Applying dissipative dynamical systems to pseudorandom number generation: Equidistribution property and statistical independence of bits at distances up to logarithm of mesh size

L. Yu. Barash

Landau Institute for Theoretical Physics - 142432 Chernogolovka, Russia

Introduction. – Pseudorandom number generation is an important component of any stochastic simulations such as molecular dynamics and Monte Carlo simulations [1]. The problem of design of reliable and fast generators is of great importance and attracts much attention [2].

There are numerous papers where chaos is considered as a requirement for good pseudorandomness. Many properties of chaotic dynamical systems are discussed in this respect: ergodicity, sensitivity to initial conditions, mixing property, local divergence of trajectories, deterministic dynamics and structural complexity. These properties resemble certain properties of pseudorandomness and are considered in the literature as desirable properties for pseudorandomness. Several pseudorandom number generators based on chaotic maps have been proposed in the literature [3,4]. However, the behavior of dynamical systems on a discrete lattice is studied much less than in continuous space and a number of corresponding important questions still remain open. In this work I show that applying dissipative dynamical systems to pseudorandom number generation can result in substantially preferable statistical behavior of the corresponding pseudorandom number sequences, compared to applying conservative dynamical systems.

The present approach extends the method of pseudorandom number generation of refs. [4,5], which is based on evolution of the ensemble of dynamical systems. Several generalizations are carried out. The connection between the statistical properties of a generator and the geometric properties of the corresponding map is uncovered. A new pseudorandom number generator is proposed. Using SSE2 technology, which is supported by all Intel and AMD processors fabricated later than 2003 [6,7], effective implementations are developed.

One of the most important properties characterizing the quality of pseudorandom sequences of numbers is the high-dimensional uniformity and the corresponding equidistribution property [8]. Unlike other essential characteristics of pseudorandom number generators such as the period length, which is studied in detail in relationship to nearly all known generators, there are not so many examples in which the high-dimensional equidistribution property was proved [8–11].

In this paper the proper choice of parameters is established, which results in the validity of the equidistribution property for the proposed generator. In particular, it is shown that the determinant of the transformation has to be an even integer in order for the property to hold. The equidistribution is established on length up to a characteristic length $\ell$: for $n \leq \ell$, each combination of successive $n$ bits taken from the RNG output occurs exactly the same number of times and has a corresponding probability $1/2^n$. 

Copyright © EPLA, 2011
The length $\ell$ turns out to depend linearly on $t$, where the mesh size $g$ (i.e., the modulus of the basic recurrence) is equal to $p \cdot 2^t$ and $p$ is an odd prime. In other words, for given $p$, one has $\ell \propto \log g$. Numerical results show that the equidistribution property still approximately holds with smaller values of $g$ (see tables 2, 3). Practically, the generators are compared with those of other modern generators (see tables 2, 3). The state of the generator consists of the $s$ realiztions. It is suggested in [4,5] to construct RNGs based on an ensemble of sequences generated by the multiple recursive method. The state of the generator consists of the values $x_i^{(n-1)}, x_i^{(n-2)} \in \{0, 1, \ldots, g-1\}, \ i = 0, 1, \ldots, s-1$. The transition function of the generator is defined by the recurrence relation

$$x_i^{(n)} = k x_i^{(n-1)} - q x_i^{(n-2)} \pmod g,$$  

where $i = 0, 1, \ldots, s-1$. The values $x_i^{(n)}, i = 0, 1, \ldots, s-1$ can be considered as $x$-coordinates of $s$ points $(x_i^{(n)}, y_i^{(n)})^T, \ i = 0, 1, \ldots, s-1$ of the $(g \times g)$-lattice on the two-dimensional torus, then each recurrence relation describes the dynamics of $x$-coordinate of a point on the two-dimensional torus:

