Inferring sequences produced by elliptic curve generators using Coppersmith's methods

Thierry Mefenza, Damien Vergnaud

Inferring Sequences Produced by Elliptic Curve Generators using Coppersmith's Methods*

Thierry Mefenza[1], Damien Vergnaud[2,]

Abstract

We analyze the security of two number-theoretic pseudo-random generators based on elliptic curves: the *elliptic curve linear congruential generator* and the *elliptic curve power generator*. We show that these recursive generators are insecure if sufficiently many bits are output at each iteration (improving notably the prior cryptanalysis of Gutierrez and Ibeas from 2007). We present several theoretical attacks based on Coppersmith's techniques for finding small roots on polynomial equations. Our results confirm that these generators are not appropriate for cryptographic purposes.

Keywords.

Keywords: Pseudo-random generator, Elliptic curve linear congruential generator, Elliptic curve power generator, Coppersmith's methods, Elliptic curves

1. Introduction

In cryptography, a pseudo-random number generator is a polynomial-time deterministic algorithm which takes as input a short random seed and outputs a long sequence which is indistinguishable in polynomial time from a truly random sequence. Pseudo-random numbers have found numerous applications in the literature. For instance they are useful in cryptography for key generation, encryption and digital signature. Number-theoretic pseudorandom generators work by iterating an algebraic map F (public or private) over a residue ring $\mathbb{Z}_N$ (where N is usually a prime number or an RSA modulus) on a secret random initial seed value $s_0 \in \mathbb{Z}_N$ to compute the intermediate state values $s_{i+1} = F(s_i) \bmod N$ for $i \in \mathbb{N}$ and outputting (some consecutive bits of) the state value s_i at each iteration. The input s_0 of the generator (and possibly the description of F) is called the seed and the output is called the pseudorandom sequence. The case where F is an affine function is known as the *linear congruential generator*. This generator is efficient and has good statistical properties. Unfortunately, it is cryptographically insecure: Boyar [Boy89] proved that - with a sufficiently long run of the pseudorandom sequence - one can recover the seed in time polynomial in the bit-size of N. It was suggested to use a non-linear algebraic map F in order to avoid these attacks. The provably secure classical *power-generator* from [BBS86] uses the algebraic map is $F : x \longmapsto x^e \bmod N$ for some integer $e \in \mathbb{N}$ and outputs asymptotically only at most $O((\log\log N)/(\log e))$ bits per multiply modulo an RSA modulus N, and hence are too slow to be used in many practical applications. Even if one wants to use this generator with no proven security guarantees, it outputs at most $O(\log N/\log e)$ bits per multiply modulo N and for this reason, this generator is used only with a very small integer e (typically $e \in \{2, 3\}$).

In 1994, Hallgren [Hal94] proposed a pseudo-random number generator based on a subgroup of points of an elliptic curve defined over a prime finite field. This generator is known as the Linear Congruential

*A preliminary version of this work appeared in the proceedings of the conference *International Computing and Combinatorics Conference – COCOON 2016* [Mef16]. This is the full version.

Email addresses: thierrymefenza@yahoo.fr (Thierry Mefenza), damien.vergnaud@lip6.fr (Damien Vergnaud)

[1]Department of Mathematics, Faculty of Sciences, University of Yaoundé I, Yaoundé, Cameroon

[2]Sorbonne Université, CNRS, Laboratoire d'Informatique de Paris 6, LIP6, Paris, France and Institut Universitaire de France

Generator on Elliptic Curves (EC-LCG). Let p be a prime number (with $p \geqslant 5$) and let E be an elliptic curve defined over a prime finite field $\mathbb{F}_p$, that is a rational curve given by the following Weierstrass equation

$$E : y^2 = x^3 + ax + b$$

for some $a, b \in \mathbb{F}_p$ with $4a^3 + 27b^2 \neq 0$. It is well known that the set $E(\mathbb{F}_p)$ of $\mathbb{F}_p$-rational points (including the special point O at infinity) forms an Abelian group with an appropriate composition rule (denoted $\oplus$) where O is the neutral element. For a given point $G \in E(\mathbb{F}_p)$, the EC-LCG generates a sequence U_n of points defined by the relation:

$$U_n = U_{n-1} \oplus G = [n]G \oplus U_0, \quad n \in \mathbb{N}$$

where $U_0 \in E(\mathbb{F}_p)$ is the initial value or seed. We refer to G as the *composer* of the EC-LCG.

For a positive integer $e > 1$ and a point $G \in E(\mathbb{F}_p)$ of order ℓ with $gcd(e, \ell) = 1$, the Elliptic Curve Power Generator (EC-PG), introduced by Lange and Shparlinksi in [LS05], generates a sequence of points V_n defined by the relation:

$$V_n = [e]V_{n-1} = [e^n]G, \quad n \in \mathbb{N}$$

where $V_0 \in E(\mathbb{F}_p)$ is the initial value or seed.

The EC-LCG and the EC-PG provide a very attractive alternative to linear congruential generators and power generators and they have been extensively studied in the literature (see [GBS00, GL01, MS02, BD02, LS05, Shp05, HS02, IS09, Mer17] and the references therein). In cryptography, one may want to use the output of these generators as a keystream for encryption. One can notice that if two consecutive values U_n, U_{n+1} of the EC-LCG generator are revealed, it is easy to find U_0 and G (see also [Mer17] for the EC-PG). In order to use the produced sequences for cryptography, one should therefore output only some bits (e.g. the most significant ones) of each coordinate of U_n or V_n, $n \in \mathbb{N}$ in the hope that this makes the resulting output sequence difficult to predict. As for the classical power generator [BBS86, BVZ12], in cryptographic uses of the EC-PG, e is always assumed to be "small" (and does not grow with the security parameter). Indeed, the computation of V_n from V_{n-1} requires $\Omega(\log(e))$ multiplications modulo p and the generator outputs at most $O(\log(p))$ bits. The EC-PG is used only with a very small integer e (typically $e \in \{2, 3\}$) and it is therefore assumed to be known to the adversary (it this is not the case, the adversary can perform an exhaustive search on all possible e and this adds only a constant factor overhead to its computational complexity).

In this paper, we show that the EC-LCG and the EC-PG are insecure if sufficiently many bits are output at each stage. Therefore a secure use of these generators requires to output fewer bits at each iteration and the efficiency of the schemes is thus degraded. Our attacks used the well-known Coppersmith's methods for finding small roots on polynomial equations. These methods have been introduced in 1996 by Coppersmith for polynomial of one or two variables [Cop96a, Cop96b] and have been generalized to many variables. These methods have been used to infer many pseudo-random generators and to cryptanalyze many schemes in cryptography (see [BCTV16, BVZ12] and the references therein). In this paper we notably used such techniques to improve the previous bounds known on the security of the EC-LCG in the literature. Our improvements are theoretical since in practice, the cost of Coppersmith's method in our case is prohibitive because of large dimension of the lattice. However, they confirm that these generators are not appropriate for cryptographic purposes.

Prior work. In the cryptography setting, the initial value U_0 and the constants G, a and b may be kept secret. Gutierrez and Ibeas [GI07] consider two settings: the case where the *composer* G is known to the attacker and a, b are kept secret and the case where the *composer* G is unknown and a, b are kept secret. In the first case, they showed that the EC-LCG is insecure if a proportion of at most $1/6$ of the least significant bits of two consecutive values of the sequence is hidden. When the *composer* is unknown, they showed heuristically that the EC-LCG is insecure if a proportion of at most $1/46$ of the least significant bits of three consecutive values of the sequence is hidden. Their result is based on a lattice basis reduction attack, using a certain linearization technique. In some sense, their technique can be seen as a special case of the problem of finding small solutions of multivariate polynomial congruences. The Coppersmith's methods also tackle

the problem of finding small solutions of multivariate polynomial congruences. Gutierrez and Ibeas due to the special structure of the polynomials involved claimed that "the Coppersmith's methods does not seem to provide any advantages", and that "it may be very hard to give any precise rigorous or even convincing heuristic analysis of this approach". Our purpose in this paper is to tackle this issue.

Contributions of the paper. We predict the EC-LCG sequence and the EC-PG sequence using Coppersmith's method for calculating the small roots of multivariate polynomials modulo an integer. The method for multivariate polynomials is heuristic since it is not proven and may fail (but in practice it works most of the time). At the end of the Coppersmith's methods we use the methods from [BCTV16] to analyze the success condition.

In the case where the *composer* is known, we showed that the EC-LCG is insecure if a proportion of at most $1/5$ of the least significant bits of two consecutive values U_0 and U_1 of the sequence is hidden. This improves the previous bound of $1/6$ of Gutierrez and Ibeas. We further improve this result by considering several consecutive values of the sequence. We showed that the EC-LCG is insecure if a proportion of at most $3/11$ of the least significant bits of these values is hidden.

We also consider the case where some most significant bits of the abscissa of several points U_{kn} are given, with $n \in \mathbb{N}$ for some fixed integer k. In this case, the addition law on the elliptic curve gives us a system of polynomials of high degree whose roots are the hidden bits. We use summation polynomials [Sem04] on elliptic curves to obtain a system of polynomials of low degree. We then showed that the EC-LCG is insecure if a proportion of at most $1/8$ of the least significant bits of two values $X(U_0)$ and $X(U_k)$ is hidden, where $X(P)$ denotes the abscissa of the point P on the curve. This attack is the first cryptanalytic result on the EC-LCG when the attacker knows some most significant bits of non-consecutive values of the generator. We further improve this result by considering several values U_{kn}, $n \in \mathbb{N}$ of the sequence. We also show that the EC-LCG is insecure if a proportion of at most $1/4$ of the least significant bits of the abscissa of these values is hidden.

In the case where the *composer* is unknown, we showed that the EC-LCG is insecure if a proportion of at most $1/24$ of the least significant bits of two consecutive values U_0 and U_1 of the sequence is hidden. This improves significantly the (heuristic) previous bound $1/46$ of Gutierrez and Ibeas. We further improve this result by considering sufficiently many consecutive values of the sequence. We showed that the EC-LCG is insecure if a proportion of at most $1/8$ of the least significant bits of these values is hidden.

Finally, we also showed that the EC-PG is insecure if a proportion of at most $1/2e^2$ of the least significant bits of the abscissa of two consecutive values V_0 and V_1 of the sequence is hidden. We improve this bound by considering several consecutive values of the sequence and we showed that the EC-PG is insecure if a proportion of at most $1/e^2$ of the least significant bits of the abscissa of these values is hidden. To our knowledge no such results are known in the literature for the EC-PG.

The table below gives a comparison between our results and those known in the literature. It gives the bound of the proportion of least significant bits hidden from each value necessary to break the EC-LCG in (heuristic) polynomial time. The basic proportion corresponds to the case where the adversary knows bits coming from the minimum number of intermediate values leading to a feasible attack; while the asymptotic proportion corresponds to the case when the bits known by the adversary knows bits coming from arbitrary number of values.

Generator	Setting	Basic proportion		Asymptotic proportion	
		Prior result	Our result	Prior result	Our result
EC-LCG	known *composer* consecutive values	$1/6$ (proven)	$1/5$ (heuristic)	–	$3/11$ (heuristic)
	known *composer* non-consecutive values	–	$1/8$ (heuristic)	–	$1/4$ (heuristic)
	unknown *composer*	$1/46$ (heuristic)	$1/24$ (heuristic)	–	$1/8$ (heuristic)
EC-PG		–	$1/2e^2$ (heuristic)	–	$1/e^2$ (heuristic)

2. Preliminaries

In this section, we collect some statements about elliptic curves, Coppersmith's methods and analytic combinatorics.

2.1. Elliptic curves

Throughout this paper, let p be a prime number (with $p \geqslant 5$). We first recall the arithmetic of the group law $\oplus$ on elliptic curves defined by the Weierstrass equation (for more details on elliptic curves, we refer to [BSS99, Was08]). Let $E : y^2 = x^3 + ax + b$ be an elliptic curve over $\mathbb{F}_p$. For two points $P = (x_P, y_P) \in E(\mathbb{F}_p)$ and $Q = (x_Q, y_Q) \in E(\mathbb{F}_p)$, with $P, Q \neq O$, the addition law $\oplus$ is defined as $R = (x_R, y_R) = P \oplus Q$ where:

- If $x_P \neq x_Q$, then

$$x_R = m^2 - x_P - x_Q \bmod p, \quad y_R = m(x_P - x_R) - y_P \bmod p, \quad \text{where, } m = \frac{y_Q - y_P}{x_Q - x_P} \bmod p \qquad (1)$$

- If $x_P = x_Q$ but $y_P \neq y_Q$, then $R = O$
- If $P = Q$ and $y_P \neq 0$, then

$$x_R = m^2 - 2x_P \bmod p, \quad y_R = m(x_P - x_R) - y_P \bmod p, \quad \text{where, } m = \frac{3x_Q^2 + a}{2y_P} \bmod p$$

- If $P = Q$ and $y_P = 0$, then $R = O$.

