *Goldreich's PRG: Evidence for Near-Optimal Polynomial Stretch* in (June 2014), 1–12. ISBN: 978-1-4799-3626-7. https://doi.org/10.1109/CCC.2014.9.
40. Schwartz, J. T. Fast Probabilistic Algorithms for Verification of Polynomial Identities. *J. ACM* **27,** 701–717. ISSN: 0004-5411. https://doi.org/10.1145/322217.322225 (1980).
41. Siegenthaler, T. Correlation-immunity of nonlinear combining functions for cryptographic applications (Corresp.) *IEEE Transactions on Information Theory* **30,** 776–780. https://doi.org/10.1109/TIT.1984.1056949 (1984).
42. Sugita, M., Kawazoe, M. & Imai, H. Relation between the XL Algorithm and Gröbner Basis Algorithms. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **E89-A,** 11–18. ISSN: 0916-8508. https://doi.org/10.1093/ietfec/e89-a.1.11 (2006).
43. Viola, E. *The Sum of d Small-Bias Generators Fools Polynomials of Degree d* in *2008 23rd Annual IEEE Conference on Computational Complexity* (2008), 124–127. https://doi.org/10.1109/CCC.2008.16.
44. Yang, B.-Y. & Chen, J.-M. *All in the XL Family: Theory and Practice* in *Information Security and Cryptology – ICISC 2004* (eds Park, C.-s. & Chee, S.) (Springer Berlin Heidelberg, Berlin, Heidelberg, 2005), 67–86. ISBN: 978-3-540-32083-8. https://doi.org/10.1007/11496618_7.
45. 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.