Predicting truncated multiple matrix congruential generators with unknown parameters

Changcun Wang, Zhaopeng Dai
School of Mathematics and Statistics, QingDao University
ccwang710@hotmail.com, dzpeng@amss.ac.cn

2024 年 10 月 9 日

Abstract

Multiple Matrix congruential generators is an important class of pseudorandom number generators. This paper studies the predictability of a class of truncated multiple matrix congruential generators with unknown parameters. Given a few truncated digits of high-order bits or low-order bits output by a multiple matrix congruential generator, we give a method based on lattice reduction to recover the parameters and the initial state of the generator.

1 Introduction

Random numbers is at the heart of Monte Carle methods and simulation, while widely used in cryptography. In the cryptographic setting, the requirements for "randomness" are somewhat different from ordinary applications in simulation, such as predictability. A pseudorandom number generator(PRNG) is considered unpredictable if the known conditional probability of the next event, given the previous history events or any other information, is no different from the known unconditional probability. The predictability of PRNGs is a crucial aspect of evaluating the security of PRNGs for cryptographic applications and numerous studies have explored the predictability of PRNGs, yielding diverse results.

As one class of PRNGs, the linear congruential generator(LCG) proposed by D. H. Lehmer in [15]. is based on the recurrence

$$a_{i+1} \equiv ba_i + c (\mathrm{mod}\, m), \text{with}\, 0 \leq a_i < m.$$

In the case that the parameters b, c, m are unknown, Plumstead [19] showed that the fixed LCG is predictability if all bits of the sequence are output. Knuth [11] considered the problems arising when only truncated high-order bits of the sequence generated by an LCG

are known. In the case that the modulus $m = 2^k$ is known and the parameters b, c are unknown, Knuth presented an attack to reconstruct the parameters and the seed of the LCG. Subsequently, the study is further extended, due to the work of Frieze [6, 7], Boyar [2], Stern[10, 21].

LCGs gained popularity due to their simplicity, efficiency, and well-established theoretical characteristics. However, contemporary standards do not recommend the use of LCGs. This is primarily attributed to their limited periods, which are bounded by the modulus m. In order to improve the period of the sequences generated by LCGs, the multiple recursive generator(MRG) is proposed and defined as

$$a_{i+n} \equiv c_{n-1}a_{i+n-1} + \cdots + c_1 a_{i+1} + c_0 a_i (\mathrm{mod}\, m), \text{with } i \geq 0,$$

where $c_0, \ldots, c_{n-1}$ are not all zero. Deng [4] showed that by appropriately choosing the parameters, MRGs can be as fast as the classical LCGs, while their periods are much longer and their empirical performances are better. In particular, ZUC [20], a stream cipher included in the 4G mobile standard, uses an MRG of order 16 over $\mathbb{Z}_{2^{31}-1}$. The characteristic polynomial of the MRG is

$$f(x) = x^{16} - (2^{15}x^{15} + 2^{17}x^{13} + 2^{21}x^{10} + 2^{20}x^4 + 2^8 + 1)$$

which is a primitive polynomial over $\mathbb{Z}_{2^{31}-1}$. Recently, insights into the predictability of MRGs have also been explored. In the case that all parameters of a MRG are known, Yang et al. [23] introduced a method based on Coppersmith's approach to recover the initial state using truncated sequences. In situations where all MRG parameters are unknown, Sun et al. [22] extended Stern's algorithm to predict truncated MRGs. Their approach involves constructing appropriate lattices to sequentially recover the modulus m, followed by the coefficients $c_0, \ldots, c_{n-1}$, and ultimately retrieving the initial state $a_0, \ldots, a_{n-1}$. Yu et al. [24] proposed an alternative method for parameter and initial state recovery, emphasizing its superior efficiency and reduced data requirements compared to Sun et al.'s approach. Notably, Yu et al. [25] successfully addressed MRGs with constant c type. There are several specific results about MRGs that can be referred to [3, 9, 12, 13, 14, 26].

In the realm of linear PRNGs, another notable type is Pseudorandom Vector Generators (PRVGs). The significance of pseudorandom vectors has been growing, particularly due to the increasing emphasis on parallelization in scientific computing. Random vectors find typical applications in parallelized probabilistic algorithms, parallel Monte Carlo and simulation methods, as well as in multivariate statistics. Matrix congruential generators (MCGs) constitute a crucial class of PRVGs. Given a vector $(u_{0,0}, \ldots, u_{0,n-1}) \in \mathbb{Z}_m^n$ and a matrix $A \in \mathbb{Z}_m^{n\times n}$ as input, an MCG outputs a sequence $u = (u_{i,0}, \ldots, u_{i,n-1})_{i\geq 0}$ which satisfies the recurrence

$$(u_{i+1,0}, \ldots, u_{i+1,n-1}) \equiv (u_{i,0}, \ldots, u_{i,n-1})A \,(\mathrm{mod}\, m), \text{with } i \geq 0.$$

While previous work in [5, 8, 18] have predominantly explored characteristics like the period of MCGs, few results have been available regarding the predictability of MCGs. This

paper focuses on predicting truncated k-order ($k \geq 2$) MMCGs, where both the matrix $A_i, (i = 1, \ldots, s-1)$ and modulus m are unknown. The approach involves constructing an appropriate lattice to recover the modulus m, followed by the retrieval of the matrix $A_i, (i = 1, \ldots, s-1)$ and the initial state. Similar to Stern's algorithm, the success of the method in this paper relies on two lattice properties that are challenging to prove mathematically.

The other parts of this article are organized as follows. Section 2 introduces the basic knowledge of lattice. Section 3 proposes our method to predict truncated MMCGs with known high-order bits. Section 4 proposes our method to predict truncated MMCGs with known low-order bits. Section 5 provides an experimental example. section 6 proposes issues for further study.

2 Preliminary

In this section,we briefly review the basic knowledge of lattice reduction.
All lattice take row vectors as a lattice basic in this paper.

Definition 1. Let $n \geq l \geq 1$ and let $v_1, v_2, \ldots, v_l$ be a set of l linearly independent vectors in $\mathbb{R}^n$. The l-dimensional lattice spanned by $v_1, v_2, \cdots, v_l$ in n-dimensional Euclidean space is defined to be

$$L = \mathbb{Z}v_1 + \mathbb{Z}v_2 + \cdots + \mathbb{Z}v_l = \{\sum_{i=1}^{l} k_i v_i \mid k_1, k_2, \ldots, k_l \in \mathbb{Z}\}.$$

For $i = 1, 2, \ldots, l$, let $v_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,n})$ and form the $l \times n$ matrix $V = (v_{i,j})$. Let V^T be the transpose of V. Then the determinant of lattice L is defined by

$$\det(\mathrm{L}) = \sqrt{\det(VV^T)}.$$

Definition 2. Let L be a lattice. The ith minimum of L denoted by $\lambda_i(L)$ is defined to be the radius of the smallest zero-centered ball containing at least i linearly independent lattice vectors. In particular, $\lambda_1(L)$ is the norm of the shortest non-zero vector in L.

