DOI: 10.17725/rensit.2022.14.437

Information Technologies Based on Noise-like Signals: IV. Algorithmic Pseudo-random Number Generators Based on Dynamic Chaos

Vladimir I. Grachev, Viktor I. Ryabenkov, Anastasiya V. Surgay, Vladimir V. Kolesov

Kotelnikov Institute of Radioengineering and Electronics of RAS, http://www.cplire.ru/
Moscow 125009, Russian Federation

E-mail: grachev@cplire.ru, ryabenkov.vi@list.ru, ya.a1997@yandex.ru, kvv@cplire.ru

Received December 11, 2022, peer-reviewed December 15, 2022, accepted December 21, 2022

Abstract: **Numerical simulation is used to investigate the statistical, fractal and structural properties of sequences of integers generated by the algorithm with delay. It is shown that the statistical properties of the generated discrete sequences, close to a random process, are provided by such generating coding algorithms, in which both the one-dimensional probability distribution and the distributions of the conditional probabilities of the generated numbers are close to uniform. The structure of the phase space of a discrete coding algorithm with delay defined on a closed interval of integers is studied. It is established that the phase space consists of a finite number of cycles of different periods, the behavior of the system on which is pseudorandom. The possibility of creating generators of this type with more complex circuits is discussed. It is shown that with an appropriate choice of parameter values, the algorithm allows the formation of a non-periodic pseudo-random sequence of arbitrary given length for encoding information in telecommunication systems.**

Keywords: **information technology, chaotic dynamics, pseudorandom sequences, redundant codes, noise-like signals**

UDC 621.391

Acknowledgments: The work was carried out within the framework of the state task of the Kotelnikov IRE of RAS from the Ministry of Education and Science of the Russian Federation.

For citation: Vladimir I. Grachev, Viktor I. Ryabenkov, Anastasiya V. Surgay, Vladimir V. Kolesov. Information Technologies Based on Noise-like Signals: IV. Algorithmic Pseudo-random Number Generators Based on Dynamic Chaos. *RENSIT: Radioelectronics. Nanosystems. Information Technologies*, 2022, 14(3):437-462e. DOI: 10.17725/rensit.2022.14.437.

Contents

1. INTRODUCTION

Currently, in a number of science areas and technology, random and pseudo-random numbers are widely used in the process of solving practical problems. These areas include mathematical modeling, cryptography, information security in computers and telecommunications networks, as well as when encoding information in ultra-wideband radio systems. To solve these problems, it is necessary to generate random numbers huge arrays with a wide variety of properties. Of greatest importance for practice are numerical sequences with a uniform distribution law. The random numbers problem is that there is no algorithmic random number generator yet. If the resulting sequence obeys some algorithmic regularity, then by definition it is not random.

Thus, the main task of such an algorithm is to generate a sequence of numbers that, not being random, would be indistinguishable from random, would not have visible patterns. In this sense, algorithms are bad and good. Moreover, the quality of the algorithm, i.e. its ability to generate numerical sequences close in properties to random ones can be verified by methods of mathematical statistics.

One of the main elements in such systems are random and pseudo-random number generators (RNG and PRNG), the quality and speed of which significantly affect the results of solving the tasks. Currently, intensive fundamental work is being carried out in the field of generating random and pseudo-random numbers, and a large number of patents and inventor's certificates are being published, which indicate an ever-increasing interest in these areas. Pseudo-random sequence generators are used in numerous applications where sequences with properties similar in their statistical characteristics to random number series are needed. The sequences of numbers formed by such generators are calculated using deterministic algorithms, which was the reason to call them pseudorandom sequences (PRS) [1].

The characteristics of these sequences are subject to a variety of often specific requirements associated with the characteristics of their specific applications. Because of this, interest in the development of new algorithms that form such pseudo-random sequences, not only does not decrease, but rather grows. This is also due to the urgent need to protect information in systems and networks that are rapidly developing on a global scale for processing, storing and transmitting information [2]. The emergence of new ideas in this area, in particular, is associated with the development of ideas about the possibility of chaotic dynamics of deterministic systems even under the assumption that they do not contain any noise [3].

Despite the fact that quite a few algorithms for generating pseudo-random sequences (PRSs) are known, in practice, as a rule, a recurrent algorithm is used. Binary sequences based on recurrence relations are quite easily implemented on a computer in the form of programs and circuitry based on high-speed multi-bit binary shift registers. The known classes of PRS, both linear and non-linear, have certain disadvantages and do not satisfy all the necessary requirements. An alternative solution to the problem is the use of noise-like signals (NLS) generated by nonlinear systems with dynamic chaos. Such NLSs, having correlation properties no worse than those of M-sequences, have a practically unlimited set of lengths, can form ensembles of both binary and multilevel signals of large volumes and are non-linear, which makes it difficult to recognize them for subsequent playback in case of unauthorized access to the coded information [4].

On the basis of a mathematical model of a ring self-oscillating system with strong amplitude-phase nonlinearity, filtering and delay, a discrete generating algorithm for a

chaotic signal has been developed and studied, which belongs to the class of algorithms of a recurrent-parametric type with delay. The algorithm form of this class in general terms has the form of a discrete functional transformation (mapping):

$$x_n = f(x_{n-1}, x_{n-2}, ..., x_{n-Nz}),$$

where x_n are the members of the generated pseudo-random sequence at the n-th step, Nz is the delay parameter that determines the number of sequence members on the delay interval x_{n-1}, x_{n-2}, ..., x_{n-Nz}, which completely determine the new value of x_n and must be are given as the initial condition at the first step, and the function $f(x)$ reflects the amplitude and phase transformations in the generating ring self-oscillatory system in the chaos mode.

The algorithm is defined on the integers set M of the natural series belonging to the closed numerical interval [M1, M2], (M2 > M1, M = M2 – M1 + 1), and forms a practically uncorrelated pseudo-random sequence of integers with a probability distribution close to uniform, and correlation characteristics that meet the requirements for coding signals. The advantage of integer sequences is that they are identically reproduced on various types of computing devices and, when implemented in hardware, are easily reproduced in circuitry [5].

One of the simplest generators that form pseudo-random sequences are generators based on the Fibonacci algorithm, which are still used in practice [6].

In the Fibonacci algorithm, when calculating each next sequence member, several previously calculated previous members are used. This is the so-called generator with lagging arguments. As a rule, a limited numerical interval is used as the definition domain of the phase space in which the representing point of the system state moves. Due to the limitedness of the phase space determined by the dimension of the algorithm, taking into account the finite accuracy of the numbers representation, sooner or later, as a result of a successive calculations chain according to a given deterministic algorithm, the trajectory of the system in its phase space must close.

This means that the trajectory will enter a cycle, and then the calculation results will be repeated after some certain (although it may be very large) number of calculation steps, which is called the period.

When developing such algorithms, they seek to find conditions for obtaining numbers sequences that have the largest possible (largest) repetition period, and at the same time, on any arbitrary trajectory section, less than the period length, possessing the characteristics random numbers sequence. The advantage of class of the algorithms with delay is that, despite the great simplicity of computational operations, they allow us to study the patterns of sequences formation with large periods depending on the characteristic parameters of the algorithm (of the interval for determining the allowable numbers {1, M}, of the length of the delay Nz). The algorithm is supplemented by the returning rule a newly calculated number to in the specified interval in case of exit from it. This operation provides an important for chaotization mixing mechanism [7].