$$x_i^{(n)} = M x_i^{(n-1)} \pmod g,$$  

where the matrix $M = (m_1, m_2)$ is a matrix with integer elements, $k = \text{Tr} M, q = \det M$ and $\text{Tr} M$ is a trace of matrix $M$. Indeed, it follows from (2) that $k x_i^{(n-1)} - q x_i^{(n-2)} = (m_1 + m_4) x_i^{(n-1)} - m_1 m_4 x_i^{(n-2)} = (x_i^{(n)} - m_2 y_i^{(n-1)}) + m_4 x_i^{(n-1)} - m_1 m_4 x_i^{(n-2)} + m_2 m_3 x_i^{(n-2)} = x_i^{(n)} - m_2 m_4 y_i^{(n-1)} - m_3 x_i^{(n-2)} + m_4 x_i^{(n-1)} - m_1 x_i^{(n-2)} = x_i^{(n)} - m_2 m_4 y_i^{(n-1)} - m_3 x_i^{(n-2)} + m_4 x_i^{(n-1)} - m_1 x_i^{(n-2)} = x_i^{(n)}$ (mod $g$). The basic recurrence (1) is therefore closely related to the so-called matrix generator of pseudorandom numbers studied in [2,13,15].

The output function is defined as follows:

$$a^{(n)} = \sum_{i=0}^{s-1} [2 x_i^{(n)} / g] \cdot 2^i,$$  

where $i = 0, 1, \ldots, s-1$, i.e. each bit of the output corresponds to its own recurrence, and $s = 32$ recurrences are calculated in parallel.

For $g = p \cdot 2^t$, where $p$ is a prime number, the characteristic polynomial $f(x) = x^p - k x + q$ is chosen to be primitive over $\mathbb{Z}_p$. Primitivity of the characteristic polynomial guarantees maximal possible period $2^p - 1$ of the output sequence for $g = p$. It is straightforward to prove that taking $g = p \cdot 2^t$ instead of $g = p$ does not reduce the value of the period.

There is an easy algorithm to calculate $x_i^{(n)}$ in (1) very quickly from $x_i^{(0)}$ and $x_i^{(1)}$ for any large $n$. Indeed, if $x_i^{(2n)} = k_n x_i^{(n)} - q_n x_i^{(0)}$ (mod $g$), then $x_i^{(4n)} = (k_n^2 - 2 q_n) x_i^{(2n)} - q_n x_i^{(0)}$ (mod $g$). As was mentioned already in [4], this helps to initialize the generator. To initialize all $s$ recurrences, the following initial conditions are used: $x_i^{(0)} = x_i^{(1)} = x_i^{(a+1)} = 0, 1, \ldots, s-1$. Here $A$ is a value of the order of $(p^2 - 1)/s$. The author has tested realizations with various values of $A$ of the order of $(p^2 - 1)/s$ and found in all cases that the specific choice of $A$ was not of importance for the properties studied in the next sections. Short cycles and, in particular, the cycle consisting of zeroes, are avoided if at least one of $x_i^{(0)}$ and $x_i^{(1)}$ is not divisible by $p$. As a result of the initialization, all $s$ initial points belong to the same orbit on the torus of the period $p^2 - 1$, while the minimal distance $A$ between the initial points along the orbit is chosen to be very large.

In addition to the realizations based on the output function (3) that takes a single bit from each linear recurrence, I have also constructed realizations based on a more general output function

$$a^{(n)} = \sum_{i=0}^{s-1} [2 v x_i^{(n)} / g] \cdot 2^{i v},$$  

where $v$ bits are taken from each recurrence and $i = 0, 1, \ldots, s-1$. For example, GM55.4-SSE realization.