Lemma 3. [17] Let L be an n-dimensional lattice. If the Gaussian heuristic holds for L, then

$$\lambda_1(L) \approx \sqrt{\frac{n}{2\pi e}} \det(L)^{\frac{1}{n}}.$$

Denote by $\|\mathbf{u}\|$ the Euclidean norm of vector $\mathbf{u}$. The following two types of difficult problems are commonly concerned in lattice.

(1) The shortest vector problem(SVP): Given a lattice $L \subset \mathbb{R}^n$, find a non-zero vector $\mathbf{u} \in L$, such that $\|\mathbf{u}\| \leq \|\mathbf{v}\|$ for any non-zero vector $\mathbf{v} \in L$.

(2)The closest vector problem(CVP): Given a lattice $L \subset \mathbb{R}^n$ and a vector $\mathbf{w} \in \mathbb{R}^n$, find a vector $\mathbf{u} \in L$, such that $\|\mathbf{u}\text{-}\mathbf{w}\| \leq \|\mathbf{v}\text{-}\mathbf{w}\|$ for any vector $\mathbf{v} \in L$.

Lemma 4. [16] Let L be an n-dimensional lattice. Assume δ is the reduction parameter of the LLL algorithm and $v_1, v_2, \ldots, v_n$ is a δ-reduced basis of L output by the LLL algorithm. Let $\lambda_i(L)$ be the ith minimum of L. Then

(1) $\|v_i\| \leq \rho^{(n-1)/2}\lambda_i(L), 1 \leq i \leq n$,
(2) $\|v_1\| \leq \rho^{(n-1)/4}\det(L)^{1/n}$,

where $\rho = \frac{4}{4\delta-1}$.

Remark 1 The value of the reduction parameter is usually set to $\delta = \frac{3}{4}$, at the time $\rho = 2$.

3 Main results

In this paper, the k-order ($k \geq 2$) of Multiple-MCG is denoted as

$$u_{i+s} \equiv u_i A_0 + u_{i+1} A_1 + \cdots + u_{i+s-1} A_{s-1} \pmod{m}, \tag{1}$$

where $u_i = (u_{i,0}, \ldots, u_{i,k-1}), (i \geq 0)$, $s \geq 1$, $m \geq 2$, and

$$A_i = \begin{pmatrix} a_{i,11} & a_{i,12} & \cdots & a_{i,1k} \\ a_{i,21} & a_{i,22} & \cdots & a_{i,2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i,k1} & a_{i,k2} & \cdots & a_{i,kk} \end{pmatrix} \in \mathbb{Z}_m^{k\times k}.$$

To guarantee that the sequence $\underline{u}$ is purely periodic, we assume that $\gcd(\det A_i, m) = 1$.

Define $l = \lceil \log m \rceil$ as the number of bits in $m-1$. Assuming that $y_{i,j}$ encompasses a fraction α of the high-order bits of $u_{i,j}$ and $z_{i,j}$ comprises the remaining low-order bits, we can express this as follows:

$$u_{i,j} = 2^{\beta l} y_{i,j} + z_{i,j}, \tag{2}$$

where $\beta = 1-\alpha$. We study the following predictability problem: Given the first N truncated digits

$$\{(y_{0,0}, y_{0,1}, \ldots, y_{0,k-1}), \ldots, (y_{N-1,0}, y_{N-1,1}, \ldots, y_{N-1,k-1})\}.$$

We want to predict the rest of the sequence accurately. Obviously, if we can recover the modulus m, the matrix A_i, $(i = 0, \ldots, s-1)$, and the initial state $(u_0, u_1, \ldots, u_{s-1})$, then we can predict the rest of the sequence. As a result, to solve the predictability problem we just need to solve the following problem.

Problem Let $y_{i,j}$ be a proportion α of the high-order bits of $u_{i,j}$. Given the first N truncated digits $((y_{0,0}, y_{0,1}, \ldots, y_{0,k-1}), \ldots, (y_{N-1,0}, y_{N-1,1}, \ldots, y_{N-1,k-1}))$, the problem is to recover the modulus m, the matrix A_i, $(i = 0, \ldots, s-1)$, and the initial state $(u_0, u_1, \ldots, u_{s-1})$.

For $i \in \{0, \ldots, s-1\}$, $j \in \{1, \ldots, k\}$, let $\alpha_{i,j} = (a_{i,1j}, \ldots, a_{i,kj})^T$, then $A_i = (\alpha_{i,1}, \ldots, \alpha_{i,k})$. By (1), we can get

$$\begin{cases} u_{i+s,0} \equiv u_i\alpha_{0,1} + u_{i+1}\alpha_{1,1} + \cdots + u_{i+s-1}\alpha_{s-1,1} \pmod m \\ u_{i+s,1} \equiv u_i\alpha_{0,2} + u_{i+1}\alpha_{1,2} + \cdots + u_{i+s-1}\alpha_{s-1,2} \pmod m \\ \vdots \\ u_{i+s,k-1} \equiv u_i\alpha_{0,k} + u_{i+1}\alpha_{1,k} + \cdots + u_{i+s-1}\alpha_{s-1,k} \pmod m \end{cases}. \tag{3}$$

Then by the recurrence relation of the sequence $\underline{u}$, for $j \geq s$, we have

$$(u_{j,0}, u_{j,1}, \ldots, u_{j,k-1}) \equiv (u_0, u_1, \ldots, u_{s-1})T_j \pmod m, \tag{4}$$

where T_j is a $sk \times k$ matrix. Let

$$T_j = \begin{pmatrix} t_{j,0} & t_{j,sk} & \cdots & t_{j,sk(k-1)} \\ t_{j,1} & t_{j,sk+1} & \cdots & t_{j,sk(k-1)+1} \\ \vdots & \vdots & \vdots & \vdots \\ t_{j,sk-1} & t_{j,2sk-1} & \cdots & t_{j,sk(k-1)+sk-1} \end{pmatrix}.$$

In particular, when $j = s$, then

$$T_s = \begin{pmatrix} A_0 \\ A_1 \\ \vdots \\ A_{s-1} \end{pmatrix}.$$

For $i = 0, 1, \ldots, k-1$, then we have

$$u_{j,i} \equiv \sum_{v=0}^{k-1} t_{j,isk+v}u_{0,v} + \sum_{v=k}^{2k-1} t_{j,isk+v}u_{1,v-k} + \cdots + \sum_{v=(s-1)k}^{sk-1} t_{j,isk+v}u_{s-1,v-(s-1)k} \pmod m. \tag{5}$$