2.2. Division polynomials of elliptic curves

With the notation from the previous paragraph, we recall some basic facts on division polynomials of elliptic curves (see again [Was08] and [BSS99] for details). They provide a way to calculate multiples of points on elliptic curves and to study the fields generated by torsion points. The *division polynomials* $\psi_m(X, Y) \in \mathbb{F}_p[X, Y]/(Y^2 - X^3 - AX - B)$, $m \geqslant 0$, are recursively defined by:

$$\begin{aligned}
\psi_0 &= 0 \\
\psi_1 &= 1 \\
\psi_2 &= 2Y \\
\psi_3 &= 3X^4 + 6AX^2 + 12BX - A^2 \\
\psi_4 &= 4Y(X^6 + 5AX^4 + 20BX^3 - 5A^2X^2 - 4ABX - 8B^2 - A^3) \\
\psi_{2m+1} &= \psi_m + 2\psi_m^3 - \psi_{m-1}\psi_{m+1}^3\,, \quad m \geqslant 2 \\
\psi_{2m} &= \psi_m(\psi_{m+2}\psi_{m-1}^2 - \psi_{m-2}\psi_{m+1}^2)/\psi_2\,, \quad m \geqslant 3,
\end{aligned}$$

where ψ_m is an abbreviation for $\psi_m(X, Y)$.

If m is odd, then $\psi_m(X, Y) \in \mathbb{F}_p[X]$ is univariate and if m is even then

$$\psi_m(X, Y) \in \psi_2(X, Y)\mathbb{F}_p[X] \text{ so } \psi_m(X, Y) \in 2Y\mathbb{F}_p[X].$$

Therefore, as $\psi_2^2(X, Y) = 4(X^3 + AX + B)$, we have $\psi_m^2(X, Y) \in \mathbb{F}_p[X]$ and $\psi_{m-1}(X, Y)\psi_{m+1}(X, Y) \in \mathbb{F}_p[X]$. In particular, we may write $\psi_{2m+1}(X)$ and $\psi_m^2(X)$.

As mentioned above, the division polynomials can be used to calculate multiples of a point on the elliptic curve E. Let $P = (x, y) \in E$ with $P \neq O$, then the abscissa of $[m]P$ is given by

$$\frac{\theta_m(x)}{\psi_m^2(x)}, \quad \text{where } \theta_m(X) = X\psi_m^2 - \psi_{m-1}\psi_{m+1}.$$

The zeros of the denominator $\psi_m^2(X)$ are exactly the abscissa of the non-trivial m-torsion points, i.e, the points $Q = (x, y) \in \overline{\mathbb{F}_p}^2 \setminus \{O\}$ on E with $[m]Q = O$. Note, that these points occur in pairs $Q = (x, y)$ and $-Q = (x, -y)$, which coincide only if $2Q = O$, i.e, if x is a zero of $\psi_2^2(X)$.

We recall that the group of m-torsion points $E[m]$, for an elliptic curve E defined over a field of characteristic p, is isomorphic to $(\mathbb{Z}/m\mathbb{Z})^2$ if $p \nmid m$ and to a proper subgroup of $(\mathbb{Z}/m\mathbb{Z})^2$ if $p \mid m$. If m is a power of p then $E[m]$ is either isomorphic to $(\mathbb{Z}/m\mathbb{Z})$ or to $\{O\}$. Accordingly, the degree of $\psi_m^2(X)$ is $m^2 - 1$ if $p \nmid m$ and strictly less than $m^2 - 1$ otherwise. In particular, for $p = 2$ and m a power of 2 we have $\deg(\psi_m^2) = m - 1$ if E is not supersingular and $\deg(\psi_m^2) = 0$ otherwise. By induction one can show that $\theta_m(X) \in \mathbb{F}_p[X]$ is monic of degree m^2.

2.3. Summation polynomials

With the same notation as above, for $n \in \mathbb{N}$, $n \geqslant 2$, we define introduce *n-th summation polynomial* $f_n = f_n(X_1, X_2, \ldots, X_n)$ introduced by Semaev in [Sem04] such that

$$f_n(x_1, \ldots, x_n) = 0$$

for $x_i \in \overline{\mathbb{F}_p}$ (the algebraic closure of $\mathbb{F}_p$ if and only if there exist $y_1, \ldots, y_n \in \overline{\mathbb{F}_p}$ such that $(x_1, y_1), \ldots, (x_n, y_n) \in E(\overline{\mathbb{F}_p})$ and

$$(x_1, y_1) \oplus \cdots \oplus (x_n, y_n) = O.$$

These polynomials have found interesting applications in cryptography (in particular for solving the discrete logarithm problem on elliptic curves defined finite fields, see [Die11] and references therein).

The following lemma gives a simple way for calculating them:

Lemma 1. *The n-th Semaev summation polynomial f_n may be defined by:*

$$\begin{aligned} f_2(X_1, X_2) &= X_1 - X_2 \\ f_3(X_1, X_2, X_3) &= (X_1 - X_2)^2 X_3^2 - 2\left((X_1 + X_2)(X_1X_2 + a) + 2a\right) X_3 + (X_1X_2 - a)^2 - 4b(X_1 + X_2) \\ f_n(X_1, \ldots, X_n) &= Res_X(f_{n-k}(X_1, \ldots, X_{n-k-1}, X), f_{k+2}(X_{n-k}, \ldots, X_n, X)), \quad n \geqslant 4 \text{ and } 1 \leqslant k \leqslant n-1 \,. \end{aligned}$$

The polynomial f_n is symmetric and of degree 2^{n-2} in each variable X_i for any $n \geqslant 3$. The polynomial f_n is absolutely irreducible and we have

$$f_n(X_1, \ldots, X_n) = f_{n-1}^2(X_1, \ldots, X_{n-1}) X_n^{2^{n-2}} + \ldots$$

2.4. Coppersmith's methods

In this section, we give a short description of Coppersmith's method for solving a multivariate modular polynomial system of equations modulo an integer N. We refer the reader to [JM06] for details and proofs.

2.4.1. Problem definition.

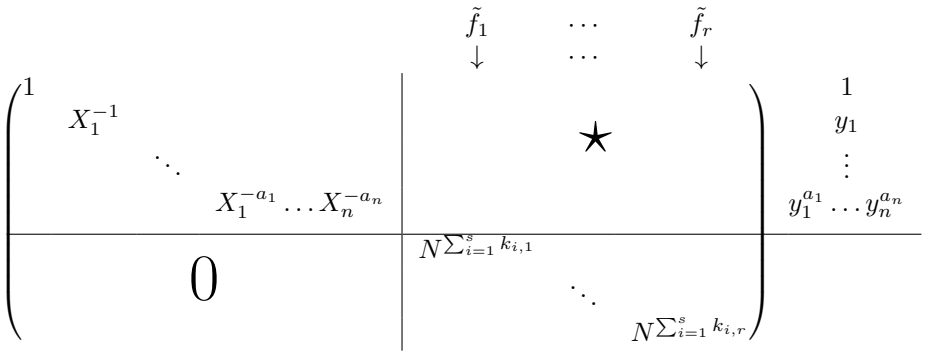
Let $f_1(y_1, \ldots, y_n), \ldots, f_s(y_1, \ldots, y_n)$ be irreducible multivariate polynomials defined over $\mathbb{Z}$, having a root $(x_1, \ldots, x_n)$ modulo a known integer N, namely $f_i(x_1, \ldots, x_n) \equiv 0 \mod N$. We want this root to be *small* in the sense that each of its components is bounded by a known value X_i. We also need to bound the sizes of X_i allowing to recover the desired root in polynomial time.

2.4.2. Polynomials collection.

In a first step, one generates a collection $\mathfrak{P}$ of polynomials $\{\tilde{f}_1, \ldots, \tilde{f}_r\}$ linearly independent having $(x_1, \ldots, x_n)$ as a root modulo N. Usually, multiples and powers of products of f_i for $i \in \{1, \ldots, s\}$ are chosen, namely $\tilde{f}_\ell = y_1^{\alpha_{1,\ell}} \cdots y_n^{\alpha_{n,\ell}} f_1^{k_{1,\ell}} \cdots f_s^{k_{s,\ell}}$ for some integers $\alpha_{1,\ell}, \ldots, \alpha_{n,\ell}, k_{1,\ell}, k_{s,\ell}$. Such polynomials satisfy the relation $\tilde{f}_\ell(x_1, \ldots, x_n) \equiv 0 \mod N^{\sum_{i=1}^s k_{i,\ell}}$, i.e., there exists an integer c_i such that $\tilde{f}_i(x_1, \ldots, x_n) = c_i N^{\sum_{j=1}^s k_{j,\ell}}$.

2.4.3. Matrix construction.

We denote as $\mathfrak{M}$ the set of monomials appearing in collection of polynomials $\mathfrak{P}$. Then each polynomial $\tilde{f}_i$ can be expressed as a vector with respect to a chosen order on $\mathfrak{M}$. We hence construct a matrix $\mathcal{M}$ as follows and we define as $\mathcal{L}$ the lattice generated by its rows:



On that figure, every row of the upper part is related to one monomial of $\mathfrak{M}$ (we assume in the figure that $\mathfrak{M}$ contains 1, y_1, and $y_1^{a_1} \dots y_n^{a_n}$ among other monomials). The left-hand side contains the bounds on these monomials (e.g., the coefficient $X_1^{-1}X_2^{-2}$ is put in the row related to the monomial $y_1y_2^2$). The right-hand side is formed by all vectors coming from $\mathfrak{P}$.

2.4.4. A short vector in a sublattice.

Let us now consider the row vector

$$r_0 = (1, x_1, \dots, x_1^{a_1} \dots x_n^{a_n}, -c_1, \dots, -c_r) \ .$$

By multiplying this vector by the matrix $\mathcal{M}$, one obtains:

$$s_0 = \left(1, \left(\frac{x_1}{X_1}\right), \dots, \left(\frac{x_1}{X_1}\right)^{a_1} \cdots \left(\frac{x_n}{X_n}\right)^{a_n}, 0, \dots, 0\right) \ .$$

The knowledge of $s_0 \in \mathcal{L}$ is sufficient to recover the root we are searching for and its norm is very small since $\|s_0\|_2 \leqslant \sqrt{\sharp\mathfrak{M}}$. Thus, the recovery of a small vector in $\mathcal{L}$, will likely lead to the recovery of the desired root $(x_1, \dots, x_n)$. To this end, we first restrict ourselves in a more appropriated subspace. Indeed, noticing that the last coefficients of s_0 are all null, we know that this vector belongs to a sublattice $\mathcal{L}'$ whose last coordinates are composed by zero coefficients. By doing elementary operations on the rows of $\mathcal{M}$, one can construct that sublattice and its determinant is the same as the one of $\mathcal{L}$.

2.4.5. Method conclusion.

From that point, one computes an LLL-reduction on the lattice $\mathcal{L}'$ and computes the Gram-Schmidt's orthogonalized basis $(b_1^\star, \dots, b_t^\star)$ of the LLL output basis $(b_1, \dots, b_t)$. Since s_0 belongs to $\mathcal{L}'$, this vector can be expressed as a linear combination of the $b_i^\star$'s. Consequently, if its norm is smaller than those of $b_t^\star$, then s_0 is orthogonal to $b_t^\star$. Extracting the coefficients appearing in $b_t^\star$, one can construct a polynomial p_1 defined over $\mathbb{Z}$ such that $p_1(x_1, \dots, x_n) = 0$. Repeating the same process with the vectors $b_{t-1}^\star, \dots, b_{t-n+1}^\star$ leads to the system $\{p_1(x_1, \dots, x_n) = 0, \dots, p_n(x_1, \dots, x_n) = 0\}$. Under the (heuristic) assumption that all created polynomials define an algebraic variety of dimension 0, the previous system can be solved (e.g., using elimination techniques such as Groebner basis) and the desired root recovered in polynomial time.

The conditions on the bounds X_i that make this method work are given by the following (simplified) inequation (see [JM06] for details):

$$\prod_{y_1^{k_1} \dots y_n^{k_n} \in \mathfrak{M}} X_1^{k_1} \cdots X_n^{k_n} < N^{\sum_{\ell=1}^{r} \sum_{i=1}^{s} k_{i,\ell}} \ . \tag{2}$$

For such techniques, the most complicated part is the choice of the collection of polynomials, what could be a really intricate task when working with multiple polynomials.