2. PSEUDO-RANDOM SEQUENCE OF INTEGERS GENERATED BY A ALGORITHM WITH DELAY AS A MARKOV PROCESS

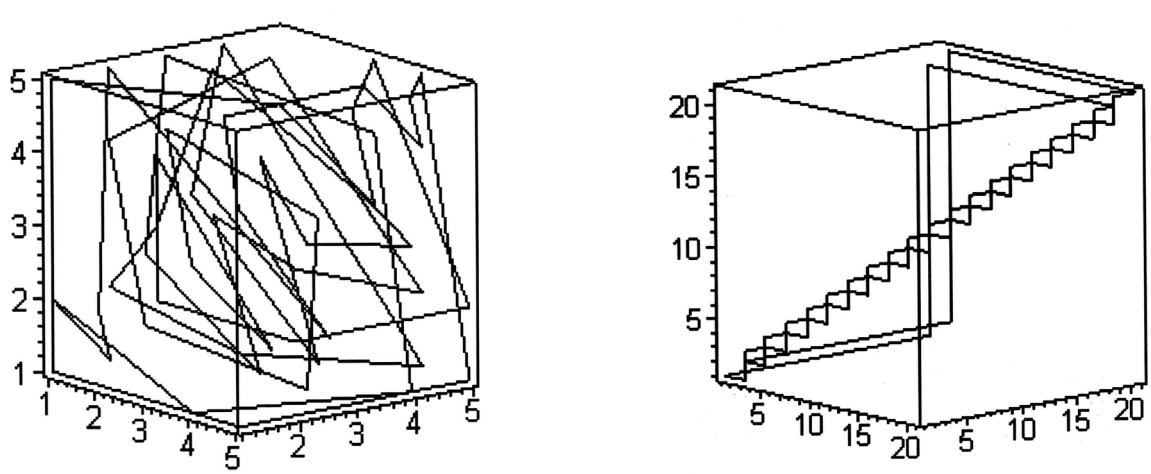
The problem of protecting information in open information and computer networks from unauthorized access, as well as the task of increasing the noise immunity of telecommunication channels, are associated with the use of complex coding algorithms and noise-like signals with a large information capacity. Therefore, the development of complex coding algorithms and criteria for an objective assessment of their statistical properties is a rather urgent task.

As a test algorithm, we consider an algorithm with a delay based on a Fibonacci-type mapping. To restrict the definition domain of the algorithm to a finite closed integer interval [1, M], M > 1, the mapping is supplemented by the operation of converting the interval [1, M] into itself with "reflecting boundaries":

$$\begin{aligned} &\tilde{x}_n = x_{n-1} + (-1)^{x_{n-Kz}} \cdot x_{n-Nz}, \quad Kz \in [2,\ Nz-1], \\ &x_n \in [1,\ M], \\ &x_n = \tilde{x}_n, \quad if \quad \tilde{x}_n \in [1,\ M], \\ &x_n = \tilde{x}_n - M, \quad if \quad \tilde{x}_n > M, \\ &x_n = \tilde{x}_n + M, \quad if \quad \tilde{x}_n < 1. \end{aligned} \tag{2.1}$$

Here Nz is the delay parameter, it determines phase space (PS) dimension and the radius vector $R_n(x_{n-1}, x_{n-2}, \ldots, x_{n-Nz})$ of this discrete dynamic system (DDS) state in this space. The number of possible states in the PS is finite and equal to M^{Nz}. Depending on the initial conditions $R_0(x_{-1}, x_{-2}, \ldots x_{-Nz})$, DDS (2.1) at each step of the algorithm describes one or another "trajectory" in the PS, which are sequential discrete transitions from one state point to another according to pseudorandom law (**Fig. 2.1**). Due to the limited scope of the PS, these trajectories form closed cycles, which, due to the uniqueness of the transformation (2.1), do not intersect and have no common points. All PS points belong to only one cycle or an isolated point with coordinates (M,M,… M). So, for example, with M = 5, Nz = 4, Kz = 3, the PS of the algorithm has one 562-stroke cycle, two cycles with a period T = 27, one 8-stroke cycle, and one singular point. At Nz = 5, Kz = 3 and M = 13 (M^{Nz} = 371293) there are cycles in the PS with periods T = 332373, 21721, 7966, 4959, 3640, and at Nz = 6, Kz = 4 and M = 15 one " long" cycle with period T = 11099897 includes the vast majority (0.974 M^{Nz}) of the PS points.

The cycles of the algorithm (1.1) have an important distinctive feature: the behavior of the dynamic system on the cycle before its closure (and we will be interested in processes just before the cycle is closed) has a random, chaotic character (Fig. 2.1). In this case, the non-periodic sequence {xn} generated by the algorithm is of a pseudo-random type. The set of points of states of a dynamic system in an PS, united in such a cycle, called a pseudo-random cycle. In contrast to the regular cycle, which corresponds to regular motion in the phase space before the cycle closes. An example of a simple regular cycle ($x_n = x_{n-3} + 2$ with the transformation of the interval [1,21] into itself) is shown in Fig. 2.1*b*. Thus, a pseudo-random cycle is a finite set of seemingly chaotic, but successive points of states of a discrete dynamical system in the phase space of the algorithm in a strictly deterministic way.



a *b*

Fig. 2.1. *Phase portrait of a signal with closed a) pseudo-random cycle, b) regular cycle.*

For practical applications, non-periodic pseudo-random sequences of great length are of the greatest interest. With an appropriate choice of parameters of the algorithm (2.1) and initial conditions, the segment of the non-periodic PRS, generated by the algorithm on a pseudo-random cycle before the system exits for a period, can be arbitrarily long and, as analysis has shown, in terms of its statistical properties, it is close to a sequence with a uniform probability distribution of the generated numbers $p(x)$. So, at the algorithm definition domain interval [1,63] (M = 63) and the delay Nz = 3, the length of the non-periodic SRP is $N = 7.8317 \cdot 10^4$, at Nz = 5 $N = 3.3174 \cdot 10^8$, at Nz = 7 $N = 1.676 \cdot 10^{12}$, at Nz = 9 the length of the non-periodic PRS is more than $5 \cdot 10^{12}$.

Note that the discrete PRS with discrete numbers $\{x_n\}$ is close in its form to the sequence of tests of the classical probability theory. Each transition in this sequence from the number xn to the next number x_{n+1}, as well as to the number x_{n+s} after s steps of the algorithm, is completely determined due to the determinism and uniqueness of the process (2.1). However, to an outside observer, it is no different from a process of random testing. Abstracting, therefore, from the determinism of process (2.1), we show that the sequence formed by it, with an appropriate choice of algorithm parameters, can be very close to a random sequence of Markov type and, moreover, to a random sequence of independent equally probable events.