| Generator    | $g$        | $k$ | $q$ | $v$ | Period  |
|--------------|------------|-----|-----|-----|---------|
| GM29.1-SSE   | 2$^{29} - 3$ | 4   | 2   | 1   | $2 \cdot 10^{12}$ |
| GM55.4-SSE   | (2$^{51} - 129$) | 256 | 176 | 4   | $5.1 \cdot 10^{30}$ |
| GQ39.1-SSE   | 2$^{29} - (2^9 - 3)$ | 8   | 48  | 1   | $2 \cdot 10^{12}$ |
| GQ37.3-SSE   | 2$^{29} - (2^9 - 3)$ | 8   | 48  | 3   | $2 \cdot 10^{12}$ |
| GQ38.4-SSE   | 2$^{29} - (2^9 - 3)$ | 8   | 48  | 4   | $2 \cdot 10^{12}$ |
calculates only $s = 8$ recurrent relations in parallel and takes $e = 4$ bits from each number. Pseudorandom 32-bit numbers can be generated if $se \geq 32$. The sequence of bits $\{2^e x_i^{(i)} / g\}$, where $i$ is fixed and $\{x_i^{(i)}\}$ is generated with relation (2) will be designated below as a stream of $v$-bit blocks generated with matrix $M$. The pairs $x_i^{(0)}, x_i^{(1)} \in \mathbb{Z}_g$ for the recurrence (1) and $x_i^{(0)}, y_i^{(0)} \in \mathbb{Z}_g$ for the recurrence (2) represent seeds for the streams of $v$-bit blocks generated with (1) and (2), respectively. Consider the set of admissible seeds containing all seeds such that at least one of the two values is not divisible by $p$. Selecting the seed at random from a uniform distribution over the set of admissible seeds determines the probability measure for output subsequences of a stream of $v$-bit blocks. Such probabilities are considered below in the next section.

The parameters for the particular constructed realizations of the generator are shown in table 1. The parameters are chosen in order for the characteristic polynomial $x^2 - kx + q$ to be primitive over $\mathbb{Z}_p$. In addition, as is shown below, value of $q$ must be divisible by $2^e$ in order for the equidistribution property to hold. Also the value of $(k + q)g$ should not exceed either $2^{32}$ or $2^{64}$ in order to effectively calculate four 32-bit recurrences or two 64-bit recurrences in parallel within SIMD arithmetic. In the particular case $t = 0$ and $e = 1$ the method reduces to that studied earlier in [4,5]. Program codes for the new generators and proper initializations are available in [16].

**Geometric properties and statistical properties.**

In [4] a connection is established between statistical properties, the results of a random walk test and geometric properties of the cat maps. Cat maps are simple chaotic dynamical systems that correspond to transformations (2) for $q = \text{det } M = 1$, i.e. hyperbolic automorphisms of the two-dimensional torus. In particular, it is proved in [4] that the probability of sequence 0000 of the first bits generated by a single cat map depends only on the trace $k$ of a matrix $M$ and for even $k$ is equal to $P = P_0 k^2 / (k^2 - 1)$, where $P_0 = 1/16$. If $k$ is odd, then all sequences of length 4 are equiprobable. The probability of sequence 0000 of length 5 is equal to $P = P_0 (1 + 1/(3k^2 - 6))$ for odd $k$, where $P_0 = 1/32$. The condition $P > P_0$ signifies that the 5-dimensional equidistribution never takes place for $q = 1$, i.e. for conservative hyperbolic automorphisms of the torus. In this work a more general case $q \neq 1$ involving dissipative dynamical systems is studied.

Figure 1 shows the regions on the torus obtained in [4] for the third points of sequences of length 5 for the matrix $(\frac{1}{2}, \frac{1}{2})$. The regions correspond to the sequences of length 5 of the first bits generated by the corresponding RNG. Each region is drawn with its own color.

![Fig. 1: (Color online) The regions on the torus obtained in [4] for the third points of sequences of length 5 for the matrix $(\frac{1}{2}, \frac{1}{2})$. Coordinates $x/y, y/g$ are used. These regions correspond to the sequences of length 5 of the first bits generated by the corresponding RNG. Each region is drawn with its own color.](image-url)

**Proposition 1.** If $M = \left(\begin{array}{ll} m_1 & m_2 \\ m_3 & m_4 \end{array} \right)$ is a matrix with integer values $m_1, m_2, m_3, m_4$, then $m_4 = \text{det } M$ and $q$ are divisible by $2^e$, where $q = \text{det } M$ and $q$ are divisible by $2^e$, the image of the lattice $g \times g$ with the transformation $M^j$ is invariant with respect to the shift $S$ for $j = 0, 1, \ldots, n$, and then all the sequences of length $n$ in a stream of $v$-bit blocks generated with matrix $M$ are equiprobable.

**Proof.** In this case the element $m_1^{(n)}$ of the matrix

$$M^n = \left(\begin{array}{ll} m_1^{(n)} & m_2^{(n)} \\ m_3^{(n)} & m_4^{(n)} \end{array} \right) \pmod{g}$$

satisfies the recurrence relation $m_1^{(n)} = km_1^{(n-1)} - qm_1^{(n-2)} \pmod{g}$. Hence $m_1^{(n)}$ is divisible by $2^e$ for any integer $n \geq 1$.