Remark 1. As noted above, the inequality 2 is simplified. In [JM06], there is an additionnal multiplicative term in the right-hand side that depends only on the dimension of the lattice and does not depend on N (and therefore only contributes to an error term as N grows with the security parameter). The non-simplified equation is

$$\prod_{y_1^{k_1}\dots y_n^{k_n}\in\mathfrak{M}} X_1^{k_1}\cdots X_n^{k_n} < \left(2^{-\omega(\omega-1)/4}\omega^{-\omega/2}\right) N^{\sum_{\ell=1}^{r}\sum_{i=1}^{s} k_{i,\ell}} \ . \tag{3}$$

where $\omega = \#\mathfrak{M}$ denotes the number of monomials.

2.5. Analytic combinatorics

In the following, we recall the analytic combinatorics methods from [FS09] to estimate the exponents of the bounds $X_1, \dots, X_n$ and of the modulo N on the monomials and polynomials appearing in the inequality (2) in Coppersmith's methods. Those methods can be used to compute the cardinalities of the sets $\mathfrak{P}$ and $\mathfrak{M}$. We used the same notations as in [BCTV16] and for more details of the methods the reader is referred to that paper.

To make things clear, we will explain the method with one simple example. Suppose we want to solve the equation: $f(y_1, \dots, y_n) = 0 \bmod N$, with $|y_i| \leqslant X_i$. We consider the set of polynomials

$$\begin{aligned}\mathfrak{P} = \{f_{\mathbf{k}} = y_1^{k_1}\dots y_n^{k_n} f^{k_\ell} \bmod N^{k_\ell} : 1 \leqslant k_\ell < t \\ \text{and } \deg(f_{\mathbf{k}}) = k_1 + \cdots + k_n + k_\ell e < te\}\,,\end{aligned}$$

The set of monomials appearing in the collection $\mathfrak{P}$ will usually look like

$$\mathfrak{M} = \{y_{\mathbf{k}} = y_1^{k_1}\dots y_n^{k_n} : 0 \leqslant \deg(y_{\mathbf{k}}) = k_1 + \cdots + k_n < te\} \ .$$

These considerations imply that for the final condition in Coppersmith's method (see Equation (2)), one needs to compute the values

$$\psi = \sum_{f_{\mathbf{k}}\in\mathfrak{P}} k_\ell \quad \text{and} \quad \forall i \in \{1,\dots,n\}, \quad \alpha_i = \sum_{y_{\mathbf{k}}\in\mathfrak{M}} k_i \ .$$

These values correspond to the exponent of N and X_i (for $i \in \{1,\dots,n\}$) in Equation (2) respectively. We show how to compute these sums ψ and α_i for polynomials in $\mathfrak{P}$ or $\mathfrak{M}$ of a certain degree.

We see $\mathfrak{P}$ (respectively $\mathfrak{M}$) as a combinatorial class with size function $S(f_{\mathbf{k}}) = \deg(f_{\mathbf{k}})$(respectively $S(y_{\mathbf{k}}) = \deg(y_{\mathbf{k}})$). We recall that a combinatorial class is a finite or countable set on which a size function is defined, satisfying the following conditions: (i) the size of an element is a non-negative integer and (ii) the number of elements of any given size is finite. We define another function χ, called a *parameter* function, such that $\chi(f_{\mathbf{k}}) = k_\ell$ (respectively $\chi(y_{\mathbf{k}}) = k_i$). This allows us to compute ψ (respectively α_i) as:

$$\psi = \chi_{<te}(\mathfrak{P}) = \sum_{a\in\mathfrak{P}:S(a)<te} \chi(a)$$

(respectively $\alpha_i = \chi_{<te}(\mathfrak{M}) = \sum_{a\in\mathfrak{P}:S(a)<te}\chi(a)$). To do so we should be able to compute given a combinatorial class $\mathfrak{A}$ ($\mathfrak{A} = \mathfrak{P}$ or $\mathfrak{A} = \mathfrak{M}$) with size function S and the parameter function χ,

$$\chi_{\leqslant p}(\mathfrak{A}) = \sum_{a\in\mathfrak{A}:S(a)\leqslant p} \chi(a) \ .$$

We proceed as follows:

1. We give another description of $\mathfrak{A}$ with respect to S and χ. This description associates to the combinatorial class an ordinary generating function (OGF) $F(z,u)$ (using Table 1, see [BCTV16] for details). When the class contains elements of different sizes (such as variables of degree 1 and polynomials of degree e), the variables in the OGF are represented by the atomic element $\mathcal{Z}$ and the polynomials by the element $\mathcal{Z}^e$, in order to take into account the degree of these polynomials. Then we "mark" the element useful for the parameter, with a new variable u. At this level we only know how to compute $\sum_{a\in\mathfrak{A}:S(a)=p}\chi(a)$. An easier way to compute $\chi_{\leqslant p}(\mathfrak{A})$ is to force all elements a of size less than or equal to p to be of size exactly p by adding enough times a *dummy* element y_0 such that $\chi(y_0)=0$. In our context of polynomials, the aim of the dummy variable y_0 is to homogenize the polynomial.

Table 1: Combinatorics constructions and their OGF

	Construction	OGF
Atomic class	$\mathcal{Z}$	$Z(z)=z$
Neutral class	ε	$E(z)=1$
Disjoint union	$\mathcal{A}=\mathcal{B}+\mathcal{C}$ (when $\mathcal{B}\cap\mathcal{C}=\emptyset$)	$A(z)=B(z)+C(z)$
Complement	$\mathcal{A}=\mathcal{B}\setminus\mathcal{C}$ (when $\mathcal{C}\subseteq\mathcal{B}$)	$A(z)=B(z)-C(z)$
Cartesian product	$\mathcal{A}=\mathcal{B}\times\mathcal{C}$	$A(z)=B(z)\cdot C(z)$
Cartesian exponentiation	$\mathcal{A}=\mathcal{B}^k=\mathcal{B}\times\cdots\times\mathcal{B}$	$A(z)=B(z)^k$
Sequence	$\mathcal{A}=\textsc{Seq}(\mathcal{B})=\varepsilon+\mathcal{B}+\mathcal{B}^2+\ldots$	$A(z)=\frac{1}{1-B(z)}$

2. We have:

$$\chi_{\leqslant}(\mathfrak{A})(z)=\sum_{p=0}^{+\infty}\chi_{\leqslant p}(\mathfrak{A})z^p=\left.\frac{\partial F(z,u)}{\partial u}\right|_{u=1},$$

3. Since Coppersmith's method is usually used in an asymptotic way, singularity analysis enables us to find the asymptotic value of the coefficients in an simple way by using the following theorem (see [FS09], page 392):

Theorem 2 (Transfer Theorem)**.** *Let $\mathfrak{A}$ be a combinatorial class with an ordinary generating function F regular enough such that there exists a value c verifying*

$$F(z)=\sum_{n=0}^{+\infty}F_n z^n\underset{z\to 1}{\sim}\frac{c}{(1-z)^\alpha}$$

for a non-negative integer α. The asymptotic value of the coefficient F_n is

$$F_n\underset{n\to\infty}{\sim}(cn^{\alpha-1})/(\alpha-1)!\ .$$

3. Predicting EC-LCG Sequences for Known Composer

Following [GI07], our results involve a parameter Δ which measures how well some values approximate the sequence elements. More precisely, we say that $W=(x_W,y_W)\in\mathbb{F}_p^2$ is a Δ-approximation to $U=(x_U,y_U)\in\mathbb{F}_p^2$ if there exist integers e,f satisfying: $|e|,|f|\leqslant\Delta$, $x_W+e=x_U$ and $y_W+f=y_U$.

For a given point $G\in E(\mathbb{F}_p)$, the EC-LCG generates a sequence U_n of points defined by the relation:

$$U_n=U_{n-1}\oplus G=[n]G\oplus U_0,\quad n\in\mathbb{N}$$

where $U_0\in E(\mathbb{F}_p)$ is the initial value or seed. We refer to G as the *composer* of the EC-LCG. In the cryptographic setting, the initial value $U_0=(x_0,y_0)$ and the constants G, a and b are supposed to be the secret key. In the following we infer the sequence output by the EC-LCG in the case where the composer G is known and the curve is kept secret.

We consider two settings: the case where the most significant bits of consecutive values U_n of the sequence is output and the case where the most significant bits of the abscissa of consecutive multiple values U_{kn} (for some fixed integer k) of the sequence is output. In the first case, we show that the generator is insecure if at least a proportion of $4/5$ of the most significant bits of two consecutive values U_0 and U_1 of the sequence is output. In the second case, We show that the generator is insecure if at least a proportion of $7/8$ of the most significant bits of two values $X(U_0)$ and $X(U_k)$ is output, $X(P)$ denoting the abscissa of the point P.

3.1. Information on consecutive inputs

Theorem 3. ***(two consecutive outputs)*** *Given Δ-approximations W_0, W_1 to two consecutive affine value U_0, U_1 produced by the EC-LCG, and given the value of the composer $G = (x_G, y_G)$. Under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed U_0 in heuristic polynomial time in $\log p$ as soon as $\Delta < p^{1/5}$*

Proof. We suppose $U_0 \notin \{-G, G\}$. Then, clearing denominators in (1), we can translate

$$U_1 = U_0 \oplus G$$

into the following identities in the field $\mathbb{F}_p$:

$$L_1 = L_1(x_0, y_0, x_1) = 0 \, mod \, p, \quad L_2 = L_2(x_0, y_0, x_1, y_1) = 0 \, mod \, p$$

where $U_0 = (x_0, y_0)$, $U_1 = (x_1, y_1)$ and

$$\begin{aligned} L_1 &= x_G^3 + x_1 x_G^2 - x_0 x_G^2 - 2x_1 x_G x_0 - x_G x_0^2 + x_0^3 + 2y_G y_0 + x_1 x_0^2 - y_G^2 - y_0^2, \\ L_2 &= y_1 x_G - y_1 x_0 - y_G x_0 + y_G x_1 - y_0 x_1 + y_0 x_G. \end{aligned}$$

Set $W_0 = (\alpha_0, \beta_0)$ and $W_1 = (\alpha_1, \beta_1)$. Then using the equalities $x_j = \alpha_j + e_j$ and $y_j = \beta_j + f_j$, for $j \in \{0, 1\}$, where $|e_j|, |f_j| < \Delta$ leads to the following polynomial system:

$$\begin{cases} f(e_0, e_1, f_0) = 0 \bmod p \\ g(e_0, e_1, f_0, f_1) = 0 \bmod p \ . \end{cases}$$

where

$$f(z_1, z_2, z_3) = A_1 z_1 + A_2 z_2 + A_3 z_3 + A_4 z_1^2 + A_5 z_1 z_2 + z_1^3 + z_1^2 z_2 - z_3^2 + A_6$$

and

$$g(z_1, z_2, z_3, z_4) = B_1 z_1 + B_2 z_2 + B_3 z_3 + B_4 z_4 + z_1 z_4 + z_2 z_3 + B_5$$

are polynomials whose coefficients A_i and B_i are functions of x_G, and the approximations values $\alpha_0, \alpha_1, \beta_0, \beta_1$. If we fix $u = z_1^3 + z_1^2 z_2 - z_3^2$ and $v = z_1 z_4 + z_2 z_3$, then the polynomial f becomes $f_1(z_1, z_2, z_3, u) = A_1 z_1 + A_2 z_2 + A_3 z_3 + A_4 z_1^2 + A_5 z_1 z_2 + u + A_6$ and g becomes $g_1(z_1, z_2, z_3, z_4, v) = B_1 z_1 + B_2 z_2 + B_3 z_3 + B_4 z_4 + v + B_5$.

Description of the attack. The adversary is therefore looking for the small solutions of the following modular multivariate polynomial system:

$$\begin{cases} f_1(z_1, z_2, z_3, u) = 0 \bmod p \\ g_1(z_1, z_2, z_3, z_4, v) = 0 \bmod p \ . \end{cases}$$

With $|z_j| < \Delta$, $|u| < X = \Delta^3$ and $|v| < Y = \Delta^2$. The attack consists in applying Coppersmith's methods for multivariate polynomials. From now, we use the following collection of polynomials (parameterized by some integer $t \in \mathbb{N}$):

$$\mathfrak{P} = \left\{ z_1^{j_1} \dots z_4^{j_4} f_1^{i_1} g_1^{i_2} \bmod p^{i_1+i_2} : \, i_1 + i_2 > 0 \, \text{and} \, j_1 + \dots + j_4 + 2i_1 + i_2 < 2t \right\}$$

The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ z_1^{i_1} z_2^{i_2} z_3^{i_3} z_4^{i_4} u^{i_5} v^{i_6} \bmod \Delta^{i_1+i_2+i_3+i_4} X^{i_5} Y^{i_6} : i_1 + \cdots + i_4 + 2i_5 + i_6 < 2t \right\}.$$

If we use for instance the monomial order lex (with $z_i < u < v$) on the set of monomials, then the leading monomial of f_1 is $LM(f_1) = u$ and $LM(g_1) = v$. Then the polynomials in $\mathfrak{P}$ are linearly independent since we have prohibited the multiplication by u and v.