As is known, a Markov process is a process without a probabilistic aftereffect, when the conditional probability for all $t > t_0$ is uniquely determined by the value of x_0 taken at the moment t_0 and does not depend on the previous history [8]. For a discrete sequence with discrete values x_n – a simple Markov chain, this means that there is a probability $p(x_j, n \mid x_i, k)$ of transition from any of the process values x_i at the k-th trial to any value x_j at the n-th trial ($n > k$, $i,j = 1, 2, \ldots, M$) [9]. In a particular case of a sequence of independent trials, the probability of transition to the state x_j coincides with the probability of this state in the nth trial $p(x_j, n \mid x_i, k) = p(x_j)$ regardless of the results of other trials. For a homogeneous Markov chain, the transition probabilities depend only on the number of steps $s = n - k$ between trials $p(x_j, n \mid x_i, k) = p(x_j, \mid x_i, s) = p_{ij}(s)$. The values $p_{ij}(s)$ form a matrix π_s of transition probabilities in s steps. For a homogeneous chain, the relation (Markov equation) [10] must be satisfied: $\boldsymbol{\pi}_s = (\boldsymbol{\pi}_1)^s$, i.e. transition probabilities in s steps are expressed in terms of transition probabilities in one step.

Consider algorithm (2.1). At first glance, the sequence generated by this algorithm with delay is not a process without aftereffect. Further more, each new value of the PRS is determined by the prehistory from the Nz values of the delay adopted at the previous stages. On the other hand, the operation of "transforming a numerical interval into itself" sort of breaks this connection (without violating the uniqueness of the process in the forward direction, but making it irreversible) with each ejection of a new number beyond the interval boundaries [1,M]. Let us check whether the relation $\pi_s = (\pi_1)^s$ is valid for the PRS formed by algorithm (2.1). That is, to what extent this sequence corresponds to the Markov equation. The assumption about the homogeneity of the process $\{x_n\}$ is quite natural if there is preliminary information about the closeness of the probability distribution $p(x)$ to the uniform one.

To visualize the results, we will carry out a numerical experiment for algorithm (2.1) with small values of parameters, but with the presence in the phase space (PS) of a pseudo-random cycle with a period sufficient to non-periodic sequence generate with the number of terms N, which providing the array necessary for statistical processing. Let M = 3, Nz = 9, Kz = 5, i.e. the algorithm has a 9-dimensional phase space and a definition domain of three numbers. In this case, there is a long cycle in the PS with a period T = 19677 with a total volume of the PS equal to $M^{Nz} = 19683$. The probability distribution of the

numbers $p(x)$ in the sequence generated by the algorithm is almost uniform with a root-mean-square deviation from this law equal to $4.8\cdot10^{-5}$, and the maximum deviation modulo $6.8\cdot10^{-5}$.

Based on the implementation of the PSP generated by the algorithm with length $N = 19677$, we determine the probabilities $P(A)$ of generating the number x_i (i = 1, 2, ..., M) by counting the occurrences of event A equal to $n(x_i)$: $P(A) = n(x_i)/\mathrm{N}$. Following the definition of conditional probability according to Kolmogorov: $P(B \mid A) = P(AB)/P(A)$, where $P(AB)$ is the probability of the simultaneous occurrence of events A and B. In the implementation of N tests, we will count the number of $n(x_j, x_i, s)$ that occurred simultaneously events A (generation of the number x_i) and B (transition from this number through s steps of the algorithm to the number x_j). Then the probability $P(AB) = n(x_j, x_i, s)/(N - s)$, and the conditional probability $P(B \mid A) = [\mathbf{n}(x_j, x_i, s)/(N - s)]/[n(x_i)/N] = p_{ij}(s)$. The transitions matrix $\boldsymbol{\pi}_s = \|p_{ij}(s)\|$. For a sequence of independent equally probable events, all $p_{ij}(s) = 1/\mathrm{M}$ and the corresponding transitions probability matrix will be denoted by $\boldsymbol{\pi}_0$.

When analyzing the PRS implementation with length $N = 19677$, the following transition matrices were obtained:

$$\pi_1 = \begin{pmatrix} 0.33335 & 0.33335 & 0.33335 \\ 0.33325 & 0.33340 & 0.33340 \\ 0.33320 & 0.33350 & 0.33320 \end{pmatrix},$$

$$\pi_2 = \begin{pmatrix} 0.33321 & 0.33352 & 0.33337 \\ 0.33326 & 0.33342 & 0.33342 \\ 0.33315 & 0.33337 & 0.33321 \end{pmatrix}, \ldots,$$

$$\pi_{20} = \begin{pmatrix} 0.33321 & 0.33382 & 0.33382 \\ 0.33342 & 0.33357 & 0.33372 \\ 0.33330 & 0.33367 & 0.33376 \end{pmatrix}.$$

The establishment of this fact alone, that all probabilities of transitions $p_{ij}(s) = p(x_j, \mid x_i, s) = p(x_j, n \mid x_i, \mathrm{k})$ exist, is already sufficient to consider this process as a Markov one [11]. In addition, we see that all matrix elements are very close to the equiprobable value $p_{ij} = 1/\mathrm{M} = 1/3$. At this, since the sum of the elements of each row of matrices π_s is equal to one, these matrices are stochastic [12]. The largest difference between the Euclidean norms of the matrices π_s from unity was less than 10^{-5}.

We will evaluate the validity of equality (2.2) for the PRS under study based on the calculation of the rms deviation of the elements of the matrices $\boldsymbol{\pi}_s$ and $(\boldsymbol{\pi}_1)^s$:

$$\sigma_S = \sqrt{(1/M^2)\sum_{i,j=1}^{M}(p_{i,j}(s) - p_{i,j}^{(s)}(1))^2} = \\ = (1/M)\cdot\|\Delta\pi_s\|, \qquad (2.2)$$

where $\|\boldsymbol{\Delta\pi}_s\|$ is the Euclidean norm of the matrix $\boldsymbol{\Delta\pi}_s = \boldsymbol{\pi}_s - (\boldsymbol{\pi}_1)^s$, and $p_{ij}(s)$ (1) are the elements of the matrix $(\boldsymbol{\pi}_1)^s$.

The obtained numerical values of σs are plotted on the graph in **Fig. 2.2** (curve 1*a*). We see that the differences of the matrix elements in the left and right parts of (2) for all transition intervals s = 1, 2, …, 20 are less than $4\cdot10^{-4}$ in absolute value or about 10^{-3} in relative value. This result shows that the tested sequence generated by algorithm (2.1) with delay can be considered as very close to a Markov process. More over, as the analysis of the proximity of the transitions matrices $\boldsymbol{\pi}_s$ to the matrix $\boldsymbol{\pi}_0 = \|p_{ij} = 1/\mathrm{M}\|$ shows, this sequence for the given values of the parameters M and Nz practically does not differ from the independent equiprobable

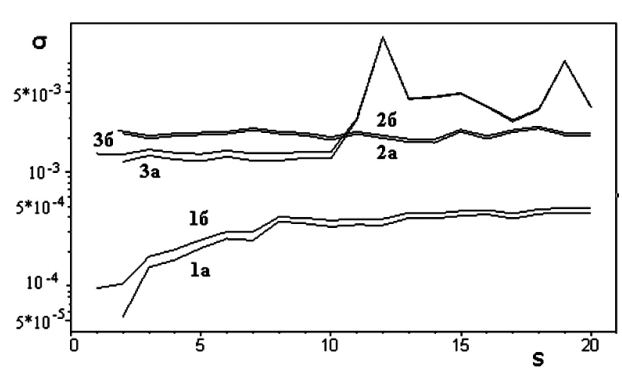


Fig. 2.2. *Root-mean-square deviation of matrix elements* $\boldsymbol{\pi}_s$ *and* $(\boldsymbol{\pi}_1)^s$.

