Attacking the Linear Congruential Generator on Elliptic Curves via Lattice Techniques

Jaime Gutierrez
Universidad de Cantabria
39071 Santander, Spain
jaime.gutierrez@unican.es

***Abstract*—In this communication we study the linear congruential generator on elliptic curves from the cryptographic point of view. We show that if sufficiently many of the most significants bits of the *composer* and of two consecutive values of the sequence are given, then one can recover the *seed* and the *composer* (even in the case where the elliptic curve is private). Our results are based on lattice reduction techniques and improve some recent approaches of the same security problem.**

***Index Terms*—Pseudorandom congruential generators, Cryptography, Lattice reduction, Elliptic curves.**

I. Introduction

A PseudoRandom Number Generator(PRNG) is a deterministic algorithm that, once initialized with some random value (called the seed), outputs a sequence that appears random, in the sense that an observer who does not know the value of the seed cannot distinguish the output from that of a (true) random bit generator. PRNG have important applications on simulations (e.g. for the Monte Carlo method), electronic games (e.g. for procedural generation), and cryptography. Good statistical properties are a vital requirement for the output of a PRNG. Cryptographic applications require the output not to be predictable from earlier outputs, and more elaborate algorithms, which do not inherit the linearity of simpler PRNGs, are needed.

There is a vast literature devoted to generating pseudorandom numbers using arithmetic of finite field and residue rings, see [36], [37], [44]. In 1994, Hallgreen [22] proposed a pseudorandom number generator based on the group of points of an elliptic curve defined over a prime finite field.

For a prime p, denote by $\mathbb{F}_p$ the field of p elements and always assume that it is represented by the set $\{0, 1, \ldots, p-1\}$. Accordingly, sometimes, where obvious, we treat elements of $\mathbb{F}_p$ as integer numbers in the above range.

Let E be an elliptic curve defined over $\mathbb{F}_p$ given by an *affine Weierstrass equation*, which for $\gcd(p, 6) = 1$ takes form

$$Y^2 = X^3 + aX + b, \tag{1}$$

for some $a, b \in \mathbb{F}_p$ with $4a^3 + 27b^2 \neq 0$.

We recall that the set $E(\mathbb{F}_p)$ of $\mathbb{F}_p$-rational points forms an abelian group, with the *point at infinity* $\mathcal{O}$ as the neutral element of this group (which does not have affine coordinates).

For a given point $G \in E(\mathbb{F}_p)$ the **Linear Congruential Generator on Elliptic Curves, EC-LCG** is a sequence U_n of pseudorandom numbers defined by the relation

$$U_n = U_{n-1} \oplus G = nG \oplus U_0, \quad n = 1, 2, \ldots, \tag{2}$$

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

It is clear that the period of the sequence (2) is equal to the order of G. The EC-LCG provides a very attractive alternative to linear and non-linear congruential generators with many applications to cryptography and it has been extensively studied in the literature, see [4], [14], [18], [19], [22], [23], [38], [39].

In the cryptographic setting, the initial value $U_0 = (x_0, y_0)$ and the constants G, a, and b are assumed to be the secret key, and we want to use the output of the generator as a stream cipher. Of course, if two consecutive values U_n are revealed, it is almost always easy to find U_0 and G. So, we output only the most significant bits of each U_n in the hope that this makes the resulting output sequence difficult to predict.

It is known that not too many bits can be output at each stage: the Linear Congruential Generator on Elliptic Curves is unfortunately (heuristically for unknown composer G and polynomial time) predictable if sufficiently many bits of its consecutive elements are revealed, see [21] and [33].

Now, we are formalising the results. Assume that the sequence (U_n) is not known, but for some n, approximations W_j of two consecutive values U_{n+j}, $j = 0, 1$ are given. The results involve another parameter Δ which measures how well the values W_j approximate the terms U_{n+j}. This parameter is assumed to vary independently of p subject to satisfying the inequality $\Delta < p$ (and is not involved in the complexity estimates of our algorithms). 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 exists integers e, f satisfying:

$$|e|, |f| \leq \Delta, \ x_W + e = x_U, \ y_W + f = y_U.$$

The case where Δ grows like a fixed power p^δ where $0 < \delta < 1$ corresponds to the situation where a positive proportion δ of the least significant bits of terms of the output sequence remain hidden.