Since $m_1^{(n)}$ is divisible by $2^e$, one has $M^n S(x, y)^T = M^n (x + g/2^e \pmod{g}, y)^T = M^n (x, y)^T + (0, m_3^{(n)} g/2^e)^T$. Hence, the set of points $A$ such that $A \in X_i$ and $M^n(A) \in X_j$ passes with the shift $S$ into the set of points $A$ such that $A \in X_{i+1} \pmod{2^n}$ and $M^n(A) \in X_j$. Let us now prove by induction that all sequences of length $n$ are equiprobable. Obviously, if $g$ is divisible by $2^e$, sequences of length 1 are equiprobable: $P(0) = P(1) = \ldots = P(2^e - 1) = 1/2^e$. Assume that all sequences of length $n - 1$ are equiprobable. Let $\alpha_i = P(\{i, i + 1\} \pmod{2^n})$, $i = 0, 1, \ldots, 2^n - 1$ be probabilities of sequences of length $n$. Then $\alpha_i = \alpha_{i+1}$, $i = 0, 1, \ldots, 2^n - 2$ because the set of
points of the lattice \( g \times g \) such that \( A \in X_i \), \( M(A) \in X_{x_i} \), \( M^{n-1}(A) \in X_{x_{n-1}} \) passes with the shift \( S \) into the set of points \( A \) of the lattice \( g \times g \) such that \( A \in X_{x_i} \) \( (i) \mod 2^{m-1}) \), \( M(A) \in X_{x_i} \), \( M^{n-1}(A) \in X_{x_{n-1}} \). On the other hand, \( \sum_{i=0}^{n-1} a_i \) is the probability of sequence \( x_1 \ldots x_{n-1} \) of length \( n-1 \) and equals \( 1/2^{(n-1)} \). Therefore, \( a_i = 1/2^{m}, i = 0,1, \ldots ,2^{m} - 1 \), and all the sequences of length \( n \) are equiprobable. Proposition 1 is proved.

The condition that the image of the lattice \( g \times g \) with the transformation \( M^j \) is invariant with respect to the shift \( S \) for \( j = 0, 1, \ldots ,n \), is used in the above consideration and is necessary for the Proposition 1. For \( j = 0 \) the invariance means that \( g \) is divisible by \( 2^m \). If \( g \) and \( m_1^{(n)} \) are divisible by \( 2^m \), then the number of points \( A \) of the lattice \( g \times g \) such that \( A \in X_0 \) and \( M^{n}(A) \in X_0 \) is equal to the number of points \( A \) of the same lattice such that \( A \in X_1 \) and \( M^{n}(A) \in X_0 \). If \( g \) is not divisible by \( 2^m \), then these numbers are approximately equal because the corresponding areas are equal and \( g \) is a large number, and the exact equality holds only if \( g \) is divisible by \( 2^m \). Figure 2 shows the sets of points \( \{ A \mid A \in X_0, M^j(A) \in X_0 \} \) and \( \{ A \mid A \in X_1, M^j(A) \in X_0 \} \) for \( M = \left( \begin{array}{c} 10 \ 17 \\ 32 \ 43 \end{array} \right) \) and \( M = \left( \begin{array}{c} 3 \ 4 \\ 2 \ 1 \end{array} \right) \).

**Proposition 2.** For \( M = \left( \begin{array}{c} 10 \ 17 \\ 32 \ 43 \end{array} \right) \) and \( M = \left( \begin{array}{c} 3 \ 4 \\ 2 \ 1 \end{array} \right) \), the sequences of length \( 1, 2, \ldots , \ell \) in a stream of bits generated with matrix \( M \) are equiprobable, where \( \ell = 2^t - 1, \ell = (t-1)/2 \) and \( \ell = (t-1)/2 \). Here \( g = p \times 2^t \), where \( p \) is an odd prime, and the matrices correspond to the realizations GM29-SSE, GM58-SSE and GM55-SSE, respectively.