tests sequence. Indeed, characterizing the difference between the matrices $\boldsymbol{\pi}_s$ and $\boldsymbol{\pi}_0$ by the rms deviation of their elements:

$$\sigma_O = \sqrt{(1/M^2)\sum_{i,j=1}^{M}(p_{i,j}(s)-1/M)^2} = (1/M)\cdot\|\Delta\pi_0\|, \quad (2.3)$$

where $\|\boldsymbol{\Delta\pi}_0\|$ is the norm of the matrix $\boldsymbol{\Delta\pi}_0 = \boldsymbol{\pi}_s - \boldsymbol{\pi}_0$, we plot the values of σ_0 obtained in the numerical experiment for s = 1, 2, …, 20 on the graph of Fig. 2.2 (curve 1*b*). We see that the differences transition matrices elements $p_{ij}(s)$ from the values of 1/M upon all analyzed transitions s do not exceed the level of $5\cdot10^{-4}$ or, in relative terms, about 0.1%.

Similar calculations were performed in the analysis of the transition matrices for the PRS of algorithm (2.1) with the parameters M = 9, Nz = 9, Kz = 5, i.e., with a definiton domain of nine numbers and a PS with a volume of 387420489 points of states. The length of the studied implementation of the PRS with the initial vector R_0(1, 1, …, 1) was chosen to be N = 180000. The probability distribution of the generated numbers $p(x)$ is close to uniform with a root-mean-square deviation from this law equal to $6.42\cdot10^{-4}$ and a maximum deviation modulo $1.45\cdot10^{-3}$. Differences in the elements of the transitions matrices $\boldsymbol{\pi}_s$ and $(\boldsymbol{\pi}_1)^s$, as well as the matrices $\boldsymbol{\pi}_s$ and $\boldsymbol{\pi}_0$, are shown by graphs 2*a* and 2*b* in Figs. 2.2. The values of the root-mean-square deviation, as it follows from the constructed dependencies, are at the level $\sigma = 2\cdot10^{-3}$, i.e. less than 1% in relative value. This also testifies in favor of making a conclusion about the closeness of the generated pseudo-random process to the Markov chain and to the sequence of independent equiprobable trials.

Curves 3*a* and 3*b* in Fig. 2.2 refer to the case of the PRS formed by algorithm (2.1) with the following parameters: M = 9, Nz = 5, Kz = 3. This variant differs from the previous case by a smaller magnitude of delay. In PS of algorithm presented cycles with periods T = 55070, 3230, 260, 130, 50. The longest pseudo-random cycle with the initial vector R_0(1, 1, …, 1) was taken for numerical analysis. The length of the studied implementation of the PRS is N = 55070. The probability distribution of the generated numbers $p(x)$ is close to uniform with a standard deviation from this law of $7.3\cdot10^{-4}$, and a maximum deviation modulo $1.05\cdot10^{-3}$, which practically does not differ from the degree of closeness to a uniform distribution of generated numbers in the previous test sequence.

Indeed, as studies have shown, the probability distribution function tends to improve as the delay parameter, and hence the dimension of the algorithm, increases, i.e. to approximation to a uniform law, but for Nz of the order of 6 and more, the density of distribution $p(x)$ practically does not differ from this law. As evidenced by the course of curves 3*a* and 3*b*, the transitions matrices $\boldsymbol{\pi}_s$ and $(\boldsymbol{\pi}_1)^s$, as well as the matrices $\boldsymbol{\pi}_s$ and $\boldsymbol{\pi}_0$ differ little over the transition intervals s = 1, 2, …, 10, but over large intervals s = 12, 13, …, the differences in matrices increase to units of percent. This computer experiment confirms that knowledge about the uniformity of the probability distribution of the occurrence of numbers $p(x)$ is still not enough to estimate the PRS as close to a sequence of independent tests. It is important that and all distributions of conditional probabilities $p(x_j, n \mid x_i, k)$ be uniform. Comparison of the course of graphs 2 and 3 in Fig. 2.2 shows that an increase in the delay from Nz = 5 to Nz = 9 leads to an improvement in the statistical characteristics of the pseudo-random process generated by the algorithm, bringing them closer to the characteristics of a Markov chain and a sequence of independent equiprobable events.

Note that the determination of matrices of probabilities transitions for large values of the parameter M requires processing large numerical arrays, therefore, for express analysis of the statistical quality of the generated PRSs, it is quite acceptable, as is known, to construct simplified transition matrices that inform only whether the probabilities of transition of $p_{ij}(s)$ are different from zero or not. The construction

of such matrices is possible when analyzing implementations that are not necessarily large in length. Herewith, not all matrics cells may turn out to be correctly filled: for large values of M and insufficient length N of the analyzed PRS ($N < s \cdot M^2$), such a matrix, even for processes with all nonzero $p_{ij}(s)$, has the form of a uniformly filled "starry sky". Nevertheless, consideration of the form of transitions matrices $p_{ij}(s)=, \neq 0$ sequentially in number s, gives important information about the quality of discrete process under study.

A process with a delay is a process with an aftereffect. However, the introduction of the operation of converting the interval [1,M] into itself into the algorithm breaks this aftereffect at each step when the number x_n goes beyond the boundaries of this interval. Herewith, the pseudo-random sequence formed by algorithm (2.1), with an appropriate choice of algorithm parameters, can actually be considered as a process without aftereffect, i.e., as a simple homogeneous Markov chain with probabilities of transitions $p_{ij}(s) \approx 1/M$. It is shown that with an appropriate choice of algorithm parameters and initial conditions, under which the difference between the transitions matrices $\boldsymbol{\pi}_s$ and $(\boldsymbol{\pi}_1)^s$ becomes noticeable, the statistical properties of the generated PRS worsen compared to a purely random process, even if herewith the probability distribution $p(x)$ is practically uniform.