The paper [21] shows an algorithm to recover the seed U_0 in deterministic polynomial time if $\Delta < p^{1/6}$ and G is public. The paper in [33] can recover 'heuristically' the seed U_0 if $\Delta < p^{1/5}$ and G is also public. On the other hand, the empirical results in [21] indicate that the threshold $p^{1/6}$ is more accurate than $p^{1/5}$, at least for primes p such that $\log_2(p) < 1000$. It seems that one of the reason is the constants hidden in the asymptotic reasoning.

In this paper, we deal in the special case when we also have an approximation to composer G. We show that given Δ if sufficiently many of the most significant bits of G and of two consecutive values U_n, U_{n+1} of the EC-LCG are given, one can recover 'heuristically' the seed U_0 and the composer G (even in the case where the elliptic curve is private) if $\Delta < p^{1/9}$.

The approach of the presented paper is similar to [21], but the equations involved are much more complex, and we are not able to provide a rigorous result.

In principle, we can not obtain any approximation to composer G from any approximations to two consecutive values U_n, U_{n+1} of the EC-LCG, because the elliptic curve group operation.

This suggests that for cryptographic applications EC-LCG should be used with great care.

For the *linear congruential generator* similar problems have been introduced by Knuth [28] and then considered in [9], [10], [16], [25], [29]; see also the surveys [11], [30]. The *quadratic congruential generator* and the *inverse congruential generator* have been studied in [6] and [17], see also the recent paper [43] for a more general problem

On the other hand, our results are substantially weaker than those known for the linear and nonlinear congruential generators. In some sense, the problem we solve can be considered as a special case of the problem of finding small solutions of multivariate polynomial congruences. For polynomial congruences in one variable such an algorithm has been given by Coppersmith [12], see also [13], [24], [26]. However in the general case only heuristic results are known. That's the approach of [33].

The remainder of the paper is structured as follows: we start with a short outline of some basic facts about lattices and the abelian group associated to an elliptic curve in Section II. In Section III we present the algorithm. Finally, we conclude with Section IV which shortly discuss the results of numerical tests of our approach.

II. Background

A. Integer Lattices

Here we collect several well-known facts about lattices which form the background to our algorithms.

We review several results and definitions of concepts related to lattices which can be found in [20]. For more details and more recent references, we also recommend consulting [1], [25], [34].

Let $\{\vec{b}_1, \ldots, \vec{b}_s\}$ be a set of linearly independent vectors in $\mathbb{R}^r$. The set

$$\mathcal{L} = \{c_1\vec{b}_1 + \cdots + c_s\vec{b}_s \ : \ c_1, \ldots, c_s \in \mathbb{Z}\}$$

is called (s-dimensional) *lattice* with *basis* $\{\vec{b}_1, \ldots, \vec{b}_s\}$. If $s = r$, the lattice $\mathcal{L}$ is of *full rank*.

To each lattice $\mathcal{L}$ one can naturally associate its *volume*:

$$vol(\mathcal{L}) = \left(\det\left(\langle\vec{b}_i, \vec{b}_j\rangle\right)_{i,j=1}^{s}\right)^{1/2},$$

where $\langle\vec{a}, \vec{b}\rangle$ denotes the inner product. This definition does not depend on the choice of the basis $\{\vec{b}_1, \ldots, \vec{b}_s\}$.

For a vector $\vec{u}$, let $\|\vec{u}\|$ denote its *Euclidean norm*. The first Minkowski theorem, see Theorem 5.3.6 in [20], gives the upper bound

$$\min\left\{\|\vec{z}\|: \ \vec{z} \in \mathcal{L} \setminus \{\vec{0}\}\right\} \le s^{1/2} vol(\mathcal{L})^{1/s}$$

on the shortest nonzero vector in any s-dimensional lattice $\mathcal{L}$ in terms of its volume.

The Minkowski bound (II-A) motivates a natural question, the *Shortest Vector Problem (SVP)*: how to find a shortest nonzero vector in a lattice. Unfortunately, there are several indications that this problem is **NP**-hard when the dimension grows. This study has suggested several definitions of a *reduced* basis $\{\vec{b}_1, \ldots, \vec{b}_s\}$ for a lattice, trying to obtain a shortest vector by the first basis element $\vec{b}_1$. The celebrated *LLL algorithm* of Lenstra, Lenstra and Lovász [32] provides a concept of *reduced* basis and an approximate solution, enough in many practice applications.