**Proof.** Let us check that the image of the lattice \( g \times g \) with the transformation \( M^j \) is invariant with respect to the shift for \( j = 0, 1, \ldots ,n \) and \( n \leq \ell \). In particular, the invariance takes place if there are integers \( r, l < t \) such that the distance between integer vectors \( (x+g_1^{j-1}, y+g_2^{j-1}) \) and \( (x, y) \) after applying transformation \( M^j \) is equal to \( (g/2, 0)^T \mod g \). This results in \( (m_1^{(j)} / 2^r + m_2^{(j)} / 2^l)^T \). For the matrix \( M = \left( \begin{array}{c} 2 \ 2 \\ 1 \ 1 \end{array} \right) \), the condition is satisfied when \( r = j/2, l = j/2 - 1 \) for even \( j \) and \( r = (j-1)/2, l = (j+1)/2 \) for odd \( j \). Thus, \( \ell = \max(j+1) = 2t - 1 \). Similarly, for each of the matrices \( M = \left( \begin{array}{c} 10 \ 17 \\ 32 \ 43 \end{array} \right) \) and \( M = \left( \begin{array}{c} 3 \ 4 \\ 2 \ 1 \end{array} \right) \), the condition is satisfied for \( \ell = (t-1)/2 \). Proposition 2 is proved.

Generally, the following statements are also valid. Consider a matrix \( M \) with integer elements and the following integer quantities: \( g = p \times 2^t \), \( q = \text{det} M = 2^u w (\mod g) \), \( k = \text{Tr} M = 2^{m_r} (\mod g) \), \( u \geq 1, t \geq v, m > 0 \). Here \( r, w \) are odd integers and \( p \) is an odd prime. Then \( i) \) all \( 2^t \) sequences of length \( j \) in a stream of \( v \)-bit blocks generated with recurrent relation (1) are equiprobable for \( j = 1, 2, \ldots, \ell \). Here \( \ell = \left\lfloor \frac{(t-v)}{\lfloor u/2 \rfloor} \right\rfloor \) for \( u \leq 2m \) and \( \ell = \left\lfloor \frac{(t-v)}{(u-m)} \right\rfloor \) for \( u > 2m \); \( ii) \) if \( k \) is even, then the image of the lattice \( g \times g \) with the transformation \( M^j \) is the lattice \( p \times p \) on the torus; \( iii) \) if \( k \) is odd, then the image of the lattice \( g \times g \) with the transformation \( M^{[j/u]} \) is not invariant with respect to the shift \( S \). Table 2: Numbers of failed tests for the batteries of tests SmallCrush, Crush, BigCrush [18], and Diehard [18]. Testing was performed with package TestU01 version TestU01-1.2.3. For each battery of tests, three numbers are displayed: the number of statistical tests with \( p \)-values outside the interval \([10^{-3}, 1 - 10^{-3}]\), number of tests with \( p \)-values outside the interval \([10^{-5}, 1 - 10^{-5}]\), and number of tests with \( p \)-values outside the interval \([10^{-10}, 1 - 10^{-10}]\).

**Table 2:** Numbers of failed tests for the batteries of tests

| Generator | SmallCrush | Crush | BigCrush |
|-----------|------------|-------|----------|
| MRG32k3a | 0.0,0      | 0.0,0 | 0.0,0    |
| LFSR113  | 0.0,0      | 1.0,0 | 6.6,6    |
| MT19937  | 0.0,0      | 2.2,2 | 2.2,2    |
| GM29.1-SSE | 0.0,0 | 0.0,0 | 0.0,0    |
| GM55.4-SSE | 0.0,0 | 0.0,0 | 0.0,0    |
| GQ58.1-SSE | 0.0,0 | 0.0,0 | 0.0,0    |
| GQ58.3-SSE | 0.0,0 | 0.0,0 | 0.0,0    |
| GQ58.4-SSE | 0.0,0 | 0.0,0 | 0.0,0    |

10003-p4
Table 3: CPU time (s) for generating 10^9 random numbers. Processors: Intel Core i7-940 and AMD Turion X2 RM-70. Compilers: gcc 4.3.3, icc 11.0.