3. PRS COMBINED GENERATOR

In [13], the characteristics of pseudo-random sequences defined on a limited interval of integers, formed by the simplest algorithms such as the Fibonacci algorithm, are proposed and studied. Expressions are given, as well as a method for calculating the maximum period TM, Nz of the PRS for a standard Fibonacci-type generator, depending on the interval of integers {1,M} and the delay parameter (dimension of the phase space) Nz. Knowing the exact value of the maximum period TM, Nz makes it possible to combine two generators, to significantly improve the statistical properties in such a way that the PRS period becomes many times greater than the period of each individual partial generator. The work of each such generator is performed in accordance with the algorithm for generating and returning a newly calculated value to the definition domain {1, M}

$$X_n = X_{n-1} + X_{n-Nz},$$
$$X_n = X_n - M \; for \; X_n > M. \tag{3.1}$$

The definition domain {1, M} and the delay Nz are different for each of the partial generators. The combined PRS generator functions as follows: two partial generators operate synchronously and the numbers generated by them at each step are added, generating a new sequence. If the result of addition is outside the interval {1, M_0}, then the return algorithm is switched on, similar to (3.1). The above can be written as:

$$\begin{aligned} &X_{1,n} = X_{1,n-1} + X_{1,n-Nz1} \rightarrow \\ &\rightarrow X_{1,n} = X_{1,n} - M_1 \; for \; X_{1,n} > M_1; \\ &X_{2,n} = X_{2,n-1} + X_{2,n-Nz2} \rightarrow \\ &\rightarrow X_{2,n} = X_{2,n} - M_2 \; for \; X_{2,n} > M_2; \\ &X_{0,n} = X_{1,n} + X_{2,n} \rightarrow \\ &\rightarrow X_{0,n} = X_{0,n} - M_0 \; for \; X_{0,n} > M_0. \end{aligned} \tag{3.2}$$

In accordance with the results of [6], in order to obtain the maximum period, it is necessary to set the initial conditions (IC) as a sequence of Nz units. Then, if $Nz_1 > Nz_2$, then for the 1st generator, in order to achieve the maximum period, IC – a sequence of Nz_1 units. And the 2nd generator as a IC, respectively, has Nz_2 units. The numbers missing to start the operation of the combined generator algorithm ($Nz_1 - Nz_2$) must first be calculated using the algorithm of the 2nd partial generator. Thus, for the combined generator, the ICs are actually Nz_1 of numbers, and the first Nz_2 of them are equal to 2. Therefore, the ICs for the combined generator will be repeated and the sequence generated by it will reach the period (T_0) when exactly (and simultaneously) the ICs are repeated for each

from partial generators. The corresponding condition can be written as:

$$N_1 T_{M1,Nz1} = N_2 T_{M2,Nz2} = T_0, \quad (3.3)$$

where N_1 and N_2 are integers. Thus, T_0 must be divisible by $T_{M1,Nz1}$ and $T_{M2,Nz2}$ without remainder and, therefore, the maximum value for T_0 is determined by the product $(T_{M1,Nz1})(T_{M2,Nz2})$ and to achieve this value it is necessary that $T_{M1,Nz1}$, $T_{M2,Nz2}$ would have no common the comultipliers and, moreover, would be multiples.

Knowing the dependence of the PRS period on the maximum value in the definition domain of M and of the delay Nz, it is possible to choose such M_1, Nz_1 and M_2, Nz_2 that the period of the combined oscillator will significantly exceed the periods of the partial oscillators $T_{M1,Nz1}$ and $T_{M2,Nz2}$. In the case of small periods, this can be checked fairly easily. For example, $T_{3,2} = 8$ and $T_{4,3} = 14$. The smallest number that is divisible by 8 and 14 without a remainder is 56. It is this value that is obtained as a result of direct generation; $T_{15,7} = 97655$, $T_{17,8} = 83520$, the period of the generated numbers sequence for the combined generator is 1631229120, which is much larger of each as periods of the partial generators. As an example in **Fig. 3.1** shows the frequency distribution of the appearance of integers in the PRS for the combined generator in the definition interval {1,257}, while the partial generators parameters are as follows: $M_1 = 257$, $Nz_1 = 11$, $M_2 = 253$ and $Nz_2 = 15$. This distribution was obtained for an array of 10^9 numbers, the maximum value is 3896607, the minimum is 3885514, the difference between them is 11093 and the difference referred to the maximum value is 0.0028; average value ~$3.89 \cdot 10^6$; rms deviation ~$1.89 \cdot 10^3$. The given frequency distribution of integers is close to uniform according to statistical criteria.

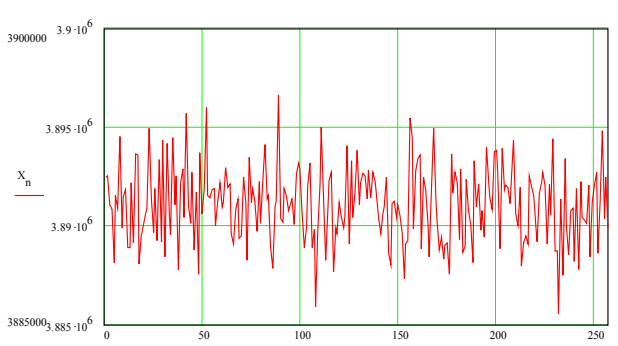


Fig. 3.1. *Distribution of the appearance of integers in the PRS for the combined generator in the definition interval {1,257}.*

The proposed algorithm and its characteristics are of methodological interest, since to obtain a period of arbitrarily long duration, the number of master oscillators can be increased, and X_{0n} can be a linear combination of the numbers X_n with weight coefficients C_n different from unity for each of the partial generators, i.e.

$$X_{0n} = \sum_{i=1}^{N} C_{in} X_{in}. \quad (3.4)$$

4. CHAOTIC ENCODING ALGORITHM BASED ON TWO-DIMENSIONAL MAPPING

A one-dimensional algorithm of the Fibonacci random number generator type $x_n = f(x_{n-1},\ldots, x_{n-Nz}, Nz, M)$ [13] was chosen as the basic discrete algorithm. General view of the two-dimensional algorithm under study:

$$\begin{aligned} x_n &= f_1(x_{n-1},\ldots,x_{n-Nz1}, y_{n-1},\ldots,y_{n-Nz2}, Nz1, Nz2, M), \\ y_n &= f_2(y_{n-1},\ldots,y_{n-Nz2}, x_{n-1},\ldots x_{n-Nz1}, Nz1, Nz2, M). \end{aligned} \quad (4.1)$$

The definition domain of the algorithm is a closed interval of integers [1, M]. In the process of generating the sequence, when the numbers x_n, y_n left the interval [1, M], the transformation of refund $x_n \rightarrow x_n \pm M$ and $y_n \rightarrow y_n \pm M$ was applied.

The phase space (PS) of the algorithm has the dimension (Nz_1+Nz_2). The number of system states in this space for an algorithm certain on a bounded discrete set is finite and equal to $\mathbf{M^{(Nz1+Nz2)}}$. Since each system state is certain on a finite and limited numbers set and the explicit form of the algorithm is a unambiguous mapping, the system will sooner or later fall into the initial state and the process will become periodic. Until leaving for the

period, the generated sequence, as showed by the numerical experiment, is pseudo random. The appearance of a period in the sequence $\{\mathbf{x}_n\}$, as well as in the sequence $\{\mathbf{y}_n\}$, is realized in the case of simultaneous exact repetition of complete initial conditions sets from the delayed members $(\mathbf{x}_{n-1}, \ldots, \mathbf{x}_{n-Nz1})$ and $(\mathbf{y}_{n-1}, \ldots, \mathbf{y}_{n-Nz2})$.

The study of the structure PS of the algorithm was carried out in the accessible for numerical analysis range of parameters M, Nz_1, Nz_2: $\mathbf{M}^{(Nz1+Nz2)} \le 10^6 \div 10^7$. **Tables 1** and **2** show the results of studying the structure PS of the algorithm for odd (M = 3) and even (M = 4) values of the parameter M in comparison with the cycles spectra of the basic one-dimensional algorithm with the corresponding parameter values.