Another related question is the *Closest Vector Problem (CVP)*: given a lattice $\mathcal{L} \subseteq \mathbb{R}^r$ and a shift vector $\vec{t} \in \mathbb{R}^r$, the goal consists on finding a vector in the set $\vec{t}+\mathcal{L}$ with minimum norm. This problem is usually expressed in an equivalent way: finding a vector in $\mathcal{L}$ closest to the target vector $-\vec{t}$. It is well known that CVP is **NP**-hard when the dimension grows.

However, both computational problems SVP and CVP are known to be solvable in deterministic polynomial time provided that the dimension of $\mathcal{L}$ is fixed (see [27], [3], [8], for example). The lattices in this paper are of fixed (and low) dimension.

In fact, lattices in this paper consist of integer solutions $\vec{x} = (x_0, \ldots, x_{s-1}) \in \ \mathbb{Z}^s$ of a system of congruences

$$\sum_{i=0}^{s-1} a_{ij}x_i \equiv 0 \bmod q_j, \qquad j = 1, \ldots, m,$$

modulo some positive integers $q_1, \ldots, q_m$. Typically (although not always) the volume of such a lattice is the product $Q = q_1 \cdots q_m$. Moreover, all the aforementioned algorithms, when applied to such a lattice, become polynomial in $\log Q$. If $\{\vec{b}_1, \ldots, \vec{b}_s\}$ is a basis of the above lattice, by the Hadamard inequality we have:

$$\prod_{i=1}^{s} \|\vec{b}_i\| \ge vol(\mathcal{L}). \tag{3}$$

B. The Group Associated to an Elliptic Curve

In this subsection we recall some basic facts about the group law on elliptic curves.

Let E be an elliptic curve defined over $\mathbb{F}_p$ given by the affine Weierstrass equation (1).

The operation $\oplus$ acts over the points $P = (x_P, y_P)$ and $Q = (x_Q, y_Q)$ of $E(\mathbb{F}_p)$ with $P, Q \neq \mathcal{O}$ as follows:

$$P \oplus Q = R = (x_R, y_R)$$

- If $x_P \neq x_Q$, then

$$x_R = m^2 - x_P - x_Q, \qquad y_R = m(x_P - x_R) - y_P, \qquad \text{where} \quad m = \frac{y_Q - y_P}{x_Q - x_P}. \tag{4}$$

- If $x_P = x_Q$ but $y_P \neq y_Q$, then $P \oplus Q = \mathcal{O}$.
- If $P = Q$ and $y_P \neq 0$, then

$$x_R = m^2 - 2x_P, \qquad y_R = m(x_P - x_R) - y_P, \qquad \text{where} \quad m = \frac{3x_P^2 + a}{2y_P}. \tag{5}$$

- If $P = Q$ and $y_P = 0$, then $P \oplus Q = \mathcal{O}$.

Our context is a pseudorandom number generator which outputs affine points in an elliptic curve. One obtains recursively them by operating a fixed composer G to the previous value. So, almost always, the above equations in the first case (4) will determine the process.

The set $E(\mathbb{F}_p)$ of $\mathbb{F}_p$-rational points forms an abelian group satisfying the Hasse-Weil inequality:

$$|\#\left(E(\mathbb{F}_p) - p - 1\right)| \le 2\sqrt{p}.$$

It is well known that the group $E(\mathbb{F}_p)$ is of the form

$$E(\mathbb{F}_p) \cong \mathbb{Z}/L\mathbb{Z} \times \mathbb{Z}/M\mathbb{Z},$$

where the integers L and M are uniquely determined with M divides L , see [2], [7], [42] for these and other general properties of elliptic curves.