|          | gcc  -O0 | gcc  -O1 | gcc  -O2 | gcc  -O3 | icc  -O0 | icc  -O1 | icc  -O2 | icc  -O3 | Source |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|--------|
| MT19937  | 13.7     | 5.7      | 6.9      | 2.6      | 17.5     | 6.5      | 2.9      | 2.9      | [9]    |
| MT19937-SSE | 5.2  | 4.8      | 5.5      | 2.0      | 4.9      | 4.7      | 2.4      | 2.0      | [5]    |
| LFSR113  | 10.4     | 4.8      | 6.8      | 3.1      | 10.2     | 5.0      | 4.6      | 4.5      | [11]   |
| LFSR113-SSE | 8.0 | 6.8      | 6.8      | 6.9      | 7.3      | 6.9      | 6.6      | 6.5      | [5]    |
| MRG32k3a | 47.9     | 36.3     | 35.3     | 25.0     | 56.1     | 33.1     | 22.8     | 28.1     | [19]   |
| MRG32k3a-SSE | 9.1 | 7.4      | 5.8      | 5.8      | 8.8      | 7.4      | 6.0      | 5.9      | [5]    |
| GM29.1-SSE | 22.6 | 19.6     | 17.5     | 18.1     | 21.2     | 18.7     | 18.2     | 18.1     | [16]   |
| GM55.4-SSE | 18.0 | 16.8     | 15.4     | 15.4     | 17.7     | 16.3     | 15.8     | 15.7     | [16]   |
| GQ58.1-SSE | 50.5 | 49.2     | 47.4     | 47.3     | 50.5     | 48.1     | 48.0     | 47.7     | [16]   |
| GQ58.3-SSE | 22.0 | 21.2     | 19.0     | 20.1     | 22.5     | 20.4     | 19.5     | 19.5     | [16]   |
| GQ58.4-SSE | 16.1 | 14.7     | 12.8     | 13.8     | 15.5     | 13.9     | 13.3     | 13.3     | [16]   |
| AMD Turion X2 RM-70 | gcc  -O0 | gcc  -O1 | gcc  -O2 | gcc  -O3 | icc  -O0 | icc  -O1 | icc  -O2 | icc  -O3 | Source |
| MT19937  | 31.0     | 17.8     | 10.8     | 7.1      | 31.0     | 18.7     | 5.2      | 4.9      | [9]    |
| MT19937-SSE | 11.3 | 10.3     | 11.1     | 6.6      | 10.8     | 9.9      | 6.0      | 6.0      | [5]    |
| LFSR113  | 14.6     | 8.7      | 9.6      | 5.3      | 14.9     | 9.1      | 6.9      | 6.8      | [11]   |
| MRG32k3a | 89.0     | 60.9     | 60.9     | 47.0     | 89.1     | 69.2     | 41.5     | 41.6     | [19]   |
| MRG32k3a-SSE | 25.9 | 22.3     | 18.4     | 18.3     | 25.6     | 22.3     | 19.0     | 19.0     | [5]    |
| GM29.1-SSE | 68.5 | 64.4     | 60.7     | 60.7     | 67.8     | 63.1     | 61.7     | 61.7     | [16]   |
| GM55.4-SSE | 59.8 | 54.8     | 53.1     | 53.0     | 58.2     | 53.6     | 52.8     | 52.8     | [16]   |
| GQ58.1-SSE | 179.6 | 179.6    | 178.3    | 177.8    | 183.1    | 178.3    | 178.5    | 178.5    | [16]   |
| GQ58.3-SSE | 75.5 | 73.9     | 70.6     | 71.1     | 74.2     | 71.9     | 70.4     | 70.4     | [16]   |
| GQ58.4-SSE | 51.9 | 51.0     | 48.2     | 48.1     | 53.1     | 49.4     | 48.2     | 48.1     | [16]   |

Therefore, according to the numerical results, the output value \( a^{(n)} \) has a uniform distribution with a very high accuracy.