Bounds for the polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(z_1^{j_1} \dots z_4^{j_4} f_1^{i_1} g_1^{i_2}) = j_1 + \cdots + j_4 + 2i_1 + i_2$ and the parameter function $\chi(z_1^{j_1} \dots z_4^{j_4} f_1^{i_1} g_1^{i_2}) = i_1 + i_2$. The degree of each variable z_i, u, v is 1, whereas the degree of f_1 is 2 and the degree of g_1 is 1. For the sake of simplicity, we can consider $0 \leqslant i_1 + i_2$, since the parameter function equals 0 for elements $z_1^{j_1} \dots z_4^{j_4} f_1^{i_1} g_1^{i_2}$ with $i_1 + i_2 = 0$.
We can describe $\mathfrak{P}$ as:

$$\prod_{i=1}^{4} \textsc{Seq}(Z) \times \textsc{Seq}(uZ^2) \times \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dummy value z_0.
This leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-z}\right)^5 \times \frac{1}{1-uz^2} \times \frac{1}{1-uz}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{z^2(1-z) + z(1-z^2)}{(1-z)^7(1-z^2)^2},$$

as $z \to 1$, $1 - z^n \sim n(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{3(1-z)}{4(1-z)^9} \sim \frac{3}{4(1-z)^8},$$

since $2t \sim 2t - 1$, this leads to:

$$\chi_{<2t}(\mathfrak{P}) \sim \frac{3}{4} \times \frac{(2t)^7}{7!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(z_1^{i_1} \dots z_4^{i_4} u^{i_5} v^{i_6}) = i_1 + \cdots + i_4 + 2i_5 + i_6$ and the parameter function $\chi(z_1^{i_1} \dots z_4^{i_4} u^{i_5} v^{i_6}) = i_1 + \cdots + i_4$. As z_1, z_2, z_3, z_4, u, v "count for" $1, 1, 1, 1, 2$ and 1 respectively in the condition of the set, we can describe $\mathfrak{M}$ as:

$$\textsc{Seq}(Z^2) \times \textsc{Seq}(Z) \times \prod_{i=1}^{4} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dummy value z_0.
Which leads to the generating function:

$$F(z,u) = \frac{1}{(1-z^2)(1-z)^2} \times \left(\frac{1}{1-uz}\right)^4.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{4z}{(1-z)^7(1-z^2)},$$

as $z \to 1$, $1 - z^n \sim n(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{2}{(1-z)^8},$$

since $2t \sim 2t - 1$, this leads to:

$$\chi_{<2t,\Delta}(\mathfrak{M}) \sim \frac{2(2t)^7}{7!}$$

Bounds for the monomials modulo X. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(z_1^{i_1}\dots z_4^{i_4}u^{i_5}v^{i_6}) = i_1+\dots+i_4+2i_5+i_6$ and the parameter function $\chi(z_1^{i_1}\dots z_4^{i_4}u^{i_5}v^{i_6}) = i_5$. As z_1, z_2, z_3, z_4, u, v "count for" $1, 1, 1, 1, 2$ and 1 respectively in the condition of the set, we can describe $\mathfrak{M}$ as:

$$\prod_{i=1}^{5} \textsc{Seq}(Z) \times \textsc{Seq}(uZ^2) \times \textsc{Seq}(Z),$$

where the last one is for the dummy value z_0.
Which leads to the generating function:

$$F(z,u) = \frac{1}{(1-z)^6} \times \left(\frac{1}{1-uz^2}\right).$$

This leads to:

$$\chi_{<2t,X}(\mathfrak{M}) \sim \frac{(2t)^7}{4\times 7!}$$

Bounds for the monomials modulo Y. We consider again the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(z_1^{i_1}\dots z_4^{i_4}u^{i_5}v^{i_6}) = i_1+\dots+i_4+2i_5+i_6$ and the parameter function $\chi(z_1^{i_1}\dots z_4^{i_4}u^{i_5}v^{i_6}) = i_6$. As z_1, z_2, z_3, z_4, u, v "count for" $1, 1, 1, 1, 2$ and 1 respectively in the condition of the set, we can describe $\mathfrak{M}$ as:

$$\prod_{i=1}^{4} \textsc{Seq}(Z) \times \textsc{Seq}(Z^2) \times \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dummy value z_0.
Which leads to the generating function:

$$F(z,u) = \frac{1}{(1-z)^5(1-z^2)} \times \left(\frac{1}{1-uz}\right).$$

This leads to:

$$\chi_{<2t,Y}(\mathfrak{M}) \sim \frac{(2t)^7}{2\times 7!}$$

Condition. If we denote by $\nu_1 = \chi_{<2t,\Delta}(\mathfrak{M})$, $\nu_2 = \chi_{<2t,X}(\mathfrak{M})$, $\nu_3 = \chi_{<2t,Y}(\mathfrak{M})$ and $\varepsilon = \chi_{<2t}(\mathfrak{P})$, the condition for Coppersmith's method is $p^\varepsilon > \Delta^{\nu_1}X^{\nu_2}Y^{\nu_3}$, ie $\Delta < p^{\frac{\varepsilon}{\nu_1+3\nu_2+2\nu_3}}$, where:

$$\frac{\varepsilon}{\nu_1+3\nu_2+2\nu_3} \sim \frac{\chi_{<2t}(\mathfrak{P})}{\chi_{<2t,\Delta}(\mathfrak{M})+3\chi_{<2t,X}(\mathfrak{M})+2\chi_{<2t,Y}(\mathfrak{M})} \sim \frac{1}{5},$$

this leads to the bound:

$$\Delta < p^{\frac{1}{5}}.$$

Complexity of the attack. The dimensions of the matrix used in Coppersmith methods depend on the cardinalities of the set of polynomials and monomials. To compute the cardinalities of the sets $\mathfrak{P}$ and $\mathfrak{M}$, we make use of the parameters functions $\chi(z_1^{j_1}\dots z_4^{j_4}f_1^{i_1}g_1^{i_2}) = 1$ and $\chi(z_1^{i_1}\dots z_4^{i_4}u^{i_5}v^{i_6}) = 1$. This leads to the generating functions:

$$F_1(z) = \left(\frac{1}{1-z}\right)^5 \times \left(\frac{1}{1-z^2}\times\frac{1}{1-z} - 1\right)$$

and

$$F_2(z) = \left(\frac{1}{1-z}\right)^5 \times \frac{1}{1-z^2}\times\frac{1}{1-z},$$

for $\mathfrak{P}$ and $\mathfrak{M}$ respectively.

We have:

$$F_1(z), F_2(z) \underset{z\to 1}{\sim} \frac{1}{2(1-z)^7},$$

which leads when t goes to infinity to the asymptotic bound:

$$\frac{1}{2} \times \frac{(2t)^6}{6!}$$

□

From the inequality (2), one can see that the attack works if

$$p^{\varepsilon} > \Delta^{\nu_1} X^{\nu_2} Y^{\nu_3},$$

with $I_1 = \{\mathbf{i} = (i_1, \ldots, i_6) | 0 \leqslant i_1 + \cdots + i_4 + 2i_5 + i_6 < 2t\}$, $I_2 = \{\mathbf{i} = (j_1, \ldots, j_4, i_1, i_2) |\, i_1 + i_2 > 0 \,\text{and}\, 0 \leqslant j_1 + \cdots + j_4 + 2i_1 + i_2 < 2t\}$,

$$\nu_1 = \sum_{\mathbf{i}\in I_1} i_1 + \cdots + i_4, \quad \nu_2 = \sum_{\mathbf{i}\in I_1} i_5, \quad \nu_3 = \sum_{\mathbf{i}\in I_1} i_6 \quad \text{and} \quad \varepsilon = \sum_{\mathbf{i}\in I_2} i_1 + i_2$$

. Putting $X = \Delta^3$ and $Y = \Delta^2$, we then have the following theoretical bound

$$\Delta < p^{\delta_{\text{theo}}},$$

where $\delta_{\text{theo}} = \frac{\varepsilon}{\nu_1 + 3\nu_2 + 2\nu_3}$. If one uses the non-simplified inequality (3), we only obtain

$$\Delta < \left(2^{-\omega(\omega-1)/4} \omega^{-\omega/2}\right) p^{\delta_{\text{theo}}}$$

where $\omega = \#\mathfrak{M}$ denotes the number of monomials. The "error term" is thus independent of p but so small that we can only conclude theoretically that the attack works only for very large p (several hundreds of bits even for only $t = 2$). However, our experiments show that the attack works for practical values of p (and even sometimes for $\Delta < p^{\delta_{\text{exp}}}$, where $\delta_{\text{exp}} > \delta_{\text{theo}}$. We give in the table below the cardinalities of the sets $\mathfrak{P}$, $\mathfrak{M}$ and the theoretical bound δ_{theo} for smaller t.

t	1	2	3	4	5	6	7	8	9
number of polynomials	1	27	188	776	2393	6111	13664	27672	51897
number of monomials	6	62	314	1106	3108	7476	16044	31548	57882
δ_{theo}	0.167	0.182	0.187	0.190	0.192	0.193	0.194	0.195	0.1953

Unfortunately, even if t is small, the constructed matrix is of huge dimension (since the number of monomials is quite large) and the computation which is theoretically polynomial-time becomes in practice prohibitive. These attacks are nethertheless good evidence of a weakness in this pseudo-random generator.

Experimental Results. We have implemented the attack in Sage 7.6 on a MacBook Air laptop computer (2,2 GHz Intel Core i7, 4 Gb RAM 1600 MHz DDR3, Mac OSX 10.10.5). Table 2 lists the dimension d of the lattice ($d = \sharp\mathfrak{P} + \sharp\mathfrak{m}$), the theoretical bound δ_{theo} and an experimental bound δ_{exp} for a m-bit prime p. We consider the family of polynomials $\mathfrak{P}$ with $t = 2$ (since for $t > 2$, the dimension of the lattice is huge). We ran several experiments for all parameters and Table 2 gives the average running time (in seconds) of the LLL algorithm, the Gram-Schmidt's orthogonalization algorithm and the Gröbner basis computation.