3.1 Searching linear dependence relations

Lemma 5. [10] Assume $Y_0, Y_1, \cdots, Y_{r-1}$ is a family of vectors with integer coordinates in the t-dimensional space, with $t < r$. Let M denote an upper bound for the absolute values of all coordinates of the various Y_is. There exist integers $\zeta_0, \zeta_1, \cdots, \zeta_{r-1}$ such that

$$\sum_{i=0}^{r-1} \zeta_i Y_i = 0,$$

where $\max|\ \zeta_i\ | \leq B$, and B is given by

$$logB = t\frac{logM + logr + 1}{r - t}.$$

Let $r = kh$, $h \in \mathbb{Z}$. For $(i = 0, 1, \ldots, h-1)$. Choose positive r, t satisfying $r > t > sk$ and construct vectors $Y_i \in \mathbb{Z}^t$,

$$\begin{aligned} Y_{ki} &= (y_{i,0}, y_{i+1,0}, \ldots, y_{i+t-1,0}), \\ Y_{ki+1} &= (y_{i,1}, y_{i+1,1}, \ldots, y_{i+t-1,1}), \\ Y_{ki+2} &= (y_{i,2}, y_{i+1,2}, \ldots, y_{i+t-1,2}), \\ &\vdots \\ Y_{ki+k-1} &= (y_{i,k-1}, y_{i+1,k-1}, \ldots, y_{i+t-1,k-1}). \end{aligned} \tag{6}$$

According to the Lemma 5, there exist integer coefficients $\eta_0, \ldots, \eta_{r-1}$, such that

$$\sum_{i=0}^{r-1} \eta_i Y_i = 0, \tag{7}$$

where $\mid \eta_i \mid \leq B$, and B satisfying

$$logB = t\frac{log(2^{\alpha l}) + logr + 1}{r-t} = t\frac{\alpha l + logr + 1}{r-t}. \tag{8}$$

The next step of the process is to discuss in which case the equation

$$U \triangleq \sum_{i=0}^{r-1} \eta_i A_i = 0 \tag{9}$$

holds, where

$$\begin{aligned} A_{ki} &= (u_{i,0}, u_{i+1,0}, \ldots, u_{i+t-1,0}), \\ A_{ki+1} &= (u_{i,1}, u_{i+1,1}, \ldots, u_{i+t-1,1}), \\ A_{ki+2} &= (u_{i,2}, u_{i+1,2}, \ldots, u_{i+t-1,2}), \\ &\vdots \\ A_{ki+k-1} &= (u_{i,k-1}, u_{i+1,k-1}, \ldots, u_{i+t-1,k-1}). \end{aligned}$$

As a preparation for recovering the unknown parameters, we still need to find a set of linear relations for $\underline{u}^N = ((u_{0,0}, u_{0,1}, \ldots, u_{0,k-1}), \ldots, (u_{N-1,0}, u_{N-1,1}, \ldots, u_{N-1,k-1}))$. However, we do not need to find the vectors $\eta = (\eta_0, \eta_1, \ldots, \eta_{r-1})$ that validate both (7) and (9). Actually, validating (9) is our ultimate goal while (7) can be neglected. Observe that if (9) holds, then

$$0 = \sum_{i=0}^{r-1} \eta_i A_i = 2^{\beta l} \sum_{i=0}^{r-1} \eta_i Y_i + \sum_{i=0}^{r-1} \eta_i Z_i,$$

where

$$\begin{aligned} Z_{ki} &= (z_{i,0}, z_{i+1,0}, \ldots, z_{i+t-1,0}), \\ Z_{ki+1} &= (z_{i,1}, z_{i+1,1}, \ldots, z_{i+t-1,1}), \\ Z_{ki+2} &= (z_{i,2}, z_{i+1,2}, \ldots, z_{i+t-1,2}), \\ &\vdots \\ Z_{ki+k-1} &= (z_{i,k-1}, z_{i+1,k-1}, \ldots, z_{i+t-1,k-1}), \end{aligned}$$

and so

$$\sum_{i=0}^{r-1} \eta_i Y_i = -2^{-\beta l} \sum_{i=0}^{r-1} \eta_i Z_i.$$

Since $|z_{i,j}| < 2^{\beta l}$, it follows that

$$\|\sum_{i=0}^{r-1}\eta_i Y_i\| \le \sum_{i=0}^{r-1}\|-2^{-\beta l}\eta_i Z_i\| \le \sqrt{t}\sum_{i=0}^{r-1}|\eta_i|,$$

This means that $\sum_{i=0}^{r-1}\eta_i Y_i$ is a short vector, which inspires us to search a short vector of the lattice rather than a vector that validates (7).

Construct the following lattice

$$L' = \begin{pmatrix} Y_0 & 1 & 0 & \cdots & 0 \\ Y_1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Y_{r-1} & 0 & 0 & \cdots & 1 \end{pmatrix},$$

and obviously, the $(r+t)$-dimensional vector

$$\mathbb{W} = (w_0, \ldots, w_{t-1}, w_t, \ldots, w_{r+t-1}) = (\sum_{i=0}^{r-1}\eta_i Y_i, \eta_0, \eta_1, \ldots, \eta_{r-1}) \in L'.$$

If η is a vector that validates (9), then $\sum_{i=0}^{r-1}\eta_i Y_i$ is a short vector, and so is $\mathbb{W}$. Therefore, the desire vectors $\eta = (\eta_0, \eta_1, \ldots, \eta_{r-1})$ that validate (9) can be obtained by applying LLL to the lattice L'.

Remark 2 If the value of r and t are chosen correctly, a single call to lattice reduction algorithm can output multiple sets of $\eta_0, \eta_1, \ldots, \eta_{r-1}$ that satisfy (9).