In most cases the image of the lattice \( g \times g \) on the torus with \( M' \) where \( j \geq 2t \) is the \((p \times p)\)-lattice, therefore it is most interesting to study the deviations from the equidistribution for the \((p \times p)\)-lattice. I have calculated the exact areas on the torus which correspond to each of the sequences for \( M = \left( \begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix} \right) \). The calculations were carried out on a PC using Class Library for Numbers [17] for exact rational arithmetics. For each of the \( 2^n \) sequences of length \( n = 1, 2, \ldots \), the corresponding set of points on the unit two-dimensional torus consists of filled polygons. Exact rational coordinates of all the vertices of each filled polygon were found. Also, the exact number of points of the \((p \times p)\)-lattice inside each polygon was calculated. The total area of the polygons for each \( n \) was divided by \( 2^n n! \). The calculation of the \( n \)th area of the \((p \times p)\)-lattice on the torus, is found in [14] for \( p = 2^{29} - 3 \). The calculations for smaller values of \( p \) and larger values of \( n \) demonstrate that the dependence of \( \log(\sigma^2) \) on \( n \) is almost linear. Calculations show that the deviations from equidistribution are negligibly small in the sense that \( \sigma(\mathbf{A}_n) \) is much smaller than \( \langle \mathbf{A}_n \rangle = 1 \), for \( n < 6.8 \log p \). In particular, for \( p = 2^{29} - 3 \) the deviations are small for \( n < 130 \).

The variance for the several points of the orbit of matrix \( M \) on the \((p \times p)\)-lattice on the torus, is found to substantially depend on the number of points and on the value of \( p \), and only weakly depend (within several percent) on the distances between the points along the orbit.

![Fig. 3: Variance of the numbers of points of the \((p \times p)\)-lattice corresponding to sequences of length \( n \) vs. \( n \). The values are normalized such that \( \langle \mathbf{A}_n \rangle = 1 \).](image-url)
Statistical testing. — Table 2 shows the results of applying the SmallCrush, PseudoDiehard, Crush and BigCrush batteries of tests taken from [18], to the generators introduced in table 1. Batteries SmallCrush, PseudoDiehard, Crush and BigCrush contain 15, 126, 144 and 160 statistical tests, respectively. For each battery of tests, table 2 displays three characteristics: the number of statistical tests with p-values outside the interval $[10^{-3}, 1 - 10^{-3}]$, number of tests with p-values outside the interval $[10^{-5}, 1 - 10^{-5}]$, and number of tests with p-values outside the interval $[10^{-10}, 1 - 10^{-10}]$. Table 2 also contains the results of statistical tests for the Mersenne Twister generator of Matsumoto and Nishimura [9], combined Tausworthe generator of L’Ecuyer [11] and combined GQ58.1-SSE, GQ58.3-SSE and GQ58.4-SSE introduced in section 4.5.4 and sect. 4.6.1 in [20]). Both LFSR113 and MT19937 fail the test scomp LinearComp that is a linear complexity test for the binary sequences (see [18]), because the bits of LFSR113 and MT19937 have a linear structure by construction. Also LFSR113 fails the test smarsa MatrixRank (see [18]). The period lengths for the generators MRG32K3A, LFSR113 and MT19937 are $3.1 \cdot 10^{57}$, $1.0 \cdot 10^{44}$ and $4.3 \cdot 10^{6003}$, respectively.

The usefulness of a RNG for a specific application in physics depends on, possibly dangerous interferences of the correlations in the specific problem and those of the RNG. Modern statistical test suites contain tests that reveal known types of correlations for the RNGs, in particular, the types that are known to result in systematic errors in Monte Carlo simulations and that were studied in [21]. One concludes that the new realizations described in this paper possess excellent statistical properties.

Speed of the generator. — I have tested the CPU times needed for generating $10^9$ random numbers. The results are shown in table 3 for Intel Core i7-940 and AMD Turion X2 RM-70 processors, respectively. The results are summarized in table 3. Compilers are used for different compilers and optimization options. The compilers are used are GNU C compiler gcc version 4.3.3 and Intel C compiler icc version 11.0. The CPU times for the realizations GM29.1-SSE, GM55.4-SSE, GQ58.1-SSE, GQ58.3-SSE and GQ58.4-SSE introduced in table 1 are compared with those for the Mersenne Twister generator of Matsumoto and Nishimura [9], the combined Tausworthe generator of L’Ecuyer [11] and the combined multiple recursive generator proposed in [19].