This bound improves the known bound $\Delta < p^{1/6}$. Next we further improve the previous bound and we show that the generator is insecure if at least a proportion of 8/11 of the most significant bits of a large number of consecutive values U_i of the sequence is output.

m	t	δ_{theo}	δ_{exp}	d	LLL time(s)	Gram-Schmidt's time(s)	Gröbner basis time(s)
128	2	0.182	0.20	89	2.908	20.910	0.567
256	2	0.182	0.12	89	2.957	46.157	6.474

Table 2: Predicting EC-LCG Sequences for Known Composer

Theorem 4. ***(more consecutive outputs)***
Given Δ-approximations W_0, W_1,...,W_n (for some integer $n > 1$) to $n+1$ consecutive affine values U_0, U_1,...,U_n produced by the EC-LCG, and given the value of the composer $G = (x_G, y_G)$. Under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed U_0 in polynomial time in $\log p$ as soon as $\Delta < p^{\frac{3n}{11n+4}}$

Proof. Let us assume, for instance that the adversary has access to $n+1$ Δ-approximations W_0, W_1,...,W_n of U_0, U_1,...,U_n produced by the EC-LCG. Then using the equalities $x_j = \alpha_j + e_j$ and $y_j = \beta_j + f_j$, for $j \in \{0, \ldots, n\}$, where $|e_j|, |f_j| < \Delta$ and $W_j = (\alpha_j, \beta_j)$ and $U_j = (x_j, y_j)$ leads to the following polynomial system:

$$\begin{cases} f_1'(e_0, e_1, f_0) = 0 \bmod p \\ g_1'(e_0, e_1, f_0, f_1) = 0 \bmod p \\ \vdots \\ f_n'(e_{n-1}, e_n, f_{n-1}) = 0 \bmod p \\ g_n'(e_{n-1}, e_n, f_{n-1}, f_n) = 0 \bmod p \ . \end{cases}$$

Where for $i \in \{1, \ldots, n\}$,

$$f_i'(z_{i-1}, z_i, z_{n+i}) = A_1 z_{i-1} + A_2 z_i + A_3 z_{n+i} + A_4 z_{i-1}^2 + A_5 z_{i-1} z_i + z_{i-1}^3 + z_{i-1}^2 z_i - z_{n+i}^2 + A_6$$

and

$$g_i'(z_{i-1}, z_i, z_{n+i}, z_{n+i+1}) = B_1 z_{i-1} + B_2 z_i + B_3 z_{n+i} + B_4 z_{n+i+1} + z_{i-1} z_{n+i+1} + z_i z_{n+i} + B_5$$

are polynomials whose coefficients A_i and B_i are functions of x_G, and the approximations values α_k, β_k, $(k \in \{i-1, i\})$. If we fix $u_i = z_{i-1}^3 + z_{i-1}^2 z_i - z_{n+i}^2$ and $v_i = z_{i-1} z_{n+i+1} + z_i z_{n+i}$, then the polynomial f_i' becomes $f_i(z_{i-1}, z_i, z_{n+i}, u_i) = A_1 z_{i-1} + A_2 z_i + A_3 z_{n+i} + A_4 z_{i-1}^2 + A_5 z_{i-1} z_i + u_i + A_6$ and g_i' becomes $g_i(z_{i-1}, z_i, z_{n+i}, z_{n+i+1}, v_i) = B_1 z_{i-1} + B_2 z_i + B_3 z_{n+i} + B_4 z_{n+i+1} + v_i + B_5$. The adversary is then looking for the solutions of the modular multivariate polynomial system:

$$\begin{cases} f_1(z_0, z_1, z_{n+1}) = 0 \bmod p \\ g_1(z_0, z_1, z_{n+1}, z_{n+2}) = 0 \bmod p \\ \vdots \\ f_n(z_{n-1}, z_n, z_{2n}) = 0 \bmod p \\ g_n(z_{n-1}, z_n, z_{2n}, z_{2n+1}) = 0 \bmod p \ . \end{cases}$$

With $|z_j| < \Delta$, $j \in \{0, \ldots, 2n+1\}$, $|u_i| < X = \Delta^3$ and $|v_i| < Y = \Delta^2$, $i = 1, \ldots, n$. We consider the following collection of polynomials:

$$\mathfrak{P} = \left\{ \begin{array}{l} \tilde{f}_{j_0,\ldots,j_{2n+1},i_1,\ldots,i_n,l_1,\ldots,l_n} = z_0^{j_0} \ldots z_{2n+1}^{j_{2n+1}} f_1^{i_1} \ldots f_n^{i_n} g_1^{l_1} \ldots g_n^{l_n} \bmod p^{i_1+l_1\cdots+i_n+l_n} \\ \text{s.t. } i_1 + l_1 + \cdots + i_n + l_n > 0 \text{ and } j_0 + \cdots + j_{2n+1} + 2(i_1 + \cdots + i_n) + l_1 + \cdots + l_n < 2t \end{array} \right\}.$$

The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ \begin{array}{l} z_0^{j_0} \dots z_{2n+1}^{j_{2n+1}} u_1^{i_1} v_1^{l_1} \dots u_n^{i_n} v_n^{l_n} \bmod \Delta^{j_0+\dots+j_{2n+1}} X^{i_0+\dots+i_n} Y^{l_0+\dots+l_n} \\ \text{s.t. } j_0 + \dots + j_{2n+1} + 2(i_1 + \dots + i_n) + l_1 + \dots + l_n < 2t \end{array} \right\}.$$

Bounds for the Polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(\tilde{f}_{j_0,\dots,j_{2n+1},i_1,\dots,i_n,l_1,\dots,l_n}) = j_0 + \dots + j_{2n+1} + 2(i_1 + \dots + i_n) + l_1 + \dots + l_n$ and the parameter function $\chi(\tilde{f}_{j_0,\dots,j_{2n+1},i_1,\dots,i_n,l_1,\dots,l_n}) = i_1 + l_1 + \dots + i_n + l_n$. We can describe $\mathfrak{P}$ as:

$$\prod_{i=0}^{2n+1} \textsc{Seq}(Z) \times \prod_{j=1}^{n} \textsc{Seq}(uZ^2) \times \prod_{k=1}^{n} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dummy variable.
This leads to the generating function:

$$F(z,u) = \frac{1}{(1-z)^{2n+3}} \times \frac{1}{(1-uz^2)^n} \times \frac{1}{(1-uz)^n}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{3n}{2^{n+1}(1-z)^{4n+4}},$$

since $2t \sim 2t-1$, we get:

$$\chi_{<2t}(\mathfrak{P}) \sim \frac{3n}{2^{n+1}} \times \frac{(2t)^{4n+3}}{(4n+3)!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function
$S(z_0^{j_0} \dots z_{2n+1}^{j_{2n+1}} u_1^{i_1} v_1^{l_1} \dots u_n^{i_n} v_n^{l_n}) = j_0 + \dots + j_{2n+1} + 2(i_1 + \dots + i_n) + l_1 + \dots + l_n$ and the parameter function $\chi(z_0^{j_0} \dots z_{2n+1}^{j_{2n+1}} u_1^{i_1} v_1^{l_1} \dots u_n^{i_n} v_n^{l_n}) = j_0 + \dots + j_{2n+1}$. We can describe $\mathfrak{M}$ as:

$$\prod_{i=1}^{n} \textsc{Seq}(Z^2) \times \prod_{i=1}^{n} \textsc{Seq}(Z) \times \prod_{i=0}^{2n+1} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dummy value y_0.
Which leads to the generating function:

$$F(z,u) = \frac{1}{(1-z^2)^n (1-z)^{n+1}} \times \frac{1}{(1-uz)^{2n+2}}.$$

As $z \to 1$, $1 - z^n \sim n(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{2n+2}{2^n(1-z)^{4n+4}},$$

since $2t \sim 2t-1$, this leads to:

$$\chi_{<2t,\Delta}(\mathfrak{M}) \sim \frac{2n+2}{2^n} \times \frac{(2t)^{4n+3}}{(4n+3)!}$$

***Bounds for the monomials modulo X.*.** We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function
$S(z_0^{j_0}\dots z_{2n+1}^{j_{2n+1}}u_1^{i_1}v_1^{l_1}\dots u_n^{i_n}v_n^{l_n}) = j_0+\dots+j_{2n+1}+2(i_1+\dots+i_n)+l_1+\dots+l_n$ and the parameter function $\chi(z_0^{j_0}\dots z_{2n+1}^{j_{2n+1}}u_1^{i_1}v_1^{l_1}\dots u_n^{i_n}v_n^{l_n}) = i_1+\dots+i_n$. We can describe $\mathfrak{M}$ as:

$$\prod_{i=0}^{2n+1}\textsc{Seq}(Z)\times\prod_{i=1}^{n}\textsc{Seq}(Z)\times\prod_{i=}^{n}\textsc{Seq}(uZ^2)\times\textsc{Seq}(Z),$$

where the last one is for the dummy value y_0.
Which leads to the generating function:

$$F(z,u)=\frac{1}{(1-z)^{3n+3}}\times\frac{1}{(1-uz^2)^n}.$$

This leads to:

$$\chi_{<2t,X}(\mathfrak{M})\sim\frac{n}{2^{n+1}}\times\frac{(2t)^{4n+3}}{(4n+3)!}$$

Bounds for the monomials modulo Y. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function
$S(z_0^{j_0}\dots z_{2n+1}^{j_{2n+1}}u_1^{i_1}v_1^{l_1}\dots u_n^{i_n}v_n^{l_n}) = j_0+\dots+j_{2n+1}+2(i_1+\dots+i_n)+l_1+\dots+l_n$ and the parameter function $\chi(z_0^{j_0}\dots z_{2n+1}^{j_{2n+1}}u_1^{i_1}v_1^{l_1}\dots u_n^{i_n}v_n^{l_n}) = l_1+\dots+l_n$. We can describe $\mathfrak{M}$ as:

$$\prod_{i=0}^{2n+1}\textsc{Seq}(Z)\times\prod_{i=1}^{n}\textsc{Seq}(Z^2)\times\prod_{i=}^{n}\textsc{Seq}(uZ)\times\textsc{Seq}(Z),$$

where the last one is for the dummy value y_0.
Which leads to the generating function:

$$F(z,u)=\frac{1}{(1-z)^{2n+3}(1-z^2)^n}\times\frac{1}{(1-uz)^n}.$$

This leads to:

$$\chi_{<2t,Y}(\mathfrak{M})\sim\frac{n}{2^n}\times\frac{(2t)^{4n+3}}{(4n+3)!}$$

Condition. If we denote by $\nu_1=\chi_{<2t,\Delta}(\mathfrak{M})$, $\nu_2=\chi_{<2t,X}(\mathfrak{M})$, $\nu_3=\chi_{<2t,Y}(\mathfrak{M})$ and $\varepsilon=\chi_{<2t}(\mathfrak{P})$, the condition for Coppersmith's method is $p^\varepsilon>\Delta^{\nu_1}X^{\nu_2}Y^{\nu_3}$, ie $\Delta<p^{\frac{\varepsilon}{\nu_1+3\nu_2+2\nu_3}}$, where:

$$\frac{\varepsilon}{\nu_1+3\nu_2+2\nu_3}\sim\frac{\chi_{<2t}(\mathfrak{P})}{\chi_{<2t,\Delta}(\mathfrak{M})+3\chi_{<2t,X}(\mathfrak{M})+2\chi_{<2t,Y}(\mathfrak{M})}\sim\frac{3n}{11n+4},$$

this leads to the expecting bound:

$$\Delta<p^{\frac{3n}{11n+4}}\underset{n\to\infty}{\to}\Delta<p^{3/11}.$$

□

3.2. Information on non-consecutive inputs

In this section, we consider an attacker with information on non-consecutive inputs. Our attacks only make use of the abscissa of the points generated by the EC-LCG. Given as above parameter Δ which measures how well some values approximate the sequence elements, we say that $w\in\mathbb{F}_p$ is a Δ-approximation to $x\in\mathbb{F}_p$ if there exist an integer e satisfying: $|e|\leqslant\Delta$ and $w+e=x$.

Theorem 5. ***(two outputs)*** *Given Δ-approximations w_0, w_k to two values $X(U_0)$, $X(U_k)$ produced by the EC-LCG. Under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed U_0 in polynomial time in $\log p$ as soon as $\Delta < p^{1/8}$.*

Proof. We set $U_0 = (x_0, y_0)$, $U_k = (x_k, y_k)$ and $G = (x_G, y_G)$. We then have the equalities:

$$x_i = w_i + e_i \quad \text{where} \quad |e_i| < \Delta,\ i \in \{0, k\}.$$

We have $U_0 - U_k = -kG$, thus $U_0 - U_k + kG = O$. Hence:

$$f_3(x_0, x_k, X(kG)) = 0,$$

where f_3 is the polynomial defined in section 2. Using the equalities $x_i = w_i + e_i$, $i \in \{0, k\}$ we obtain the polynomial equation:

$$f(e_0, e_k) = 0,$$

where $f(y_1, y_2) = f_3(w_0 + y_1, w_0 + y_2, X(kG))$. If we consider monomials with respect to lexicography ordering, we have $LM(f) = y_1^2 y_2^2$. f is a polynomial of degree 4. We consider the following collection of polynomials:

$$\mathfrak{P} = \{\tilde{f}_{j_1,j_2,i} = y_1^{j_1} y_6^{j_6} f^i \bmod p^i : i > 0 \quad \text{and} \quad j_1 + j_2 + 4i < 4t \\ \text{and } (j_1 < 2 \vee j_2 < 2)\},$$

One can check that the polynomials $\tilde{f}_{j_1,j_2,i}$ are linearly independent since $LM(f) \neq y_1^{j_1} y_2^{j_2}$ for each $\tilde{f}_{j_1,j_2,i}$ from $\mathfrak{P}$. The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ z_1^{j_1} z_2^{j_2} \bmod \Delta^{j_1+j_2} : j_1 + j_2 < 4t \right\},$$