3.2 Constructing congruence equations

We can appropriately determine the values of r and t through the aforementioned process. we can get

$$\begin{cases} \sum_{i=0}^{k-1}\eta_i u_{0,i} + \sum_{i=k}^{2k-1}\eta_i u_{1,i-k} + \cdots + \sum_{i=(s-1)k}^{sk-1}\eta_i u_{s-1,i-(s-1)k} + \eta_{sk}u_{s,0} + \cdots + \eta_{r-1}u_{h-1,k-1} = 0 \\ \sum_{i=0}^{k-1}\eta_i u_{1,i} + \sum_{i=k}^{2k-1}\eta_i u_{2,i-k} + \cdots + \sum_{i=(s-1)k}^{sk-1}\eta_i u_{s,i-(s-1)k} + \eta_{sk}u_{s+1,0} + \cdots + \eta_{r-1}u_{h,k-1} = 0 \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ \sum_{i=0}^{k-1}\eta_i u_{t-1,i} + \sum_{i=k}^{2k-1}\eta_i u_{t,i-k} + \cdots + \sum_{i=(s-1)k}^{sk-1}\eta_i u_{t+s-2,i-(s-1)k} + \eta_{sk}u_{t+s-1,0} + \cdots + \eta_{r-1}u_{t+h-2,k-1} = 0 \end{cases} \tag{10}$$

By (10), we can get

$$\begin{pmatrix} u_{0,0} & \cdots & u_{0,k-1} & \cdots & u_{s-1,0} & \cdots & u_{s-1,k-1} \\ u_{1,0} & \cdots & u_{1,k-1} & \cdots & u_{s,0} & \cdots & u_{s,k-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_{t-1,0} & \cdots & u_{t-1,k-1} & \cdots & u_{t+s-2,0} & \cdots & u_{t+s-2,k-1} \end{pmatrix} \begin{pmatrix} t_0 \\ t_1 \\ \vdots \\ t_{sk-1} \end{pmatrix} \equiv \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \bmod m, \tag{11}$$

where

$$t_i = \eta_i + \sum_{j=sk}^{2sk-1} \eta_j t_{s,k(j-sk)+i} + \sum_{j=2sk}^{3sk-1} \eta_j t_{s+1,k(j-2sk)+i} + \cdots + \sum_{j=(h-1)k}^{hk-1} \eta_j t_{h-1,k(j-(h-1)k)+i}, \tag{12}$$

for $0 \le i \le sk-1$.

If (11) has only the trivial solution over $\mathbb{Z}_m^{sk}$, then we can get

$$\begin{cases} t_0 \equiv 0 \mod m \\ t_1 \equiv 0 \mod m \\ \quad \vdots \\ t_{sk-1} \equiv 0 \mod m \end{cases}. \tag{13}$$

So，the modulus m and the matrix A can be recovered by (13) which is to be seen in Sects. 3.3 and 3.4. Therefore, it is necessary to discuss the condition that (11) has only trivial solution.

Denote by T the coefficient matrix in (11). Obviously, if select a sk-order submatrix from T randomly, and the probability that the determinant of the submatrix is coprime to m is $6/\pi^2$. So the probability that (11) has only the trivial solution is about $C_t^{sk} \times 6/\pi^2$, then the probability is almost equal to 1.

3.3 Recovering the modulus m

In this section we will prove that for any $i \in \{0, 1, \ldots, sk-1\}$, the modulus m can be recovered by $t_i \equiv 0 \bmod m$ in (13).

From (13), we know that $t_i \equiv 0 \bmod m$, i.e.,

$$t_i = \eta_i + \sum_{j=sk}^{2sk-1} \eta_j t_{s,k(j-sk)+i} + \sum_{j=2sk}^{3sk-1} \eta_j t_{s+1,k(j-2sk)+i} + \cdots + \sum_{j=(h-1)k}^{hk-1} \eta_j t_{h-1,k(j-(h-1)k)+i} \equiv 0,$$

then there exists an integer u_i making the following equation holds.

$$\eta_i = u_i m - \Big(\sum_{j=sk}^{2sk-1} \eta_j t_{s,k(j-sk)+i} + \sum_{j=2sk}^{3sk-1} \eta_j t_{s+1,k(j-2sk)+i} + \cdots + \sum_{j=(h-1)k}^{hk-1} \eta_j t_{h-1,k(j-(h-1)k)+i} \Big).$$

Constructing the following lattice $L(t_i)$ whose dimension is $r - sk + 1$

$$L(t_i) = \begin{pmatrix} m & 0 & 0 & \cdots & 0 \\ -t_{s,i} & 1 & 0 & \cdots & 0 \\ -t_{s,k+i} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -t_{h-1,k(k-1)+i} & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

The determinant of $L(t_i)$ is m. Denoted by ω_j the j row vector of $L(t_i)$ for $0 \leq j \leq r - sk$. Then we have

$$\eta(i) = u_i\omega_0 + \sum_{j=sk}^{r-1} \eta_j\omega_{j-sk+1} \in L(t_i),$$

where $\eta(i) = (\eta_i, \eta_{sk}, \eta_{sk+1}, \ldots, \eta_{r-1})$.

From Sections 3.1 and 3.2, it is evident that with a finite truncated sequence of $h+t-1$ vectors represented as $y_l(h+t-1) = ((y_{l,0}, \ldots, y_{l,k-1}), \ldots, (y_{l+h+t-2,0}, \ldots, y_{l+h+t-2,k-1}))$ for some $l \geq 0$, it becomes possible to identify at least one $\eta(i)$ such that $t_i \equiv 0 \bmod m$. If we are acquainted with d distinct finite truncated sequences (where some y_l may be shared by distinct finite truncated sequences), the conclusion can be drawn that we can obtain at least d such vectors

$$\eta(i)^{(j)} = (\eta_i^{(j)}, \eta_{sk}^{(j)}, \eta_{sk+1}^{(j)}, \ldots, \eta_{r-1}^{(j)}), \quad 0 \leq j \leq d-1$$

satisfying $t_i \equiv 0 \bmod m$.

Let

$$M(t_i) = \begin{pmatrix} {\eta_i}^{(0)} & {\eta_{sk}}^{(0)} & {\eta_{sk+1}}^{(0)} & \cdots & {\eta_{r-1}}^{(0)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ {\eta_i}^{(d-1)} & {\eta_{sk}}^{(d-1)} & {\eta_{sk+1}}^{(d-1)} & \cdots & {\eta_{r-1}}^{(d-1)} \end{pmatrix}.$$

The row vectors of $M(t_i)$ are not necessary linearly independent. So we reduce $M(t_i)$ through LLL algorithm and let $L(t_i)^*$ be the lattice generated by the output reduced basis. Since

$$\eta(i)^{(j)} = (\eta_i^{(j)}, \eta_{sk}^{(j)}, \eta_{sk+1}^{(j)}, \ldots, \eta_{r-1}^{(j)}) \in L(t_i)$$

for any $j \in \{0, \ldots, d-1\}$. Obviously, $L(t_i)^*$ is a sublattice of $L(t_i)$.

Similarly to case of LCGs, the lattice $L(t_i)^*$ also has the following two properties which were found by Stern [10, 21]:

(1) As d grows, the dimension of $L(t_i)^*$ very quickly increases to full rank $r - sk + 1$;

(2) Once the dimension of $L(t_i)^*$ reaches full rank, the determinant of $L(t_i)^*$ is a multiple of m, and very quickly decreases to m.