III. The algorithm

Assume that a, b are unknown, but the prime p is given to us. We show that when we are given Δ-approximations $\bar{G}$ to the composer $G = (x_G, y_G) \in E(\mathbb{F}_p)$ and W_n, W_{n+1} to (respectively) two consecutive affine values U_n, U_{n+1} produced by the EC-LCG; we show that the value $U_n = (x_n, y_n)$ can be recovered from this information if the approximations $W_j, j = 0, 1$ and $\bar{G}$ are sufficiently good. To simplify the notation, we assume that $n = 0$ from now on.

We write $\bar{G} = (\gamma_x, \gamma_y)$ and $W_j = (\alpha_j, \beta_j)$, $U_j = (x_j, y_j)$, for $j = 0, 1$; so there exist integers h_x, h_y and e_j, f_j for j=0, 1 with:

$$\begin{aligned} x_G = \gamma_x + h_x, \quad y_G = \gamma_y + h_y, \quad & \& \quad |h_x|, |h_y| \le \Delta \\ x_j = \alpha_j + e_j, \quad & y_j = \beta_j + f_j \\ |e_j|, |f_j| \le \Delta, \quad & j = 0, 1. \end{aligned} \tag{6}$$

So, our input of this algorithm consists of $\alpha_0, \beta_0, \alpha_1, \beta_1, \gamma_x, \gamma_y \in \mathbb{F}_p$ and the positive integer Δ.

We suppose that U_0 and U_1 are not G or $-G$. Then, clearing denominators in equations (4), we can translate

$$U_1 = U_0 \oplus G \tag{7}$$

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

$$L_1 = L_1(x_G, y_G, x_0, y_0, x_1) \equiv 0 \bmod p$$

and

$$L_2 = L_2(x_G, y_G, x_0, y_0, x_1, y_1) \equiv 0 \bmod p,$$

where

$$\begin{aligned} L_1 = {} & {x_G}^3 + x_1{x_G}^2 - x_0{x_G}^2 - 2\,x_1x_Gx_0 - x_G{x_0}^2 \\ & + {x_0}^3 + 2y_Gy_0 + x_1{x_0}^2 - {y_G}^2 - {y_0}^2, \\ L_2 = {} & y_1x_G - y_1x_0 - y_Gx_0 + y_Gx_1 - y_0x_1 + y_0x_G. \end{aligned} \tag{8}$$

Using the equalities $x_G = \gamma_x + h_x$, $y_G = \gamma_y + h_y$ for one hand, and $x_j = \alpha_j + e_j$ and $y_j = \beta_j + f_j$ ($j = 0, 1$) for the other hand, equations (8) become for L_1 :

$(3\alpha_0^2+2\alpha_0\alpha_1-2\alpha_0\gamma_x-2\alpha_1\gamma_x-\gamma_x^2)e_0+(\alpha_0^2-2\alpha_0\gamma_x+\gamma_x^2)e_1+(-2\beta_0+2c_y)f_0+(-\alpha_0^2-2\alpha_0\alpha_1-2\alpha_0\gamma_x+2\alpha_1\gamma_x+3\gamma_x^2)h_x+(2\beta_0-2c_y)h_y+(3\alpha_0+\alpha_1-\gamma_x)e_0^2+(2\alpha_0-2\gamma_x)e_0e_1+$
$+(-2\alpha_0-2\alpha_1-2\gamma_x)e_0h_x+(-2\alpha_0+2\gamma_x)e_1h_x+$
$+(-\alpha_0+\alpha_1+3\gamma_x)h_x^2+$
$e_0^3+e_0^2e_1-e_0^2h_x-2e_0e_1h_x-e_0h_x^2+e_1h_x^2+h_x^3-f_0^2+2f_0h_y-h_y^2$
$=$

$$-\alpha_0^3-\alpha_0^2\alpha_1+\alpha_0^2\gamma_x+2\alpha_0\alpha_1\gamma_x+\alpha_0\gamma_x^2-\alpha_1\gamma_x^2-\gamma_x^3+\beta_0^2-2\beta_0c_y+c_y^2$$

and for L_2 :

$(-\beta_1-\gamma_y)e_0+(-\beta_0+\gamma_y)e_1+(-\alpha_1+\gamma_x)f_0+(-\alpha_0+\gamma_x)f_1+(\beta_0+\beta_1)h_x+(-\alpha_0+\alpha_1)h_y+$
$-e_1f_0-e_0f_1+f_0h_x+f_1h_x-e_0h_y+e_1h_y$
$=$

$$\alpha_1\beta_0+\alpha_0\beta_1-\beta_0\gamma_x-\beta_1\gamma_x+\alpha_0\gamma_y-\alpha_1\gamma_y$$