Bounds for the Polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(\tilde{f}_{j_1,j_2,i}) = j_1 + j_2 + 4i$ and the parameter function $\chi(\tilde{f}_{j_1,j_2,i}) = i$. Since the degree of each variable z_i is 1 and the degree of f is 4, we can describe $\mathfrak{P}$ as:

$$\textsc{Seq}(uZ^4) \times \big((\varepsilon + Z)(\textsc{Seq}(Z) + Z^2\textsc{Seq}(Z))\big) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value y_0.
This leads to the generating function:

$$F(z, u) = \frac{(1+z)(1+z^2)}{(1-z)^2} \times \frac{z^4}{1 - uz^4}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u, z)\right|_{u=1} = \frac{z^4(1+z)(1+z^2)}{(1-z)^2(1-z^4)^2}$$

as $z \to 1$, $1 - z^4 \sim 4(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u, z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{1}{4(1-z)^4},$$

since $4t \sim 4t - 1$, this leads to:

$$\chi_{<4t}(\mathfrak{P}) \sim \frac{1}{4} \times \frac{(4t)^3}{3!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(y_1^{j_1}y_2^{j_2}) = j_1 + j_2$ and the parameter function $\chi(y_1^{j_1}y_2^{j_2}) = j_1 + j_2$. Since the degree of each z_i is 1, we can then described $\mathfrak{M}$ as:

$$\prod_{i=1}^{2} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value y_0.
Which leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-uz}\right)^2 \times \frac{1}{1-z}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{2z}{(1-z)^4},$$

as $z \to 1$, $1 - z^n \sim n(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{2}{(1-z)^4},$$

since $4t \sim 4t - 1$, this leads to:

$$\chi_{<4t}(\mathfrak{M}) \sim \frac{2(3t)^3}{3!}$$

Condition. If we denote by $\nu = \chi_{<4t}(\mathfrak{P})$, and $\varepsilon = \chi_{<4t}(\mathfrak{M})$, the condition for Coppersmith's method is $p^\nu > \Delta^\varepsilon$, ie $\Delta < p^{\frac{\nu}{\varepsilon}}$, where:

$$\frac{\nu}{\varepsilon} \sim \frac{\chi_{<4t}(\mathfrak{P})}{\chi_{<4t}(\mathfrak{M})} \sim \frac{1}{8},$$

this leads to the expecting bound:

$$\Delta < p^{\frac{1}{8}}.$$

□

We can further improve the previous bound and we show that the generator is insecure if at least a proportion of 1/4 of the most significant bits of a large number of abscissa of the values U_{ki} of the sequence is output.

Theorem 6. ***(more outputs)*** *Given Δ-approximations w_0, w_k,...,w_{kn} to $n+1$ values $X(U_0)$, $X(U_k)$,..., $X(U_{kn})$ produced by the EC-LCG. Under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed U_0 in polynomial time in $\log p$ as soon as $\Delta < p^{\frac{n}{4(n+1)}}$.*

Proof. We set $U_{kt} = (x_{kt}, y_{kt})$, for $t = 0, \ldots, n$ and $G = (x_G, y_G)$. We then have the equalities:

$$x_i = w_i + e_i \quad \text{where} \quad |e_i| < \Delta,\ i \in \{0, k, \ldots, nk\}.$$

We have $U_{kt} - U_{k(t+1)} = -kG$, for $t = 0, \ldots, n-1$. Thus $U_{kt} - U_{k(t+1)} + kG = O$. Hence:

$$f_3(x_{tk}, x_{k(t+1)}, X(kG)) = 0,$$

where f_3 is the polynomial defined in section 2. Using the equalities $x_i = w_i + e_i$, $i \in \{0, k, \ldots, kn\}$ we obtain the polynomial system:

$$f_j(e_{(j-1)k}, e_{jk}) = 0, \quad j \in \{1, \ldots, n\}$$

where $f_j(y_{j-1}, y_j) = f_3(w_{(j-1)k} + y_{j-1}, w_{jk} + y_j, X(kG))$. If we consider monomials with respect to lexicography ordering, we have $LM(f_k) = y_{j-1}^2 y_j^2$. f_j is a polynomial of degree 4. We consider the following collection of polynomials:

$$\mathfrak{P} = \{\tilde{f}_{j_0,\ldots,j_n,i_k} = y_0^{j_0} \ldots y_n^{j_n} f_k^{i_k} \bmod p^{i_k} : \quad k \in \{1,\ldots,n\};\ (j_{k-1} < 2 \vee j_k < 2) \\ (i_k > 0) \text{ and } (j_0 + \cdots + j_n + 4i_k) < 4t\},$$

One can check that the polynomials $\tilde{f}_{j_0,\ldots,j_n,i_k}$ are linearly independent. The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ z_0^{j_0} \ldots z_n^{j_n} \bmod \Delta^{j_0+\cdots+j_n} : j_0 + \cdots + j_n < 4t \right\}$$

,

Bounds for the Polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(\tilde{f}_{j_0,\ldots,j_n,i_k}) = j_0 + \cdots + j_n + 4i_k$ and the parameter function $\chi(\tilde{f}_{j_0,\ldots,j_n,i_k}) = i_k$. Since the degree of each variable z_i is 1 and the degree of f is 4, we can describe $\mathfrak{P}$ as:

$$\sum_{k=1}^{n} \textsc{Seq}(uZ^4) \times \big((\varepsilon + Z)(\textsc{Seq}(Z) + Z^2\textsc{Seq}(Z))\big) \times \prod_{j=0, j\neq k-1,k}^{n} \textsc{Seq}(Z) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value z_0.
This leads to the generating function:

$$F(z,u) = \sum_{k=1}^{n} \frac{(1+z)(1+z^2)}{(1-z)^{n+1}} \times \frac{1}{1-uz^4}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{nz^4(1+z)(1+z^2)}{(1-z)^{n+1}(1-z^4)^2}$$

as $z \to 1$, $1 - z^4 \sim 4(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{n}{4(1-z)^{n+3}},$$

since $4t \sim 4t - 1$, this leads to:

$$\chi_{<4t}(\mathfrak{P}) \sim \frac{n}{4} \times \frac{(4t)^{n+2}}{(n+2)!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(y_1^{j_1} y_2^{j_2}) = j_0 + \cdots + j_n$ and the parameter function $\chi(y_0^{j_0} \ldots y_n^{j_n}) = j_0 + \cdots + j_n$. Since the degree of each z_i is 1, we can then described $\mathfrak{M}$ as:

$$\prod_{i=0}^{n} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value z_0.
Which leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-uz}\right)^{n+1} \times \frac{1}{1-z}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{2z}{(1-z)^{n+3}},$$

as $z \to 1$, $1 - z^n \sim n(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{n+1}{(1-z)^{n+3}},$$

since $4t \sim 4t-1$, this leads to:

$$\chi_{<4t}(\mathfrak{M}) \sim \frac{(n+1)(4t)^{n+2}}{(n+2)!}$$

Condition. If we denote by $\nu = \chi_{<4t}(\mathfrak{P})$, and $\varepsilon = \chi_{<4t}(\mathfrak{M})$, the condition for Coppersmith's method is $p^\nu > \Delta^\varepsilon$, ie $\Delta < p^{\frac{\nu}{\varepsilon}}$, where:

$$\frac{\nu}{\varepsilon} \sim \frac{\chi_{<4t}(\mathfrak{P})}{\chi_{<4t}(\mathfrak{M})} \sim \frac{n}{4(n+1)},$$

this leads to the expecting bound:

$$\Delta < p^{\frac{n}{4(n+1)}} \underset{n\to\infty}{\to} \Delta < p^{\frac{1}{4}}.$$

□

4. Predicting EC-LCG Sequences for Unknown Composer

In this section, we infer the EC-LCG in the case where the composer G is unknown and the curve is kept secret. In the following, We show that the generator is insecure if at least a proportion of $23/24$ of the most significant bits of three consecutive values U_0 and U_1 and U_2 of the sequence is output.

Theorem 7. ***(three consecutive outputs)*** *Given Δ-approximations W_0, W_1, W_2 to three consecutive affine values U_0, U_1, U_2 produced by the EC-LCG. Under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed U_0 and the composer G in polynomial time in $\log p$ as soon as $\Delta < p^{1/24}$.*

Proof. We set $U_0 = (x_0, y_0)$, $U_1 = (x_1, y_1)$, $U_2 = (x_2, y_2)$, $W_0 = (\alpha_0, \beta_0)$, $W_1 = (\alpha_1, \beta_1)$ and $W_2 = (\alpha_2, \beta_2)$. We then have the equalities:

$$x_i = \alpha_i + e_i,\ y_j = \beta_j + f_j, \quad \text{where} \quad |e_i|,\ |f_i| < \Delta,\ i \in \{0,1,2\}. \tag{4}$$

We also have:

$$\begin{cases} y_0^2 = x_0^3 + ax_0 + b \\ y_1^2 = x_1^3 + ax_1 + b \\ y_2^2 = x_2^3 + ax_2 + b \ . \end{cases}$$

Eliminating the curve parameters a, b and assuming that $U_2 \neq \pm U_1$(that is, $x_2 \neq x_1$), we obtain the following equation:

$$y_2^2(x_0 - x_1) + x_2^3(x_1 - x_0) + x_0^3(x_2 - x_1) + y_0^3(x_1 - x_2) + x_1^3(x_0 - x_2) + y_1^2(x_2 - x_0) = 0$$

Using the equalities (3), leads to the equation:

$$f(e_0, e_1, e_2, f_0, f_1, f_2) = 0 \bmod p$$

where f is a polynomial of degree 4 whose coefficients are functions of $\alpha_0, \alpha_1, \alpha_2$, β_0, β_2, and β_2.

Description of the attack. The adversary is therefore looking for the solutions smaller than Δ of the following modular multivariate polynomial equation:

$$f(z_1, \ldots, z_6) = 0 \bmod p$$

The attack consists in applying Coppersmith's methods as in the former subsection. If we consider monomials with respect to lexicographic order, then the leading monomial of f is $z_1^3 z_2$. From now on, we use the following collection of polynomials:

$$\mathfrak{P} = \{\tilde{f}_{j_1,\ldots,j_6,i} = z_1^{j_1} \ldots z_6^{j_6} f^i \bmod p^i : i > 0 \quad \text{and} \quad j_1 + \cdots + j_6 + 4i < 4t \quad \text{and } (0 \leqslant j_1 < 3 \vee j_2 = 0)\},$$

One can check that the polynomials $\tilde{f}_{j_1,\ldots,j_6,i}$ are linearly independent since $LM(f) \neq z_1^{j_1} \ldots z_6^{j_6}$ for each $\tilde{f}_{j_1,\ldots,j_6,i}$ from $\mathfrak{P}$. The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ z_1^{j_1} \ldots z_6^{j_6} \bmod \Delta^{j_1+\cdots+j_6} : j_1 + \cdots + j_6 < 4t \right\}.$$

Bounds for the Polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(\tilde{f}_{j_1,\ldots,j_6,i}) = j_1 + \cdots + j_6 + 4i$ and the parameter function $\chi(\tilde{f}_{j_1,\ldots,j_6,i}) = i$. Since the degree of each variable z_i is 1 and the degree of f is 4, we can describe $\mathfrak{P}$ as:

$$\prod_{i=1}^{4} \textsc{Seq}(Z) \times \textsc{Seq}(uZ^4) \times \left((\underbrace{\varepsilon + Z + Z^2}_{z_1})(\underbrace{\varepsilon + Z\textsc{Seq}(Z)}_{z_2}) + \underbrace{Z^3\textsc{Seq}(Z)}_{z_1} \right) \times \textsc{Seq}(Z),$$

where the last one is for the dummy value z_0.
This leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-z}\right)^5 \times \frac{1}{1-uz^4} \times \left((1+z+z^2)(1+z/(1-z)) + \frac{z^3}{1-z} \right).$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{1+z+z^2+z^3}{(1-z)^6} \times \frac{z^4}{(1-z^4)^2}$$

as $z \to 1$, $1 - z^4 \sim 4(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{1}{4(1-z)^8},$$

since $4t \sim 4t - 1$, this leads to:

$$\chi_{<4t}(\mathfrak{P}) \sim \frac{1}{4} \times \frac{(4t)^7}{7!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(z_1^{j_1} \ldots z_6^{j_6}) = j_1 + \cdots + j_6$ and the parameter function $\chi(z_1^{j_1} \ldots z_6^{j_6}) = j_1 + \cdots + j_6$. Since the degree of each z_i is 1, we can then described $\mathfrak{M}$ as:

$$\prod_{i=1}^{6} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dummy value z_0.
Which leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-uz}\right)^6 \times \frac{1}{1-z}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{6z}{(1-z)^8},$$

as $z \to 1$, $1 - z^n \sim n(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{6}{(1-z)^8},$$

since $4t \sim 4t - 1$, this leads to:

$$\chi_{<4t}(\mathfrak{M}) \sim \frac{6(3t)^7}{7!}$$

Condition. If we denote by $\nu = \chi_{<4t}(\mathfrak{P})$, and $\varepsilon = \chi_{<4t}(\mathfrak{M})$, the condition for Coppersmith's method is $p^\nu > \Delta^\varepsilon$, ie $\Delta < p^{\frac{\nu}{\varepsilon}}$, where:

$$\frac{\nu}{\varepsilon} \sim \frac{\chi_{<4t}(\mathfrak{P})}{\chi_{<4t}(\mathfrak{M})} \sim \frac{1}{24},$$

this leads to the expecting bound:

$$\Delta < p^{\frac{1}{24}}.$$

Complexity of the attack. To compute the cardinalities of the sets $\mathfrak{P}$ and $\mathfrak{M}$, we make used of the parameters functions $\chi(\tilde{f}_{j_1,\ldots,j_6,i}) = 1$ and $\chi(z_1^{j_1} \ldots z_6^{j_6}) = 1$. This leads to the generating functions:

$$F_1(z) = \left(\frac{1}{1-z}\right)^5 \times \frac{z^4}{1-z^4} \times \frac{1+z+z^2+z^3}{1-z}$$

and

$$F_2(z) = \frac{1}{(1-z)^7},$$

for $\mathfrak{P}$ and $\mathfrak{M}$ respectively.
Asymptotic bounds. We have:

$$F_1(z), F_2(z) \underset{z\to 1}{\sim} \frac{1}{(1-z)^7},$$

which leads when t goes to infinity to the asymptotic bound:

$$\frac{(4t)^6}{6!}$$

Concrete bounds. We give in the table below the cardinalities of the sets $\mathfrak{P}$ and $\mathfrak{M}$ for smaller t.