***

The author thanks L. N. SHICHUR for useful discussions and remarks. The support of RFBR grant 11-07-00197 is acknowledged.

REFERENCES

[1] Beach K. S. D., Lee P. A. and Monthoux P., Phys. Rev. Lett., 92 (2004) 026401; Landau D. P. and Binder K., A Guide to Monte Carlo Simulations in Statistical Physics (Cambridge University Press, Cambridge) 2000; Pueber S. C. and Wiring R. B., Annu. Rev. Nucl. Part. Sci., 51 (2001) 53; Lüchow A., Annu. Rev. Phys. Chem., 51 (2000) 501; Bizzarri A. R., J. Phys.: Condens. Matter, 16 (2004) R83.

[2] Knuth D. E., The Art of Computer Programming, Vol. 2, 3rd edition (Addison-Wesley, Reading, Mass.) 1997.

[3] Falcioni M. et al., Phys. Rev. E, 72 (2005) 016220; Lozi R., Indian J. Ind. Appl. Math., 1 (2008) 1; Kocarev L. and Jakimoski G., IEEE Trans. Circuits Syst., 50 (2003) 123; Li P. et al., Phys. Lett. A, 349 (2006) 467; Patidar V. and Sud K. K., Electron. J. Theor. Phys., 6 (2009) 327; Pareek N. K. et al., Int. J. Netw. Secur., 10 (2010) 32.

[4] Barash L. and Shchur L. N., Phys. Rev. E, 73 (2006) 036701.

[5] Barash L. Yu. and Shchur L. N., Comput. Phys. Commun., 182 (2011) 1518.

[6] http://www.intel.com/support/procesors/pentium4/ab/CS-029967.htm.

[7] http://support.amd.com/us/Processor_TechDocs/24592.pdf.

[8] Toth J. P. R., Robinson W. D. and Eagle D. J., J. ACM, 20 (1973) 469; Fushimi M. and Tezuka S., Commun. ACM, 4 (1983) 516; Couture R., L’Ecuyer P. and Tezuka S., Math. Comput., 60 (1993) 749; Tezuka S. and L’Ecuyer P., ACM Trans. Model. Comput. Simul., 1 (1991) 99; L’Ecuyer P., Math. Comput., 65 (1996) 203.

[9] Matsumoto M. and Nishimura T., ACM Trans. Model. Comput. Simul., 8 (1998) 3.

[10] Panneton F., L’Ecuyer P. and Matsumoto M., ACM Trans. Math. Software, 32 (2006) 1.

[11] L’Ecuyer P., Math. Comput., 68 (1999) 261.

[12] L’Ecuyer P. and Simard R., ACM Trans. Math. Software, 33 (2007) article 22.

[13] Grothe H., Stat. Pap., 28 (1987) 233.

[14] L’Ecuyer P., Commun. ACM, 33 (1990) 85.

[15] Niederreiter H., Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing, edited by Niederreiter H. and Shiue P. J.-S., Lecture Notes in Statistics, Vol. 106 (Springer-Verlag) 1995.

[16] http://www.comphys.ru/barash/rng_sse2.zip.

[17] http://www.ginac.de/CLN/.

[18] L’Ecuyer P. and Simard R., TestU01: A Software Library in ANSI C for Empirical Testing of Random Number Generators (2002), Software user’s guide, http://www.iro.umontreal.ca/~simardr/testu01/tu01.html.

[19] L’Ecuyer P., Oper. Res., 47 (1999) 159.

[20] L’Ecuyer P., Handbook of Simulation, edited by JERRY-BANKS (Wiley) 1998, p. 93.

[21] Ferrenberg A. M., Landau D. P. and Wong Y. J., Phys. Rev. Lett., 69 (1992) 3392; Shchur L. N., Comput. Phys. Commun., 121-122 (1999) 83; Grassberger P., Phys. Lett., 181 (1983) 43; Shchur L. N., Heringa J. R. and Blöte H. W. J., Physica A, 241 (1997) 579; Shchur L. N. and Blöte H. W. J., Phys. Rev. E, 55 (1997) R4005; Schmid F. and Wilding N. B., Int. J. Mod. Phys. C, 6 (1995) 781.