3.4 Recovering the matrix $A_0, \ldots, A_{s-1}$

In this section we will show that the matrix $A_i, i \in \{0, \ldots, s-1\}$ can be recovered by (13).

Building on the discussion in Section 3.3, it is apparent that $L(t_i)^*$ forms a sublattice of $L(t_i)$. Upon achieving full rank, the lattice $L(t_i)^*$ ensures that its determinant becomes a multiple of m. As new vectors are added, the determinant of $L(t_i)^*$ rapidly decreases, eventually reaching m. When the determinant of $L(t_i)^*$ reaches m, we have

$$L(t_i)^* = L(t_i).$$

Let

$$L(t_i)^* = \begin{pmatrix} \tau_{0,0} & \tau_{0,1} & \tau_{0,2} & \cdots & \tau_{0,r-sk} \\ \tau_{1,0} & \tau_{1,1} & \tau_{1,2} & \cdots & \tau_{1,r-sk} \\ \tau_{2,0} & \tau_{2,1} & \tau_{2,2} & \cdots & \tau_{2,r-sk} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tau_{r-sk,0} & \tau_{r-sk,1} & \tau_{r-sk,2} & \cdots & \tau_{r-sk,r-sk} \end{pmatrix}.$$

Choose a large constant $K > m2^{(r-sk)/2}$. Multiply the jth column vector of $L(t_i)^*$ by K for $sk+1 \le j \le r-sk$ and denoted by G the new lattice, that is

$$G = \begin{pmatrix} \tau_{0,0} & \tau_{0,1} & \cdots & \tau_{0,sk} & K\tau_{0,sk+1} & \cdots & K\tau_{0,r-sk} \\ \tau_{1,0} & \tau_{1,1} & \cdots & \tau_{1,sk} & K\tau_{1,sk+1} & \cdots & K\tau_{1,r-sk} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \tau_{sk,0} & \tau_{sk,1} & \cdots & \tau_{sk,sk} & K\tau_{sk,sk+1} & \cdots & K\tau_{sk,r-sk} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \tau_{r-sk,0} & \tau_{r-sk,1} & \cdots & \tau_{r-sk,k} & K\tau_{r-sk,sk+1} & \cdots & K\tau_{r-sk,r-sk} \end{pmatrix}.$$

Reduce G through LLL algorithm and denoted by $H = (h_{u,v})_{0 \le u,v \le r-sk}$ the reduced matrix. Let H_j be the jth row vector of H for $0 \le j \le r-sk$. Then we have the following conclusion.

Theorem 6. If $L(t_i)^* = L(t_i)$, then the lattice generated by $H_0, \ldots, H_{sk}$ is equal to the lattice generated by $(m, 0, 0, \ldots, 0)$, $(-t_{s,i}, 1, 0, \ldots, 0)$, $\ldots$, $(-t_{s,k(sk-1)+i}, 0, \ldots, 1, 0, \ldots, 0)$.

Through Theorem 6, we can get the matrix A_i, for $i \in \{0, \ldots, s-1\}$.

3.5 Recovering the initial state $(u_0, u_1, \ldots, u_{s-1})$

We have known the modulus m and $A_i, i \in \{0, \ldots, s-1\}$. Our objective now is to recover the initial state $(u_0, u_1, \ldots, u_{s-1})$. Given that $(y_0, y_1, \ldots, y_{s-1})$ is known, by (2), recovering $(u_0, u_1, \ldots, u_{s-1})$ is essentially equivalent to recovering $(z_0, z_1, \ldots, z_{s-1})$. Yang [23] has successfully addressed this challenge by transforming it into the task of finding small integer solutions to systems of linear congruences. He then applied the method proposed by Frieze [6] to solve this problem. Let's briefly outline this method.

Suppose the first $d(d > sk)$ consecutive truncated digits are given, we need to recover $(z_0, z_1, \ldots, z_{s-1})$. By (2) and (5), for $i \in \{0, \ldots, k-1\}$, $s \le j \le d-1$, we can get

$$(E - z_{j,i}) \equiv 2^{\beta l}(y_{j,i} - F) \bmod m, \tag{14}$$

where

$$E = \sum_{v=0}^{k-1} t_{j,isk+v} z_{0,v} + \sum_{v=k}^{2k-1} t_{j,isk+v} z_{1,v-k} + \cdots + \sum_{v=(s-1)k}^{sk-1} t_{j,isk+v} z_{s-1,v-(s-1)k},$$

$$F = \sum_{v=0}^{k-1} t_{j,isk+v} y_{0,v} + \sum_{v=k}^{2k-1} t_{j,isk+v} y_{1,v-k} + \cdots + \sum_{v=(s-1)k}^{sk-1} t_{j,isk+v} y_{s-1,v-(s-1)k}.$$

It can be seen that to recover the unknown $(z_0, z_1, \ldots, z_{s-1})$, it suffices to solve (14). Let

$$L(m, d+s(k-1)) = \begin{pmatrix} m & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & m & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & m & 0 & 0 & \cdots & 0 \\ t_{s,isk} & t_{s,isk+1} & \cdots & t_{s,isk+sk-1} & -1 & 0 & \cdots & 0 \\ t_{s+1,isk} & t_{s+1,isk+1} & \cdots & t_{s+1,isk+sk-1} & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{d-1,isk} & t_{d-1,isk+1} & \cdots & t_{d-1,isk+sk-1} & 0 & 0 & \cdots & -1 \end{pmatrix}.$$

Let $z = (z_0, \ldots, z_{s-1}, z_{s,i}, \ldots, z_{d-1,i})$ and let $H = d + s(k-1)$. Denoted by λ_H the Hth minimum of the lattice $L(m, d+s(k-1))$. The solution of (14) has been solved in [6].

Lemma 7. [6] The system of modular Eqs.(14) has at most one solution $z \in \mathbb{Z}^H$ satisfying

$$\|z\| \leq m\lambda_H^{-1} 2^{-(H-1)/2-1}.$$

Furthermore, there is a polynomial time algorithm that either finds z or proves that no such z exists.

Since $|z_{i,j}| < 2^{\beta l}$, then we have $\|z\| < \sqrt{H}2^{\beta l}$, by Lemma 7, if

$$\sqrt{H}2^{\beta l} \leq m\lambda_H^{-1} 2^{-(H-1)/2-1}$$

then

$$\alpha k \geq \frac{logH}{2} + log\lambda_H + \frac{H+1}{2},$$

then we can recover the initial state $(u_0, \ldots, u_{s-1})$ by solving (14).