Now, we linearize this polynomial system. Writing

$$\begin{aligned}
& A_0 \equiv -\alpha_0^3-\alpha_0^2\alpha_1+\alpha_0^2\gamma_x+2\alpha_0\alpha_1\gamma_x+\alpha_0\gamma_x^2-\alpha_1\gamma_x^2-\gamma_x^3+\beta_0^2 \\
& -2\beta_0c_y+c_y^2 \bmod p, \\
& A_1 \equiv 3\alpha_0^2+2\alpha_0\alpha_1-2\alpha_0\gamma_x-2\alpha_1\gamma_x-\gamma_x^2 \bmod p, \\
& A_2 \equiv -2\beta_0+2c_y \bmod p,\ \ A_3 \equiv -2\beta_0+2c_y \bmod p,\ \ A_4 \equiv 0 \bmod p, \\
& A_5 \equiv -\alpha_0^2-2\alpha_0\alpha_1-2\alpha_0\gamma_x+2\alpha_1\gamma_x+3\gamma_x^2 \bmod p, \\
& A_6 \equiv 2\beta_0-2c_y \bmod p, \quad A_7 \equiv 3\alpha_0+\alpha_1-\gamma_x \bmod p, \\
& A_8 \equiv 2\alpha_0-2\gamma_x \bmod p, \quad A_9 \equiv -2\alpha_0-2\alpha_1-2\gamma_x \bmod p, \\
& A_{10} \equiv -2\alpha_0+2\gamma_x \bmod p, \quad A_{11} \equiv -\alpha_0+\alpha_1+3\gamma_x \bmod p, \\
& A_{12} \equiv 0 \bmod p, \quad A_{13} \equiv 1 \bmod p, \\
& B_0 \equiv \alpha_1\beta_0+\alpha_0\beta_1-\beta_0\gamma_x-\beta_1\gamma_x+\alpha_0\gamma_y-\alpha_1\gamma_y \bmod p, \\
& B_1 \equiv -\beta_1-\gamma_y \bmod p, \quad B_2 \equiv -\beta_0+\gamma_y \bmod p, \\
& B_3 \equiv -\alpha_0+\gamma_x \bmod p, \quad B_4 \equiv -\alpha_0+\gamma_x \bmod p, \\
& B_5 \equiv \beta_0+\beta_1 \bmod p, \quad B_6 \equiv -\alpha_0+\alpha_1 \bmod p, \quad B_7 \equiv 0 \bmod p \\
& B_8 \equiv 0 \bmod p, \quad B_9 \equiv 0 \bmod p, \quad B_{10} \equiv 0 \bmod p, \quad B_{11} \equiv 0 \bmod p, \\
& B_{12} \equiv 1 \bmod p,
\end{aligned} \tag{9}$$

we obtain that vector

$\vec{E} =$
$(\Delta^2e_0, \Delta^2e_1, \Delta^2f_0, \Delta^2f_1, \Delta^2h_x, \Delta^2h_y, \Delta e_0^2, \Delta e_0e_1, \Delta e_0h_x, \Delta e_1h_x, \Delta h_x^2, \Delta(-e_1f_0-e_0f_1+f_0h_x+f_1h_x-e_0h_y+e_1h_y), e_0^3+e_0^2e_1-e_0^2h_x-2e_0e_1h_x-e_0h_x^2+e_1h_x^2+h_x^3-f_0^2+2f_0h_y-h_y^2)$
$=$
$(\Delta^2E_1, \Delta^2E_2, \Delta^2E_3, \Delta^2E_4, \Delta^2E_5, \Delta^2E_6, \Delta E_7, \Delta E_8, \Delta E_9, \Delta E_{10}, \Delta E_{11}, \Delta E_{12}, E_{13})$

is a solution to the following linear system of congruences:

$$\begin{aligned}
& \sum_{i=1}^{6} A_iX_i + \sum_{i=7}^{12} \Delta A_iX_i + \Delta^2A_{13}X_{13} \equiv \Delta^2A_0 \bmod p, \\
& \sum_{i=1}^{6} B_iX_i + \sum_{i=7}^{12} \Delta B_iX_i + \Delta^2B_{13}X_{13} \equiv \Delta^2B_0 \bmod p, \\
& X_1 \equiv X_2 \equiv X_3 \equiv X_4 \equiv X_5 \equiv X_6 \equiv 0 \bmod \Delta^2, \\
& X_7 \equiv X_8 \equiv X_9 \equiv X_{10} \equiv X_{11} \equiv X_{12} \equiv 0 \bmod \Delta.
\end{aligned} \tag{10}$$