In Tables 1 and 2, the cycles number of same period is indicated in round brackets.

The PS of the algorithm under study consists of a set of cycles of different multiplicity and length and one special isolated point with coordinates (M, M, ..., M). It can be seen from Table 1 that for odd M all cycles have a single multiplicity, just like in the PS of the basic algorithm. At the same time, there is no obvious regularity between the cycles sizes in the PS of the compared algorithms. The size of the largest cycle is ~0.5 of the total number of states in the phase space $\mathbf{M}^{(Nz1 + Nz2)}$.

Table 1

Nz = 3	18, 8, 1	$\mathbf{M}^{Nz} = 27$
Nz = 4	44, 29, 7, 1	$\mathbf{M}^{Nz} = 81$
Nz = 5	118, 70, 22, 16, 13, 3, 1	$\mathbf{M}^{Nz} = 243$
Nz = 6	457, 100, 61, 31, 28, 26, 25, 1	$\mathbf{M}^{Nz} = 729$
$Nz_1 = 4$ $Nz_2 = 3$	1258, 351, 270, 88, 26, 1	$\mathbf{M}^{(Nz1+Nz2)} = 2187$
$Nz_1 = 5$ $Nz_2 = 3$	3614, 862, 798, 645, 496, 70, 16, 1	$\mathbf{M}^{(Nz1+Nz2)} = 6561$
$Nz_1 = 5$ $Nz_2 = 4$	8789, 5677, 2725, 1391, 613, 207, 39, 1	$\mathbf{M}^{(Nz1+Nz2)} = 19683$
$Nz_1 = 6$ $Nz_2 = 4$	24844, 23261, 5908, 2781, 400, 1	$\mathbf{M}^{(Nz1+Nz2)} = 59049$

Table 2

Nz = 3	14(4), 7, 1	$\mathbf{M}^{Nz} = 64$
Nz = 4	30(8), 15, 1	$\mathbf{M}^{Nz} = 256$
Nz = 5	42(22), 21, 14(4), 7, 6(2), 3, 1	$\mathbf{M}^{Nz} = 1024$
Nz = 6	126(32), 63, 1	$\mathbf{M}^{Nz} = 4096$
$Nz_1 = 4$ $Nz_2 = 3$	186(68), 93, 62(8), 31, 1	$\mathbf{M}^{(Nz1+Nz2)} = 16384$
$Nz_1 = 5$ $Nz_2 = 3$	60(544), 30(8), 15, 1	$\mathbf{M}^{(Nz1+Nz2)} = 65536$
$Nz_1 = 5$ $Nz_2 = 4$	465(412), 31(3), 1	$\mathbf{M}^{(Nz1+Nz2)} = 262144$
$Nz_1 = 6$ $Nz_2 = 4$	84(1149), 42(43), 21, 14(4), 1	$\mathbf{M}^{(Nz1+Nz2)} = 1048576$

For even values of M (Table 2), the cycles in the PS are usually short and multiple, as in the case of the basic algorithm. Cycles spectra of the two-dimensional algorithm with parameters Nz_1 and Nz_2 do not contain cycles of partial basic algorithms with $Nz = Nz_1$ and $Nz = Nz_2$, but have basic algorithm cycles with $Nz = (Nz_1 + Nz_2)/2$ with addition doubled period cycles. In this case, the main periods of cycles of the two-dimensional algorithm differ by an integer number of times from the fundamental period in each of the cycles series: for example, with $Nz_1 = 4$, $Nz_2 = 3$, the spectrum of cycles is 31 (fundamental period), 62, 93, 186. Such a character of the cycles spectrum is also characteristic of the one-dimensional algorithm for even values of M.

Table 1 shows that the cycles size of the greatest length of the two-dimensional algorithm is almost two orders of magnitude larger than the cycle size of the corresponding one-dimensional algorithm for $Nz_1 = Nz$. But this gain is due not so much to the specific two-dimensional mapping features compared to the one-dimensional analog, but to a real increase in the PS dimension. The ratio between the cycle length of the maximum size and the full number of states in the PS remains the same ~0.5.

The statistical characteristics evaluation should be carried out not at small, but at real, i.e. relatively large parameters values corresponding to developed chaos and the formation of long pseudo-random sequences with good correlation properties. Therefore, the calculations were performed with parameters M = 255, $Nz_1 = 16$,

$Nz_2 = 11$. It is shown that the two-dimensional algorithm generates a pseudo-random sequence with an almost uniform probability distribution $p(x) = 1/M$. For a sequence segment with $N = 210000$, the difference from this distribution is: relative average difference modulo $\Delta\mathbf{p}_{av} = 0.028$ at maximum $\Delta\mathbf{p}_{max} = 0.10$, rms $\boldsymbol{\sigma} = 0.002$.

The evaluation of the correlation characteristics of the generated sequences was carried out on the basis of the analysis of 100 pairs of unclipped and clipped segments of 128 and 1024 symbols, sequentially generated by the algorithm without any selection, including without selection by code balance. It was found that the emissions level of auto- and cross-correlation functions did not exceed following values: $(1.5\div4.8)/\sqrt{N}$ for segments with $N = 128$ and $(2.5\div4.9)/\sqrt{N}$ for segments with $N = 1024$, which consistent with the appropriate level side emissions of purely random sequences correlation functions with uniform distribution, and sequences generated by basic algorithm.

Counting blocks of identical symbols at the clipped sequence implementation of 270,000 numbers showed that the appearance probability of such blocks completely obeys the law $\mathbf{p(k) = 1/2^k}$ up to a block of size $\mathbf{k} = 12$ with insignificant differences from this law for blocks of $\mathbf{k} = 13\div18$ characters. The latter differences are due more to the data insufficiency for results statistical processing than the properties of the algorithms themselves.

The evaluation of the signals system volume generated by the two-dimensional and basic algorithms was estimated by selecting balanced codes with specified correlation properties from the generated clipped sequence. It is shown that for the same sequence implementation lengths, the selected codes number and rate of their selection are close for both compared algorithms.

The phase space structure of two-dimensional algorithm is analyzed. The periods spectrum of cyclic trajectories in phase space is found, which differ in initial conditions. It has been established that the pseudo-random sequences statistical properties generated by basic discrete algorithm and algorithm with a two-dimensional mapping are close with comparable parameters. However, the two-dimensional algorithm has an increased complexity, which greatly complicates its reconstruction based on the implementation of generated by the algorithm sequence.

5. METHODS OF CHAOTIC ALGORITHMS FRACTAL ANALYSIS

For the chaotic signals effective implementation in radio engineering complexes, telecommunication systems, as well as for their use as an information carrier in new generation information technologies, along with conventional methods for studying statistical and correlation characteristics, it is necessary to develop alternative estimate methods of algorithm structural complexity and PRS fractal dimension. [14].

The fractal analysis methods of random number generators currently include the determination of dynamic systems fractal dimensions, computer processing of both the these systems trajectory in phase space (PS), and formed by system of processes in projections on plane in the PS and on coordinate axes. In the latter case, we are talking about the study of the properties of the sequence directly generated by the algorithm.