4 The algorithm with known low-order bits

Problem Let $z_{i,j}$ be the low-order bits of $u_{i,j}$. Given the first N truncated digits

$$((z_{0,0}, z_{0,1}, \ldots, z_{0,k-1}), \ldots, (z_{N-1,0}, z_{N-1,1}, \ldots, z_{N-1,k-1})),$$

the problem is to recover the modulus m, the matrix A_i, $(i = 0, \ldots, s-1)$, and the initial state $(u_0, u_1, \ldots, u_{s-1})$.

Let $K = \lceil \frac{1}{2^{\beta l}} (\sqrt{\frac{2\pi e r}{r+t}} B)^{\frac{r+t}{t}} \rceil$. Construct the following lattice

$$L_2 = \begin{pmatrix} K2^{\beta l} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & K2^{\beta l} & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & K2^{\beta l} & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ Kz_{0,0} & Kz_{1,0} & \cdots & Kz_{t-1,0} & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ Kz_{0,1} & Kz_{1,1} & \cdots & Kz_{t-1,1} & 0 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ Kz_{0,k-1} & Kz_{1,k-1} & \cdots & Kz_{t-1,k-1} & 0 & 0 & \cdots & 1 & 0 & 0 & 0 \\ Kz_{1,0} & Kz_{2,0} & \cdots & Kz_{t,0} & 0 & 0 & \cdots & 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ Kz_{h-1,k-1} & Kz_{h,k-1} & \cdots & Kz_{h+t-2,k-1} & 0 & 0 & \cdots & 0 & 0 & 0 & 1 \end{pmatrix}$$

and obviously, the $(r+t)$-dimensional vector

$$\mathbb{W} = (w_0, \ldots, w_{t-1}, w_t, \ldots, w_{r+t-1}) = (0, \ldots, 0, \eta_0, \eta_1, \ldots, \eta_{r-1}) \in L_2.$$

According to Lemma 3, the shortest vector $\lambda_1(L_2)$ of L_2 is $\lambda_1(L_2) \approx \sqrt{\frac{r+t}{2\pi e}} (2^{\beta l} K)^{\frac{t}{r+t}}$. Due to

$$\|\eta\| = \sqrt{\sum_{i=0}^{r-1} \eta_i^2} \leq \sqrt{r} B \leq \sqrt{\frac{r+t}{2\pi e}} (2^{\beta l} K)^{\frac{t}{r+t}}.$$

Therefore, the desire vectors $\eta = (\eta_0, \eta_1, \ldots, \eta_{r-1})$ that validate (9) can be obtained by applying LLL to the lattice L_2. Then we can get (10) and (11), the modulus m and A_i, $i = 1, \ldots, s-1$ can be recovered by (13), more details in Sects 3.3 and 3.4.

4.1 Recovering the initial state $(u_0, u_1, \ldots, u_{s-1})$

We have known the modulus m and $A_i, i \in \{0, \ldots, s-1\}$. Our objective now is to recover the initial state $(u_0, u_1, \ldots, u_{s-1})$. Given that $(z_0, z_1, \ldots, z_{s-1})$ is known, by (2), recovering $(u_0, u_1, \ldots, u_{s-1})$ is essentially equivalent to recovering $(y_0, y_1, \ldots, y_{s-1})$.

Suppose the first $d (d > sk)$ consecutive truncated digits are given, we need to recover $(y_0, y_1, \ldots, y_{s-1})$. Let $\widehat{y}_i = y_i - 2^{\alpha l - 1}$, then

$$-2^{\alpha l - 1} \leq \widehat{y}_i \leq 2^{\alpha l - 1}.$$

Let $\gcd(2^{\beta l}, m) = 2^{\rho l}$, then $0 \leq \rho \leq \beta$, by (14), we have

$$2^{\rho l} \mid z_{j,i} - E.$$

By (14), we can get

$$2^{(\beta-\rho)l} y_{j,i} \equiv 2^{(\beta-\rho)l} F - \frac{(z_{j,i} - E)}{2^{\rho l}} \bmod \frac{m}{2^{\rho l}} \tag{15}$$

if $\rho < \beta$, where $\gcd(2^{(\beta-\rho)l}, \frac{m}{2^{\rho l}}) = 1$. Multiply (15) by $[2^{-(\beta-\rho)l}]_{\mathrm{mod}\,\frac{m}{2^{\rho l}}}$, we can get

$$y_{j,i} \equiv F - b_j \bmod \frac{m}{2^{\rho l}},\ s \le j \le d-1,$$

where $b_j = [2^{-(\beta-\rho)l} \cdot \frac{(z_{j,i}-E)}{2^{\rho l}}]_{\mathrm{mod}\,\frac{m}{2^{\rho l}}}$.

we can get

$$y_{j,i} \equiv F - b_j \bmod \frac{m}{2^{\rho l}},\ s \le j \le d-1,$$

if $\rho = \beta$, where $b_j = [\frac{(z_{j,i}-E)}{2^{\rho l}}]_{\mathrm{mod}\,\frac{m}{2^{\rho l}}}$.

By Kannan's embedding technique, we construct the following lattice

$$L_3 = \begin{pmatrix} 2^{\alpha l-1} & 2^{\alpha l-1} & \cdots & 2^{\alpha l-1} & b_s + 2^{\alpha l-1} & b_{s+1} + 2^{\alpha l-1} & \cdots & b_{d-1} + 2^{\alpha l-1} \\ 0 & 1 & \cdots & 0 & t_{s,isk} & t_{s+1,isk} & \cdots & t_{d-1,isk} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & t_{s,isk+sk-1} & t_{s+1,isk+sk-1} & \cdots & t_{d-1,isk+sk-1} \\ 0 & 0 & \cdots & 0 & \frac{m}{2^{\rho l}} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \frac{m}{2^{\rho l}} & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & \frac{m}{2^{\rho l}} \end{pmatrix}.$$

The target vector $\mathbf{v} = (-2^{\alpha l-1}, \widehat{y}_{0,0}, \ldots, \widehat{y}_{s-1,k-1}, \widehat{y}_{s,i}, \ldots, \widehat{y}_{d-1,i}) \in L_3$, and

$$\|\mathbf{v}\| \le 2^{\alpha l-1}\sqrt{d + s(k-1) + 1}.$$

Obviously, the upper bound for the target vector is a conservative estimation of its length. Inspired by [1], heuristically, $y_{0,0}, \ldots, y_{0,k-1}, y_{1,0}, \ldots, y_{s-1,k-1}, y_{s,i}, \ldots, y_{d-1,i}$ are randomly uniformly distributed, so we use the expected norm of a uniformly distributed vector instead. The target vector $\mathbf{v}$ has the following expected squared norm

$$\begin{aligned} \mathbb{E}[\|\mathbf{v}\|^2] &= \mathbb{E}[\sum_{j=0}^{s-1}\sum_{t=0}^{k-1}(y_{j,t} - 2^{\alpha l-1})^2 + \sum_{j=s}^{d-1}(y_{j,i} - 2^{\alpha l-1}) + (2^{\alpha l-1})^2] \\ &= H \cdot \frac{1}{2^{\alpha l}} \cdot \sum_{i=0}^{2^{\alpha l}-1}(i - 2^{\alpha l-1})^2 + (2^{\alpha l-1})^2 \\ &= (\frac{H}{3} + 1) \cdot (2^{\alpha l-1})^2 + \frac{H}{6}. \end{aligned}$$