Moreover, $\vec{E}$ is a relatively short vector.

Let $\mathcal{L}$ be the lattice consisting of integer solutions $\vec{X} = (X_1, X_2, \ldots, X_{13}) \in \mathbb{Z}^{13}$ of the system of congruences:

$$\begin{aligned}
& \sum_{i=1}^{6} A_iX_i + \sum_{i=7}^{12} \Delta A_iX_i + \Delta^2A_{13}X_{13} \equiv 0 \bmod p, \\
& \sum_{i=1}^{6} B_iX_i + \sum_{i=7}^{12} \Delta B_iX_i + \Delta^2B_{13}X_{13} \equiv 0 \bmod p, \\
& X_1 \equiv X_2 \equiv X_3 \equiv X_4 \equiv X_5 \equiv X_6 \equiv 0 \bmod \Delta^2, \\
& X_7 \equiv X_8 \equiv X_9 \equiv X_{10} \equiv X_{11} \equiv X_{12} \equiv 0 \bmod \Delta.
\end{aligned} \tag{11}$$

We compute a solution $\vec{T}$ of the system of congruences (10), using linear diophantine equations methods. Applying an algorithm solving the CVP for the shift vector $\vec{T}$ and the lattice $\mathcal{L}$, we obtain a vector

$\vec{F} =$
$(\Delta^2F_1, \Delta^2F_2, \Delta^2F_3, \Delta^2F_4, \Delta^2F_5, \Delta^2F_6, \Delta F_7, \Delta F_8, \Delta F_9, \Delta F_{10}, \Delta F_{11}, \Delta F_{12}, F_{13})$

We have $\vec{F} = \vec{v} + \vec{T}$ (where $\vec{v}$ is the lattice vector returned by the CVP algorithm) is the vector of minimal norm satisfying equations (10), hence $\vec{F}$ must have norm at most equal to the norm of the solution $\vec{E}$. Note that we can compute $\vec{F}$ in polynomial time from the information we are given. We might hope that $\vec{E}$ and $\vec{F}$ are the same.

The so-called "Gaussian heuristic" suggests that and s-dimensional lattice $\mathcal{L}$ with volume $vol(\mathcal{L})$ is unlikely to have a nonzero vector which is substantially shorter than $vol(\mathcal{L})^{1/s}$. Moreover, if it is known that such a very short vector does exist, then up to a scalar factor it is likely to be the only vector with this property. On the other hand, the volume of the 12-dimensional lattice $\mathcal{L}$ defined by equations (11) is;

$$vol(\mathcal{L}) = p^2\Delta^{12}\Delta^6 = p^2\Delta^{18}$$

Then, vector $\vec{E}$ is likely to be the one founded whenever

$$\Delta^3 < p^{2/12}\Delta^{18/12},$$

this is,

$$\Delta < p^{1/9}$$

IV. Computational results

We have proposed an algorithm to recover a sequence of pseudorandom numbers produced by EC-LCG. The input required include approximations to some pseudorandom values. The quality of those approximations is the measure used to characterise when the algorithm output the expected sequence.

We have performed some numerical tests with a SageMath implementation. First, we generate an elliptic curve over a prime finite field of a desired size by choosing randomly in $\mathbb{F}_p$ parameters a, b for the equation $Y^2 = X^3 + aX + b$. Then, we generated randomly some composers G and some points in the curve by choosing the first coordinate and trying to solve the equation. For several composers and points, and EC-LCG is simulated, and approximations to some composer and some consecutive values are given as input to our algorithm. We have selected several primes of several sizes and note the obtained success threshold.