t	1	2	3	4	5
number of polynomials	0	84	1716	12376	54264
number of monomials	84	1716	12376	54264	177100

□

This bound improves the known bound $\Delta < p^{1/46}$. Next, we further improve the previous bound and we show that the generator is insecure if at least a proportion of 7/8 of the most significant bits of an infinite consecutive values U_i of the sequence is output.

Theorem 8. ***(more consecutive outputs)***
Given Δ-approximations W_0, W_1,...,W_{n+1} (for some integer $n > 1$) to $n+2$ consecutive affine values U_0, U_1,...,U_{n+1} produced by the EC-LCG. Under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed U_0 and the composer G in polynomial time in $\log p$ as soon as $\Delta < p^{n/4(2n+4)}$.

Proof. Let us assume, for instance that the adversary has access to $n+1$ Δ-approximations $W_0, W_1,\ldots,W_{n+1}$ of $U_0, U_1,\ldots,U_{n+1}$ produced by the EC-LCG. Then using the equalities $x_j = \alpha_j + e_j$ and $y_j = \beta_j + f_j$, for $j \in \{0,\ldots,n\}$, where $|e_j|, |f_j| < \Delta$ and $W_j = (\alpha_j, \beta_j)$ and $U_j = (x_j, y_j)$ and the fact that $y_j^2 = x_j^3 + ax_j + b$, $j \in \{0,\ldots,n+1\}$ and eliminating the curve parameters from three consecutive points U_j, U_{j+1}, U_{j+2},$j \in \{0,\ldots,n-1\}$ leads to the following polynomial system:

$$\begin{cases} f_1(e_0, e_1, e_2, f_0, f_1, f_2) = 0 \bmod p \\ \vdots \\ f_n(e_{n-1}, e_n, e_{n+1}, f_{n-1}, f_n, f_{n+1}) = 0 \bmod p \ . \end{cases}$$

Where f_j are polynomials of degrees 4 and $LM(f_i) = z_{i-1}^3 z_i$. The adversary is then looking for the solutions of the modular multivariate polynomial system:

$$\begin{cases} f_1(z_0, z_1, z_2, z_{n+2}, z_{n+3}, z_{n+4}) = 0 \bmod p \\ \vdots \\ f_n(z_{n-1}, z_n, z_{n+1}, z_{2n+1}, z_{2n+2}, z_{2n+3}) = 0 \bmod p \ . \end{cases}$$

We consider the following collection of polynomials:

$$\mathfrak{P} = \left\{ \begin{array}{l} \tilde{f}_{j_0,\ldots,j_{2n+3},\alpha_i} = z_0^{j_0} \ldots z_{2n+3}^{j_{2n+3}} f_i^{\alpha_i} \bmod p^{\alpha_i} \\ \text{s.t. } i \in \{1,\ldots,n\};\ (j_{i-1} < 3 \vee j_i = 0) \\ (\alpha_i > 0) \text{ and } (j_0 + \cdots + j_{2n+3} + 4\alpha_i) < 4t \end{array} \right\}.$$

The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ z_0^{j_0} \ldots z_{2n+3}^{j_{2n+3}} \bmod \Delta^{j_0+\cdots+j_{2n+3}} : j_0 + \cdots + j_{2n+3} < 4t \right\},$$

Bounds for the Polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(\tilde{f}_{j_0,\ldots,j_{2n+3},\alpha_i}) = j_0 + \cdots + j_{2n+3} + 4\alpha_i$ and the parameter function $\chi(\tilde{f}_{j_0,\ldots,j_{2n+3},\alpha_i}) = \alpha_i$. We can describe $\mathfrak{P}$ as:

$$\sum_{i=1}^{n} \prod_{\substack{j=0 \\ j\in\{i-1,i\}}}^{2n+3} \textsc{Seq}(Z) \times \textsc{Seq}(uZ^4) \times \left((\varepsilon + Z + Z^2)(\varepsilon + Z\textsc{Seq}(Z)) + Z^3\textsc{Seq}(Z) \times \varepsilon\right) \times \textsc{Seq}(Z),$$

where the last one is for the dummy variable.
This leads to the generating function:

$$F(z,u) = \left(\frac{1}{(1-z)^{2n+3}} \times \frac{1}{1-uz^4} \right) \times n \left((1 + z + z^2)(1 + z/(1-z)) + \frac{z^3}{1-z} \right).$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{n}{4(1-z)^{2n+6}},$$

since $4t \sim 4t - 1$, this leads to:

$$\chi_{<4t}(\mathfrak{P}) \sim \frac{n}{4} \times \frac{(4t)^{2n+5}}{(2n+5)!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(z_0^{j_0}\dots z_{2n+3}^{j_{2n+3}}) = j_0+\dots+j_{2n+3}$ and the parameter function $\chi(z_0^{j_0}\dots z_{2n+3}^{j_{2n+3}}) = j_0+\dots+j_{2n+3}$. We can describe $\mathfrak{M}$ as:

$$\prod_{i=1}^{2n+4} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dummy variable.
This leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-uz}\right)^{2n+4} \times \frac{1}{1-z}.$$

We get

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{2n+4}{(1-z)^{2n+6}},$$

since $4t \sim 4t-1$, this leads to:

$$\chi_{<4t}(\mathfrak{M}) \sim (2n+4) \times \frac{(4t)^{2n+5}}{(2n+5)!}$$

Condition. If we denote by $\nu = \chi_{<4t}(\mathfrak{P})$, and $\varepsilon = \chi_{<4t}(\mathfrak{M})$, the condition for Coppersmith's method is $p^\nu > \Delta^\varepsilon$, ie $\Delta < p^{\frac{\nu}{\varepsilon}}$, where:

$$\frac{\nu}{\varepsilon} \sim \frac{\chi_{<4t}(\mathfrak{P})}{\chi_{<4t}(\mathfrak{M})} \sim \frac{n}{4(2n+4)},$$

This leads to the expecting bound:

$$\Delta < p^{\frac{n}{4(2n+4)}} \underset{n\to\infty}{\to} \Delta < p^{1/8}.$$

□

5. Predicting the Elliptic curve power generator

We show that one can predict the sequences generated by the EC-PG in the case where the constants a, b and e are known. We show that the generator is insecure if at least a proportion of $1-\frac{1}{2e^2}$ of the most significant bits of two consecutive values $X(V_0)$ and $X(V_1)$ is output.

Theorem 9. ***(two consecutive outputs)*** *Given Δ-approximations w_0, w_1 to two consecutive values $X(V_0)$, $X(V_0)$ produced by the EC-PG and under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed V_0 in heuristic polynomial time in $\log p$ as soon as $\Delta < p^{\frac{1}{2e^2}}$*

Proof. We put $V_0 = (x_0, y_0)$, $V_1 = (x_1, y_1)$. We have $x_1 = \frac{\theta_e(x_0)}{\psi_e^2(x_0)}$ since $V_1 = eV_0$. Using the equalities $x_0 = w_0 + \alpha_0$ and $x_1 = w_1 + \alpha_1$ with $\alpha_i < \Delta$, we have $f(\alpha_1, \alpha_0) = 0$, where $f(y_1, y_2) = (y_1+w_1)\psi_e^2(y_2+w_0) - \theta_e(y_2+w_0)$. We are looking for small modular modulo p. If we monomials with respect to lexicography ordering, then the leading monomial of f is $y_1 y_2^{e^2-1}$. f is a polynomial of degree e^2. We consider the following collection of polynomials (parameterized by some integer $t \in \mathbb{N}$):

$$\mathfrak{P} = \left\{ \begin{array}{l} \widetilde{f}_{j_1,j_2,i} = y_1^{j_1} y_2^{j_2} f^i \bmod p^i : i > 0 \,\text{and}\, j_1+j_2+e^2 i < e^2 t \\ \text{and}\, (j_1 = 0 \vee 0 \leqslant j_2 \leqslant e^2-2) \end{array} \right\}.$$

One can check that the polynomials $\widetilde{f}_{j_1,j_2,i}$ are linearly independent since $LM(\widetilde{f}_{j_1,j_2,i}) \neq y_1^{j_1} y_2^{j_2}$ for each $\widetilde{f}_{j_1,j_2,i}$. The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ y_1^{j_1} y_2^{j_2} \bmod \Delta^{j_1+j_2} : j_1+j_2 < e^2 t \right\}.$$

Bounds for the Polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(\tilde{f}_{j_1,j_2,i}) = j_1 + j_2 + e^2 i$ and the parameter function $\chi(\tilde{f}_{j_1,j_2,i}) = i$. Since the degree of each variable z_i is 1 and the degree of f is e^2, we can describe $\mathfrak{P}$ as:

$$\textsc{Seq}(uZ^{e^2}) \times \left((\underbrace{\varepsilon + Z + \cdots + Z^{e^2-2}}_{y_2})(\underbrace{\varepsilon + Z\textsc{Seq}(Z)}_{y_1}) + \underbrace{Z^{e^2-1}\textsc{Seq}(Z)}_{y_2} \right) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value y_0.
This leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-z}\right)^2 \times \frac{1}{1-uz^{e^2}} \times \left(1 + z + \cdots + z^{e^2-1}\right).$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{z^{e^2}(1 + z + \cdots + +z^{e^2-1})}{(1-z)^2(1-z^{e^2})^2}$$

as $z \to 1$, $1 - z^{e^2} \sim e^2(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{1}{e^2(1-z)^4},$$

since $e^2 t \sim e^2 t - 1$, this leads to:

$$\chi_{<e^2 t}(\mathfrak{P}) \sim \frac{1}{e^2} \times \frac{(e^2 t)^3}{3!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(z_1^{j_1} \dots z_6^{j_6}) = j_1 + \cdots + j_6$ and the parameter function $\chi(y_1^{j_1} y_2^{j_2}) = j_1 + j_2$. Since the degree of each z_i is 1, we can then described $\mathfrak{M}$ as:

$$\prod_{i=1}^{2} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value y_0.
Which leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-uz}\right)^2 \times \frac{1}{1-z}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{2z}{(1-z)^4},$$

which leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{2}{(1-z)^4},$$

since $e^2 t \sim e^2 t - 1$, this leads to:

$$\chi_{<e^2 t}(\mathfrak{M}) \sim \frac{2(e^2 t)^3}{3!}$$

Condition. If we denote by $\nu = \chi_{<e^2t}(\mathfrak{P})$, and $\varepsilon = \chi_{<e^2t}(\mathfrak{M})$, the condition for Coppersmith's method is $p^\nu > \Delta^\varepsilon$, ie $\Delta < p^{\frac{\nu}{\varepsilon}}$, where:

$$\frac{\nu}{\varepsilon} \sim \frac{\chi_{<e^2t}(\mathfrak{P})}{\chi_{<e^2t}(\mathfrak{M})} \sim \frac{1}{2e^2},$$

this leads to the expecting bound:

$$\Delta < p^{\frac{1}{2e^2}}.$$

□

Theorem 10. ***(more consecutive outputs)*** *Given Δ-approximations w_0, w_1,...,w_n (for some integer $n > 1$) to $n+1$ consecutive values $X(V_0)$, $X(V_1)$,..., $X(V_1)$ produced by the EC-PG and under the heuristic assumption that all created polynomials we get by applying Coppersmith's method with the polynomial set $\mathfrak{P}$ below define an algebraic variety of dimension 0, one can recover the seed V_0 in heuristic polynomial time in $\log p$ as soon as $\Delta < p^{\frac{n}{(n+1)e^2}}$*