By Lemma 3,

$$\lambda_1(L_3) = \sqrt{\frac{H+1}{2\pi e}}(2^{\alpha l-1})^{\frac{1}{H+1}}(\frac{m}{2^{\rho l}})^{\frac{d-s}{H+1}}.$$

Heuristically, if

$$\sqrt{(\frac{H}{3} + 1) \cdot (2^{\alpha l-1})^2 + \frac{H}{6}} \le \sqrt{\frac{H+1}{2\pi e}}(2^{\alpha l-1})^{\frac{1}{H+1}}(\frac{m}{2^{\rho l}})^{\frac{d-s}{H+1}}, \tag{16}$$

then $\mathbf{v}$ could be recovered by reducing L_3 through the LLL algorithm. If this is achieved, we can recover $y_0, \ldots, y_{s-1}$ and then the initial state $u_0, \ldots, u_{s-1}$ by simple computation.

To give a lower bound in the number of truncated digits, here we simplify (16) by using proper scaling techniques. First we increase slightly the left-hand side of inequality (16) to

$$\sqrt{(\frac{H+3}{3})\cdot(2^{\alpha l-1})^2+\frac{H+3}{6}}=\sqrt{(H+3)\cdot[\frac{(2^{\alpha l-1})^2}{3}+\frac{1}{6}]},$$

then

$$\sqrt{(H+3)\cdot[\frac{(2^{\alpha l-1})^2}{3}+\frac{1}{6}]}\leq\sqrt{\frac{H+1}{2\pi e}}(2^{\alpha l-1})^{\frac{1}{H+1}}(\frac{m}{2^{\rho l}})^{\frac{d-s}{H+1}}.$$

Next, we square and take the logarithm that

$$\log(1+\frac{2}{H+1})+\log(\frac{(2^{\alpha l-1})^2}{3}+\frac{1}{6})+\log(2\pi e)\leq\frac{2(\alpha l-1)}{H+1}+\frac{2(d-s)\cdot\log m}{H+1}-\frac{2\rho l(d-s)}{H+1}.$$

Since

$$\log(1+\frac{2}{H+1})\leq\frac{2}{\ln 2}\cdot\frac{1}{H+1},$$

we replace $\log(1+\frac{2}{H+1})$ by $\frac{2}{\ln 2}\cdot\frac{1}{H+1}$, it follows that

$$d\geq\frac{\frac{2}{\ln 2}+(s(k-1)+1)Q-2(\alpha l-1-s\log m+\rho ls)}{2\log m-Q-2\rho l},$$

where $Q=\log(2\pi e)+\log(\frac{(2^{\alpha l-1})^2}{3}+\frac{1}{6})$.

5 Example

We have performed a lot of experiments to verify the correctness and effectiveness of our method. In our experiments, the parameters A_0, A_1 and the initial states u_0, u_1 are chosen randomly in $\mathbb{Z}_m$. All the experiments are performed on our personal computer(Windows 11, AMD R7-5800H, 3.20GHz). The lattice reduction algorithm used is the LLL algorithm in the Maple.

Now we define a k-MMCG and choose randomly its parameters and initial states as follow.

$$u_0=(326521,453100),u_1=(398002,488987),A_0=\begin{pmatrix}5&2\\3&4\end{pmatrix},A_1=\begin{pmatrix}3&5\\1&8\end{pmatrix},m=2^{19}-1.$$

As assumed in the previous text, we suppose that the modulus m, the matrix A_0, A_1, and the initial state u_0, u_1 are unknown, and we intend to recover the parameters and the initial state by means of 11 high-order bits of a part of the consecutive sequence, that is, $\alpha = 11/19$. The experimental process involves the following three steps.

Step 1 We select $r = 48$ and $t = 17$, and proceed to repeatedly apply the methodology outlined in Sect. 3.1 to obtain a set of such vectors $(\eta_0, \eta_1, \ldots, \eta_{47})$ that satisfies the condition $t_i \equiv 0 \bmod m$, for $i = 0, 1, 2, 3$.

Step 2 For $i \in \{0, 1, 2, 3\}$, we proceed with recovering the matrix A_0, A_1. The modulus m can be retrieved during the process of recovering the parameters $a_{i,11}, a_{i,12}, a_{i,21}, a_{i,22}$,

$i = \{0, 1\}$. In the recovery of $a_{i,11}, a_{i,12}, a_{i,21}, a_{i,22}$, we utilize the method detailed in Sects. 3.3 and 3.4. Here, we construct the matrix $M(t_i)$ by employing the vectors $\eta(i) = (\eta_i, \eta_2, \ldots, \eta_{47})$ obtained in step 1, and then reduce $M(t_i)$ using the LLL algorithm. With

Table 1: MMCG-Using only the first row

Coefficients	Reach full rank	Reach m	Coefficients	Reach full rank	Reach m
$a_{0,11}$	45	53	$a_{1,11}$	45	46
$a_{0,12}$	45	53	$a_{1,12}$	45	46
$a_{0,21}$	45	46	$a_{1,21}$	45	49
$a_{0,22}$	45	46	$a_{1,22}$	45	49

the inclusion of more row vectors, the dimension of the lattice $L(t_i)^*$ rapidly reaches full rank 45, and the determinant of $L(t_i)^*$ becomes a multiple of m, eventually decreased to m. When the determinant of $L(t_i)^*$ equals m, we confirm that $L(t_i)^* = L(t_i)$, facilitating the retrieval of the matrix A_0, A_1 through the procedure delineated in Sect. 3.4.

Only the first row vector output by the LLL algorithm in the step 1 is used here. The experimental results are shown in Table 1. The column named "Reach full rank" indicates the number of row vectors needed for the dimension of $L(t_i)^*$ to reaches full rank for the first time. The column named "Reach m" shows the quantity of row vectors wanted for the determinant of $L(t_i)^*$ to reaches m for the first time.

Step 3 When the modulus m and the matrix A_0, A_1 have been recovered, we use the method in Sect. 3.5 to recover the initial state u_0, u_1. In the experiments, we can use the first $d = 12$ consecutive truncated digits $\{(y_{0,0}, y_{0,1}), \ldots, (y_{11,0}, y_{11,1})\}$ to recover the initial state u_0, u_1.