For the fractal processing algorithms effective application, it is necessary to represent an algebraic object – a numbers sequence or signs in form of a graphic image. As geometric images that characterize the chaotic algorithms properties, you can choose a step-by-step mapping on plane of the recurrent sequence members (**Fig. 5.1**a), the two-dimensional section of chaotic algorithm multidimensional phase space (Fig. 5.1b), as well as the projection of the chaotic algorithm multidimensional phase space onto one of coordinate planes with or

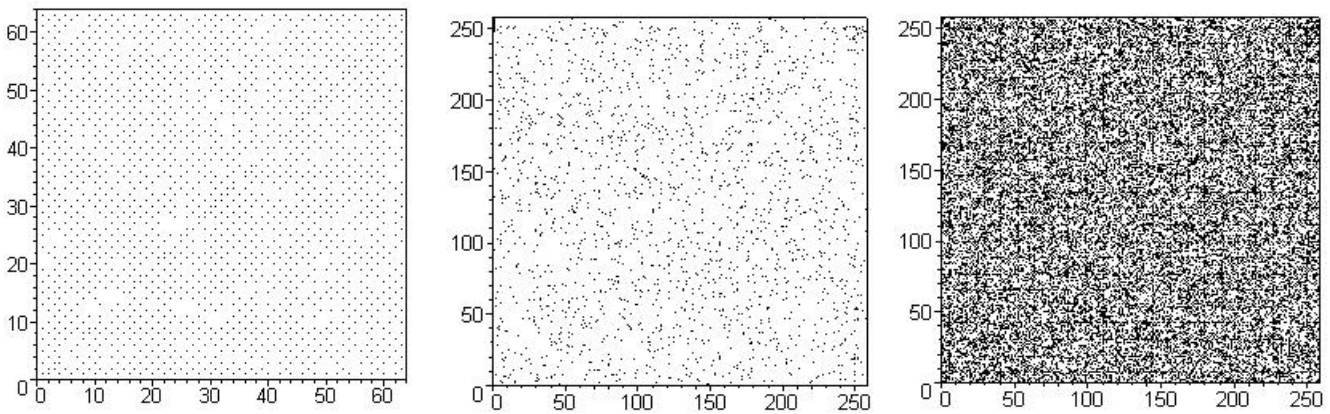


Fig. 5.1. *a) Mapping on the plane of members pairs of a recurrent chaotic sequence, b) Two-dimensional section of the multidimensional phase space of a chaotic algorithm, c) The FP projection of a chaotic algorithm with delay.*

without the multiplicity factoring each point of system state (Fig. 5.1c). An example of such a FP algorithm projection with parameters N = 30000 is shown (presented) in Fig. 5.1*c*.

The fractal characteristics were measured using the brightness field of the images. When measurements by two-dimensional field, two methods can be used. The first – the "sliding window" method – allows you to get the dependence $S = f(\delta)$, where S is measured parameter that determines the fractal signature, δ is smoothing window size. The second is a method for measuring of local dispersion dimension, which consists in measuring the brightness variance of a image small area on two scales. This method makes it possible to obtain a fractal dimensions spectrum by an image.[14].

For this purpose, we analyzed the simplest algorithms for generating of integers $\{x_n\}$ pseudo-random sequences with delay, using the Fibonacci mapping and its modifications:

$$Algorithm\ F-1 \quad \tilde{x}_n = x_{n-1} + (-1)^{x_{n-Kz}} x_{n-Nz} \quad 5(1)$$

$$Algorithm\ F-2 \quad \tilde{x}_n = x_{n-1} + (-1)^{x_{n-Nz}} x_{n-Nz} \quad 5(2)$$

$$Algorithm\ F-3 \quad \tilde{x}_n = x_{n-1} + x_{n-Nz} \quad 5(3)$$

where Nz and Kz are algorithms parameters, $2 \le Kz \le (Nz - 1)$. In contrast to [13], the sign in front of the retarded term in F-1 and F-2 does not change randomly independently, but is determined by the system internal dynamics. The feedback parameter Nz determines the phase space dimension of algorithm and, accordingly, the radius vector dimension $R_n(x_{n-1}, x_{n-2}, ..., x_{n-Nz})$ of the discrete dynamical system state at each step.

The phase space (PS) volume of the Fibonacci mapping of dimension Nz is practically unlimited. For the real application of PRS algorithms in radio engineering systems and the formation of modulating digital signals of a finite capacity, it is necessary to set the algorithm definition domain on a numbers finite set of a natural series closed interval [1, M], where $M > 1$. For this, mappings (5.1-5.3) must be supplemented by the operation of converting numerical interval [1, M] into itself, for example, of following form:

$$\begin{aligned} &x_n = \tilde{x}_n, && if \quad \tilde{x}_n \in [1, M], \\ &x_n = \tilde{x}_n - M, && if \quad \tilde{x}_n > M, \\ &x_n = \tilde{x}_n + M, && if \quad \tilde{x}_n < 1. \end{aligned} \quad (5.4)$$

This transformation, corresponding to segment contraction [1, M] into a ring, plays an important role in the chaotic behavior mechanism of these dynamical systems. This operation limits the phase space volume, making it finite, equal of $V_{PS} = M^{Nz}$ state points and provides additional trajectories mixing in the phase space. The operations of mapping the interval [1, M] into itself make the transformations of the algorithm ambiguous, which does not allow restoring the

formula and parameters of the algorithm by the clipped process well-known implementation.

It should be noted that one transformation of numer interval into itself is not enough for effective trajectories mixing in phase space. A randomization certain mechanism should already be contained in the mapping function. In this case, this is provided by the Fibonacci mapping properties. These two conditions – the phase space limited volume and the presence of a powerful mixing mechanism – are necessary conditions for any dynamic system chaotic behavior.

Algorithm F-4 based on the Fibonacci mapping (5.3) was also considered as an alternative, but with a different operation of converting the numerical interval [1, *M*] into itself, the type of reflecting boundary:

$$\begin{aligned} x_n &= \tilde{x}_n, && if \quad \tilde{x}_n \in [1, M], \\ x_n &= M, && if \quad \tilde{x}_n > 2 \cdot M, \\ x_n &= 2 \cdot M - \tilde{x}_n, && if \quad M < \tilde{x}_n < 2 \cdot M. \end{aligned} \tag{5.5}$$

Depending on the initial conditions choice, the radius vector Rn describes a trajectory in the algorithm phase space, which is successive discrete transitions from one point of the dynamic system (DS) state to another according to a random law. These motion "trajectories" of a discrete DS in the PS, due to the limited PS volume, form closed cycles, which, due to the transformations uniqueness, do not intersect and have no common points. In addition, cycle pools and isolated points can exist in the PS. The cycles of the studied algorithms F-1, F-2, F-3, F-4 have an important distinctive feature: the behavior of dynamic system before the cycle is closed (and also on the movement trajectory of the pool, if it exists) is chaotic, and generated by algorithm non-periodic sequence at the same time – pseudo-random type.

The such points set in the PS, united in a cycle, we called a pseudo-random cycle (PRC) if the non-periodic process formed by the algorithm before the cycle is closed is chaotic, in contrast to the regular cycle, which corresponds a regularic process before the DDS exits to a period. A pseudo-random cycle (until it closes) corresponds to an irregular motion in the phase space, and a regular cycle corresponds to a regular one. In both cases, the behavior of the dynamic system on the cycle is completely determined. The trajectory of a pseudo-random cycle is a deterministic set of points of discrete dynamical system states chaotically following one another in the entire volume of the algorithm phase space. An analogue of the pseudo-random cycle of a discrete system is the continuous dynamical system strange attractor [15].