References

[1] M. Ajtai, R. Kumar and D. Sivakumar, "A sieve algorithm for the shortest lattice vector problem", *Proc. 33rd ACM Symp. on Theory of Comput. (STOC 2001)*, Association for Computing Machinery, 2001, 601–610.

[2] R. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange and K. Nguyen, "Elliptic and hyperelliptic curve cryptography: Theory and practice", CRC Press, 2005.

[3] L. Babai, "On Lovasz Lattice Reduction and the Nearest Lattice Point Problem", *Combinatorica*, **6**, 1–6, 1986.

[4] P. Beelen and J. Doumen, "Pseudorandom sequences from elliptic curves", *Finite Fields with Applications to Coding Theory*, Cryptography and Related Areas, Springer-Verlag, Berlin, 2002, 37-52.

[5] L. Beshaj, J. Gutierrez, T. Shaska, "Weighted greatest common divisors and weighted heights", *Journal Number Theory*, https://doi.org/10.1016/j.jnt.2019.12.012, 2020.

[6] S. R. Blackburn, D. Gomez-Perez, J. Gutierrez and I. E. Shparlinski, "Predicting nonlinear pseudorandom number generators", *Math. Computation*, **74** (2005), 1471–1494.

[7] I. Blake, G. Seroussi and N. Smart, "Elliptic curves in cryptography", *London Math. Soc., Lecture Note Series,* **265**, Cambridge Univ. Press, 1999.

[8] J. Bloemer, A. May, "A tool kit for Finding small roots of Bivariate Polynomial over the Integers", *Advances in Cryptology-Crypto 2003*, LNCS **2729**, Springer Verlag, 2003, 27–43.

[9] J. Boyar, "Inferring sequences produced by pseudo-random number generators", *J. ACM*, **36** (1989), 129–141

[10] J. Boyar, "Inferring sequences produces by a linear congruential generator missing low–order bits", *J. Cryptology* **1** (1989) 177–184.

[11] E. F. Brickell and A. M. Odlyzko, "Cryptanalysis: A survey of recent results", *Contemp. Cryptology*, IEEE Press, NY, 1992, 501–540.

[12] D. Coppersmith: "Small solutions to polynomial equations and low exponent RSA vulnerabilities". *J. Cryptology* **10 (4)**, 1997, 233–260.

[13] D. Coppersmith: "Finding a Small Root of a Bivariate Integer Equations; Factoring with High Bits Known". U. Maurer (Ed), *Proc. EUROCRYPT-96*, LNCS **1070**, Springer-Verlag, Berlin 1996, 155–156.

[14] E. El Mahassni and I. E. Shparlinski, " On the uniformity of distribution of congruential generators over elliptic curves", *Proc. Intern. Conf. on Sequences and their Applications*, Bergen 2001, Springer-Verlag, London, 2002, 257–264.

[15] G. Frey and T. Shaska: "Curves, Jacobians, and cryptography", *Algebraic Curves and Their Applications. AMS Contemporary Mathematics*, pp. 279-344, 2019.

[16] A. M. Frieze, J. Håstad, R. Kannan, J. C. Lagarias and A. Shamir, "Reconstructing truncated integer variables satisfying linear congruences", *SIAM J. Comp.*, **17** (1988), 262–280.

[17] D. Gomez-Perez, J. Gutierrez and A. Ibeas, "Attacking the Pollard Generator", *IEEE. Trans. Information Theory*, **vol. 52** (2006), no. 12, 5518–5523.

[18] G. Gong, T. A. Berson and D. A. Stinson, "Elliptic curve pseudorandom sequence generators", *Lect. Notes in Comp. Sci.*, Springer-Verlag, Berlin, 1758 (2000), 34–49.

[19] G. Gong and C. C. Y. Lam, "Linear recursive sequences over elliptic curves", *Proc. Intern. Conf. on Sequences and their Applications*, Bergen 2001, Springer-Verlag, London, 2002, 182-196.

[20] M. Grötschel, L. Lovász and A. Schrijver, *Geometric algorithms and combinatorial optimization*, Springer-Verlag, Berlin, 1993.

[21] J. Gutierrez and A. Ibeas: "Inferring sequences produced by a linear congruential generator on elliptic curves missing high-order bits". *Designs, Codes Cryptography*, 45(2): 199-212 (2007).