Table 2: MMCG-Using only the first row(low-order)r=50,t=18

Coefficients	Reach full rank	Reach m	Coefficients	Reach full rank	Reach m
$a_{0,11}$	47	48	$a_{1,11}$	47	49
$a_{0,12}$	47	48	$a_{1,12}$	47	49
$a_{0,21}$	47	48	$a_{1,21}$	47	49
$a_{0,22}$	47	48	$a_{1,22}$	47	49

Step 1′ When the modulus m and the matrix A_0, A_1 have been recovered, we use the method in Sect. 4.1 to recover the initial state u_0, u_1. In the experiments, we can use the first $d = 12$ consecutive truncated digits $\{(y_{0,0}, y_{0,1}), \ldots, (y_{11,0}, y_{11,1})\}$ to recover the initial state u_0, u_1.

6 Conclusion

In this paper, we study the predictability of truncated k-MMCGs

$$u_{i+s} \equiv u_i A_0 + u_{i+1} A_1 + \cdots + u_{i+s-1} A_{s-1} \pmod{m}, \text{ with } i \geq 0$$

Assume that the modulus m, the matrix $A_0, \ldots, A_{s-1}$ and the initial state $u_0, \ldots, u_{s-1}$ are unknown. Given a few truncated digits of high-order bits or low-order bits, we give a method based on lattice reduction to predict the rest of the sequence. The limitations of our approach primarily revolve around one aspects: we currently lack a valid heuristic analysis for the choices of r and t. Open question include whether our method can succeed establish a valid heuristic analysis for the choices of r and t.

References

[1] Martin R Albrecht and Nadia Heninger. On bounded distance decoding with predicate: Breaking the "lattice barrier" for the hidden number problem. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 528–558. Springer, 2021.

[2] Joan Boyar. Inferring sequences produced by pseudo-random number generators. *Journal of the ACM (JACM)*, 36(1):129–141, 1989.

[3] Scott Contini and Igor E Shparlinski. On stern's attack against secret truncated linear congruential generators. In *Australasian Conference on Information Security and Privacy*, pages 52–60. Springer, 2005.

[4] Lih-Yuan Deng and Dennis KJ Lin. Random number generation for the new century. *The American Statistician*, 54(2):145–150, 2000.

[5] Jürgen Eichenauer-Herrmann, Holger Grothe, and Jürgen Lehn. On the period length of pseudorandom vector sequences generated by matrix generators. *Mathematics of Computation*, 52(185):145–148, 1989.

[6] Alan M. Frieze, Johan Hastad, Ravi Kannan, Jeffrey C. Lagarias, and Adi Shamir. Reconstructing truncated integer variables satisfying linear congruences. *Siam Journal on Computing*, 17(2):262–280, 1988.

[7] Alan M Frieze, Ravindran Kannan, and Jeffrey C Lagarias. Linear congruential generators do not produce random sequences. In *25th Annual Symposium onFoundations of Computer Science, 1984.*, pages 480–484. IEEE, 1984.

[8] Holger Grothe. Matrix generators for pseudo-random vector generation. *Statistische Hefte*, 28(1):233–238, 1987.

[9] Min-Qiang Huang. Analysis and cryptologic evaluation of primitive sequences over an integer residue ring. *Ph. D. dissertation, Graduate School, Univ. Sci. Technol. China, Beijing, China*, 1988.

[10] Antoine Joux and Jacques Stern. Lattice reduction: A toolbox for the cryptanalyst. *Journal of Cryptology*, 11:161–185, 1998.

[11] Donald E. Knuth. *The Art of Computer Programming: Seminumerical Algorithms*. Addison-Wesley, Reading, WA, 1969.

[12] Aleksei Sergeevich Kuzmin and Aleksandr Aleksandrovich Nechaev. Reconstruction of a linear recurrence of maximal period over a galois ring from its highest coordinate sequence. 2011.

[13] AS Kuzmin, GB Marshalko, and AA Nechaev. Reconstruction of a linear recurrence over a primary residue ring. *Memoires in Discrete Mathematics*, 12:155–194, 2009.

[14] AS Kuzmin and AA Nechaev. Linear recurring sequences over galois rings. *Algebra and Logic*, 34(2):87–100, 1995.

[15] Derrick H Lehmer. Mathematical models in large-scale computing units. *Ann. Comput. Lab.(Harvard University)*, 26:141–146, 1951.

[16] Arjen K Lenstra, Hendrik Willem Lenstra, and László Lovász. Factoring polynomials with rational coefficients. *Mathematische annalen*, 261:515–534, 1982.

[17] Phong Q Nguyen and Damien Stehlé. LLL on the average. In *Algorithmic Number Theory: 7th International Symposium, ANTS-VII, Berlin, Germany, July 23-28, 2006. Proceedings 7*, pages 238–256. Springer, 2006.

[18] Harald Niederreiter. A pseudorandom vector generator based on finite field arithmetic. *Math. Japonica*, 31(5):759–774, 1986.

[19] Joan B Plumstead. Inferring a sequence generated by a linear congruence. In *23rd Annual Symposium on Foundations of Computer Science (sfcs 1982)*, pages 153–159. IEEE, 1982.

[20] ETSI SAGE. Specification of the 3gpp confidentiality and integrity algorithms 128-eea3 & 128-eia3. document 4: Design and evaluation report, 2011.

[21] Jacques Stern. Secret linear congruential generators are not cryptographically secure. In *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*, pages 421–426. IEEE, 1987.

[22] Hong-Yu Sun, Xuan-Yong Zhu, and Qun-Xiong Zheng. Predicting truncated multiple recursive generators with unknown parameters. *Designs, Codes and Cryptography*, 88(6):1083–1102, 2020.

[23] JB Yang. Reconstructing truncated sequences derived from primitive sequences over inter residue rings. *PLA Information Engineering University, Zhengzhou*, 2017.

[24] Han-Bing Yu, Qun-Xiong Zheng, Yi-Jian Liu, Jing-Guo Bi, Yu-Fei Duan, Jing-Wen Xue, You Wu, Yue Cao, Rong Cheng, Lin Wang, et al. An improved method for predicting truncated multiple recursive generators with unknown parameters. *Designs, Codes and Cryptography*, 91(5):1713–1736, 2023.

[25] Hanbing YU and Qunxiong ZHENG. A lattice-based method for recovering the unknown parameters of truncated multiple recursive generators with constant. *Chinese Journal of Electronics*, 33:1–10, 2023.

[26] XY Zhu. Some results on injective mappings of primitive sequences modulo prime powers. *PLA Information Engineering University, Zhengzhou*, 2004.