Proof. We put $V_i = (x_i, y_i)$, $i \in \{0, \ldots, n\}$. We have $x_{j+1} = \frac{\theta_e(x_j)}{\psi_e^2(x_j)}$ since $V_{j+1} = eV_j$ for $j \in \{0, \ldots, n-1\}$. Using the equalities $x_i = w_i + \alpha_i$, $i \in \{0, \ldots, n\}$ with $\alpha_i < \Delta$, we have $f_j(\alpha_j, \alpha_{j-1}) = 0$, for $j \in \{1, \ldots, n\}$ where $f_j(y_{j-1}, y_j) = (y_{j-1} + w_j)\psi_e^2(y_j + w_{j-1}) - \theta_e(y_j + w_{j-1})$. We are then looking for small modular modulo p. We use the Coppersmith's methods to recover the desired solution in polynomial time. If we monomials with respect to lexicography ordering, then the leading monomial of each f_j is $y_{j-1}y_j^{e^2-1}$. f_j is a polynomial of degree e^2. We consider the following collection of polynomials (parameterized by some integer $t \in \mathbb{N}$):

$$\mathfrak{P} = \left\{ \begin{array}{l} \widetilde{f}_{j_1,\ldots,j_n,i_k} = y_1^{j_1} \ldots y_n^{j_n} f_k^{i_k} \bmod p^{i_k} : i_k > 0, \quad k \in \{1, \ldots, n\}, \quad \text{and} \\ j_1 + \cdots + j_n + e^2 i_k < e^2 t \quad \text{and} \quad (j_{k-1} = 0 \vee 0 \leqslant j_k \leqslant e^2 - 2) \end{array} \right\}.$$

One can check that the polynomials $\widetilde{f}_{j_1,\ldots,j_n,i_k}$ are linearly independent since $LM(\widetilde{f}_{j_1,\ldots,j_n,i_k}) \neq y_1^{j_1} \ldots y_n^{j_n}$ for each $\widetilde{f}_{j_1,\ldots,j_n,i_k}$. The list of monomials appearing within this collection can be described as:

$$\mathfrak{M} = \left\{ y_0^{j_0} \ldots y_n^{j_n} \bmod \Delta^{j_0+\cdots+j_n} : j_0 + \cdots + j_n < e^2 t \right\}.$$

Bounds for the Polynomials modulo p. We consider the set $\mathfrak{P}$ as a combinatorial class, with the size function $S(\tilde{f}_{j_1,\ldots,j_n,i_k}) = j_1 + \cdots + j_2 + e^2 i_k$ and the parameter function $\chi(\tilde{f}_{j_1,\ldots,j_n,i_k}) = i_k$. Since the degree of each variable z_i is 1 and the degree of f_k is e^2, we can describe $\mathfrak{P}$ as:

$$\sum_{k=1}^n \textsc{Seq}(uZ^{e^2}) \times \left(\underbrace{(\varepsilon + Z + \cdots + Z^{e^2-2})}_{y_k} \underbrace{(\varepsilon + Z\textsc{Seq}(Z))}_{y_{k-1}} + \underbrace{Z^{e^2-1}\textsc{Seq}(Z)}_{y_k} \right) \times \prod_{j=0, j\neq k, k-1}^n \textsc{Seq}(Z) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value z_0.
This leads to the generating function:

$$F(z,u) = \sum_{k=1}^n \left(\frac{1}{1-z} \right)^{n+1} \times \frac{1 + z + \cdots + z^{e^2-1}}{1 - uz^{e^2}}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{nz^{e^2}(1 + z + \cdots + +z^{e^2-1})}{(1-z)^{n+1}(1-z^{e^2})^2}$$

as $z \to 1$, $1 - z^{e^2} \sim e^2(1-z)$ leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{n}{e^2(1-z)^{n+3}},$$

since $e^2 t \sim e^2 t - 1$, this leads to:

$$\chi_{<e^2t}(\mathfrak{P}) \sim \frac{n}{e^2} \times \frac{(e^2t)^{n+2}}{(n+2)!}$$

Bounds for the monomials modulo Δ. We consider the set $\mathfrak{M}$ as a combinatorial class, with the size function $S(z_1^{j_1} \dots z_6^{j_6}) = j_1 + \cdots + j_6$ and the parameter function $\chi(y_0^{j_0} \dots y_n^{j_2}) = j_1 + \cdots + j_n$. Since the degree of each y_i is 1, we can then described $\mathfrak{M}$ as:

$$\prod_{i=0}^{n} \textsc{Seq}(uZ) \times \textsc{Seq}(Z),$$

where the last one is for the dumming value z_0.
Which leads to the generating function:

$$F(z,u) = \left(\frac{1}{1-uz}\right)^{(n+1)} \times \frac{1}{1-z}.$$

We have

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} = \frac{(n+1)z}{(1-z)^{n+3}},$$

which leads to:

$$\left.\frac{\partial F}{\partial u}(u,z)\right|_{u=1} \underset{z\to 1}{\sim} \frac{n+1}{(1-z)^(n+3)},$$

since $e^2t \sim e^2t - 1$, this leads to:

$$\chi_{<e^2t}(\mathfrak{M}) \sim \frac{(n+1)(e^2t)^(n+2)}{(n+2)!}$$

Condition. If we denote by $\nu = \chi_{<e^2t}(\mathfrak{P})$, and $\varepsilon = \chi_{<e^2t}(\mathfrak{M})$, the condition for Coppersmith's method is $p^\nu > \Delta^\varepsilon$, ie $\Delta < p^{\frac{\nu}{\varepsilon}}$, where:

$$\frac{\nu}{\varepsilon} \sim \frac{\chi_{<e^2t}(\mathfrak{P})}{\chi_{<e^2t}(\mathfrak{M})} \sim \frac{n}{(n+1)e^2},$$

this leads to the expecting bound:

$$\Delta < p^{\frac{n}{(n+1)e^2}} \underset{n\to\infty}{\to} \Delta < p^{\frac{1}{e^2}}.$$

□

6. Conclusion

We analyzed the security of the Elliptic Curve Linear Congruential Generator (EC-LCG) and of the Elliptic Curve Power Generator (EC-PG). In the case where the *composer* is known, we showed that the EC-LCG is insecure if at least a proportion of 8/11 of the most significant bits of an arbitrary large number of consecutive values U_i of the sequence is output. We also tackle the case where the most significant bits of an arbitrary large number of non consecutive values (namely the most significant bits of the abscissa of values U_{ki} for some fixed integer k) of the sequence is output and we showed that the EC-LCG is insecure if at least a proportion of 3/4 of the most significant bits is output. Furthermore, we consider the cryptographic setting where the *composer* is unknown and we showed that this generator is insecure if at least a proportion of 7/8 of the most significant bits of an arbitrary large number of consecutive values U_i of the sequence is output. Finally, we showed that the EC-PG is insecure if a proportion of at least $1 - 1/e^2$ of the most significant bits of the abscissa of an arbitrary large number of consecutive values V_i of the sequence is output. However, our results are theoretical since in practice, the performance of Coppersmith's method in our attacks is prohibitive because of large dimension of the constructed lattice but they are good evidences of the weaknesses of these generators.

References

[BBS86]] L. Blum, M. Blum, and M. Shub, A simple unpredictable pseudo-random number generator, SIAM J. Comp., Vol. 15 (1986) pp. 364–383.

[BCTV16] F. Benhamouda, C. Chevalier, A. Thillard, and D. Vergnaud. Easing coppersmith methods using analytic combinatorics: Applications to public-key cryptography with weak pseudorandomness. In C. -M. Cheng, et al, editors, *PKC 16*, PartII, volume 9615 of *Lect. Notes Comput. Sci.*, pages 36–66, 2016.

[BD02] P. Beelen, and J. Doumen. Pseudorandom sequences from elliptic curves. Finite fields with applications to coding theory. *Cryptography and related areas.Springer-Verlag, Berlin, pages 37–52, 2002.*

[BSS99] I. F. Blake, G. Seroussi, and N. P. Smart. *Elliptic curves in cryptography.* Cambridge: Cambridge University Press, 1999.

[BVZ12] A. Bauer, D. Vergnaud, and J-C. Zapalowicz. Inferring sequences produced by nonlinear pseudorandom number generators using Coppersmith's methods. In M. Fischlin, J. Buchmann, and M. Manulis, editors, *PKC 12*, volume 7293 of *Lect. Notes Comput. Sci.*, pages 609–626, 2012.

[Boy89] J. Boyar. Inferring sequences produced by a linear congruential generator missing low-order bits. Journal of Cryptology 1(3), 177–184 (1989)

[Cop96a] D. Coppersmith. Finding a small root of a univariate modular equation. In U. M. Maurer, editor, *EUROCRYPT 96*, volume 1070 of *Lect. Notes Comput. Sci.*, pages 155–165, 1996.

[Cop96b] D. Coppersmith. Finding a small root of a bivariate integer equation; factoring with high bits known. In U. M. Maurer, editor, *EUROCRYPT 96*, volume 1070 of *Lect. Notes Comput. Sci.*, pages 178–189, 1996.

[Die11] C. Diem. On the discrete logarithm problem in elliptic curves *Compos. Math.* 147 (1), 75104, 2011

[FS09] P. Flajolet and R. Sedgewick. *Analytic Combinatorics.* Cambridge University Press (January 2009).

[GBS00] G. Gong, T. A. Berson, and D. A. Stinson. Elliptic curve pseudorandom sequence generators. volume 1758 of *Lect. Notes Comput. Sci.*, pages 34–49, 2000.

[GI07] J. Gutierrez and A. Ibeas. Inferring sequences produced by a linear congruential generator on elliptic curves missing high-order bits. *Design Code Cryptogr*, 45, 199–212, 2007.

[GL01] G. Gong and C. C. Y. Lam. Linear recursive sequences over elliptic curves. *In: Proc. intern. conf. on sequences and their applications, Bergen 2001.* Springer-Verlag, London, pages 182–196, 2001.

[Hal94] S. Hallgren. Linear congruential generators over elliptic curves. *Preprint CS-94-143, Dept. of Comp. Sci., 1994.*

[HS02] F. Hess and I. E. Shparlinski. On the linear complexity and multidimensional distribution of congruential generators over elliptic curves. *Design Code Cryptogr*, 35:111–117, 2005.

[JM06] E. Jochemsz, and A. May. A strategy for finding roots of multivariate polynomials with new applications in attacking RSA variants. In X. Lai, K. Chen, editors, *ASIACRYPT 06*, volume 4284 of *Lect. Notes Comput. Sci.*, pages 267–282, 2006.

[LS05] T. Lange and I. E. Shparlinski. Certain exponential sums and random walks on elliptic curves. *Canad. J. Math*, 57 (2005), 338–350.

[Mef16] T. Mefenza. Inferring sequences produced by a linear congruential generator on elliptic curves using Coppersmith's methods. In T. N. Dinh, M. T. Thai, editors, Computing and Combinatorics *COCOON 16*, volume 9797 of *Lect. Notes Comput. Sci.*, pages 293–304, 2016.

[Mer17] L. Merai. Predicting the elliptic curve congruential generator. *Appl. Algebr. Eng. Comm.* 28 (3), (2017), 193–203.

[MS02] E. Mahassni and I. E. Shparlinski. On the uniformity of distribution of congruential generators over elliptic curves. *In: Proc. intern. conf. on sequences and their applications.* Bergen 2001. Springer-Verlag, London, pages 257–264, 2002.

[Sem04] I. A. Semaev. Summation polynomials and the discrete logarithm problem on elliptic curves. *Cryptology ePrint Archive*, Report 2004/031, 2004.

[IS09] I. E. Shparlinski. Pseudorandom Number Generators from Elliptic Curves. *Contemporary Mathematics., volume 477, 2009.*

[Shp05] I. E. Shparlinski. Pseudorandom points on elliptic curves over finite fields. *Preprint*, 2005.

[Was08] L. C. Washington. *Elliptic curves. Number theory and cryptography. 2nd ed.* Boca Raton, FL: Chapman and Hall/CRC, 2nd ed. edition, 2008.