[22] S. Hallgren, "Linear congruential generators over elliptic curves", *Preprint CS-94-143, Dept. of Comp. Sci.*, Cornegie Mellon Univ., 1994, 1-10.

[23] F. Hess and I. E. Shparlinski, "On the linear complexity and multidimensional distribution of congruential generators over elliptic curves", *Designs, Codes and Cryptography*, 35 (2005), 111–117.

[24] N. A. Howgrave-Graham, "Finding small roots of univariate modular equations revisited", *Proc. 6th IMA Intern. Conf on Cryptography and Coding*, Lect. Notes in Comp. Sci., vol. 1355, Springer-Verlag, Berlin, 1997, 131–142.

[25] A. Joux and J. Stern, "Lattice reduction: A toolbox for the cryptanalyst", *J. Cryptology*, **11** (1998), 161–185.

[26] E. Jochemz and A. May, "A Strategy for Finding Roots of Multivariate Polynomials with New Applications in Attacking RSA Variants" *n Advances in Cryptology (Asiacrypt 2006)*, Lecture Notes in Computer Science, Springer-Verlag, 2006.

[27] R. Kannan, "Minkowski's convex body theorem and integer programming", *Math. Oper. Res.*, **12** (1987), 415–440.

[28] D. E. Knuth, "Deciphering a linear congruential encryption", *IEEE Trans. Inf. Theory* **31** (1985), 49–52.

[29] H. Krawczyk, "How to predict congruential generators", *J. Algorithms*, **13** (1992), 527–545.

[30] J. C. Lagarias, "Pseudorandom number generators in cryptography and number theory", *Proc. Symp. in Appl. Math.*, Amer. Math. Soc., Providence, RI, **42** (1990), 115–143.

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

[32] A. K. Lenstra, H. W. Lenstra and L. Lovász, "Factoring polynomials with rational coefficients", *Mathematische Annalen*, **261** (1982), 515–534.

[33] T. Mefenza, "Inferring Sequences Produced by a Linear Congruential Generator on Elliptic Curves Using Coppersmith's Methods". *COCOON 2016. LNCS 9797* 293-304, 2016.

[34] D. Micciancio and S. Goldwasser, "Complexity of lattice problems", Kluwer Acad. Publ., 2002.

[35] M. Naor and O. Reingold, "Number theoretic constructions of efficient pseudo-random functions", *Proc 38th IEEE Symp. on Found. of Comp. Sci.,* IEEE, 1997, 458–467.

[36] H. Niederreiter, "Design and analysis of nonlinear pseudorandom number generators", in G.I. Schueller and P. D. Spanos (Eds) *Monte Carlo Simulation*, A.A. Balkema Publishers, Rotterdam, 2001, 3–9.

[37] I. E. Shparlinski, "Cryptographic applications of analytic number theory ", *Birkhauser,* 2003.

[38] I. E. Shparlinski, "Orders of points on elliptic curves", *Affine Algebraic Geometry*, Amer. Math. Soc., 2005, 245–252.

[39] I. E. Shparlinski, "Pseudorandom Points on Elliptic Curves over Finite Fields ", *Recent trends in Cryptography* , Contemporary Mathematics, v.477, Amer.Math. Soc., 121-141. 2009.

[40] I. E. Shparlinski, "On the Naor-Reingold pseudo-random function from elliptic curves", *Appl.Algebra in Engin., Commun. and Computing,* 11 (2000), 27–34.

[41] I. E. Shparlinski and J. H. Silverman, "On the linear complexity of the Naor-Reingold pseudorandom function from elliptic curves", *Designs, Codes and Cryptography,* 24 (2001), 279–289.

[42] J. H. Silverman, "The arithmetic of elliptic curves", Springer-Verlag, Berlin, 1995.

[43] H. Sun, X Zhu, Q. Zheng: "Predicting truncated multiple recursive generators with unknown parameters". *Des. Codes Cryptography*. https://doi.org/10.1007/s10623-020-00729-8, 2020.

[44] A. Winterhof, "Recent Results on Recursive Nonlinear Pseudorandom Number Generators", *(Invited Paper). SETA* 2010: 113-124, 2010.