The differences between pseudo-random and regular cycles are quite intuitive. Periodic motion is always regular, but regular motion is not necessarily periodic. Thus, PRC is such a discrete dynamical system (DDS) movement, which, at consideration intervals less than a period, is random chaotic, and at intervals greater than a period (more precisely, $N > 2T = 2Np$), the system behavior should be considered already as regular and periodic.

Depending on the values of the parameters Nz ≥ 3, Kz and M, there are a cycles whole row of different periods in the phase space of the F-1, F-2, F-3 algorithms. Each long ($N \sim V_{PS}$) cycle before its closure corresponds to a non-periodic PRS with an almost uniform distribution $p(x) \approx 1/M$ of generated numbers in a given interval of the domain $p(x) \approx 1/M$ and with uniform distributions of conditional probabilities. To characterize the chaotic set of points fractal properties on the PRC, we confine ourselves to an analysis of the geometric and correlation dimensions [16].

The fractal analysis methods can be applied, in principle, to any numerical set. In particular, in the discrete DS study, these methods can be directly applied to the points set of system states in an *n*-dimensional PS, and can also be applied to the points set of these states projections onto selected surfaces in the phase space. The

fractal analysis methods can also be successfully extended to phase space projections onto coordinate axes; in the latter case, we are dealing with the fractal properties study of sequences directly formed by discrete algorithms.

To characterize the fractal properties of a chaotic set of points on a pseudorandom cycle in an Nz-dimensional PS, we restrict ourselves to analyzing the Euclidean D and correlation dimensions D_2. Computer analysis was carried out for parameters small values of the chaotic algorithm with delay, which is of fundamental importance for estimating the PRC majority properties. With an increase in the algorithm dimension, the DDS behavior becomes much more complicated and the generated PRSs statistical characteristics improve.

An correlation dimension D_2 estimate of studied pseudo-random motion of a discrete dynamical system along a trajectory in a multidimensional PS can be given based on correlation integral $C(l)$ calculation given on the set of distances l between all pairs of DS state vectors on a cycle in PS, plotting the dependence $\lg C(l) = f(\lg(l))$ shown in **Fig. 5.2**, and determining angular coefficient of straight part at it.

Curve 1 in this figure corresponds to the algorithm with parameters Nz = 3, Kz = 2, M = 15, PRC with initial vector $R_0(1, 1, 1)$ and process

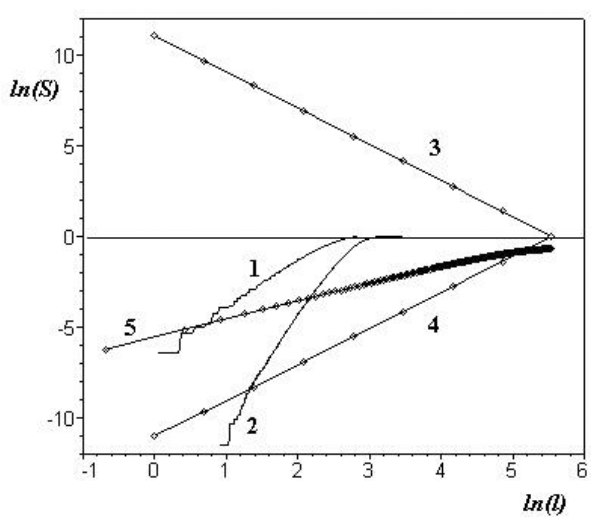


Fig. 5.2. *Fractal signatures for determining the Hausdorff and correlation dimensions.*

implementation length N = 630. By calculating the local angular coefficient, we can give the following correlation dimension estimate of the cycle under study: $D_2 \sim 2.4$. The obtained value is consistent with the Euclidean dimension D = 3, $D_2/D \sim 0.8$. The value of the latter ratio can serve as the degree characteristic of filling uniformity of the full PS volume with cycle points.

Curve 2 in Fig. 5.2 corresponds to the logarithm of correlation integral for the PRC with $R_0(1, 1, ..., 1)$ algorithm with parameters Nz = 7, Kz = 4, M = 15, N = 630. Graphs 1 and 2 of the function $\lg C(l) = f(\lg(l))$ in Fig. 5.2 are identical to each other, but have a extended straight sections different angular coefficient due to PS dimensions differences. For curve 2, the angular coefficient corresponds to the correlation dimension of the analyzed cycle $D_2 \sim 5.85$, $D = 7$, $D_2/D = 0.83$. Note that algorithm long cycles correspond to PRSs with good statistical and correlation properties, especially when the delay Nz increases more than 5.

An fractal characteristics analysis of the projection points of the DS states onto the two-dimensional coordinate plane (X_{-1}, X_{-2}), taking into account their multiplicity, was carried out by covering the numerical set with elementary cells with side l, calculating the required number of them $S(l)$ and then calculating the Hausdorff and correlation dimensions. Curve 3 in Fig. 5.2 corresponds to the dependency $\ln S(l) = f(\ln(l))$ of the PRC projection with the initial vector $R_0(2, 2, \ldots, 2)$ of the algorithm with parameters Nz = 16, Kz = 9, M = 255, the length of the process implementation N = 650000. D_0 = 2.0, D = 2, D_2/D = 1. Curve 4 in the figure was obtained by calculating the sums of squares logarithm of the observed occurrence frequencies of the DS state projections in unit cells covering the numerical set, which gives the following estimate of the PRC correlation dimension projection D_2 = 1.989, D = 2, D_2/D = 0.994.

The correlation dimension definition by the standard method, applied to a one-dimensional ($D = 1$) chaotic array of $N = 6500$ PRS numbers, formed by an algorithm with parameters Nz = 16, Kz = 9, $M = 255$ (curve 5 in Fig. 5.2) gave the correlation dimension value $D_2 = D_2/D = 0.988$, which indicates a fairly good filling uniformity interval [1, M] with generated numbers. This is confirmed by the analysis of the one-dimensional probability distribution of the numbers in the sequence.

On **Fig. 5.3** the computer calculation results of fractal signatures $\ln S = f(\ln a)$ are given, where S is the brightness characteristic of the graphic PS image of the chaotic algorithm, a is the window side (the measuring window is square, the relative window size varied from 3 to 30 pixels).

It can be seen from figure that all signatures have sections with different dominant slopes, which characterizes degree of statistical connection between corresponding members of recurrent chaotic sequence. Numerical analysis showed that the fractal signatures of mappings of algorithms with good mixing (weak statistical connection between members pairs of a recurrent chaotic sequence) are characterized by a smaller spread and almost the same slope.

Methods of computer analysis were used to study chaotic algorithms F-1, F-2, F-3, F-4 with delay with different characteristics (delay parameter, various mixing mechanisms). For the F-1 algorithm, the correlation dimension of the points set on cycle with initial vector $R_0(8,6,7,1)$ (curve 1) is equal to $D_2 = 3.3$. The obtained value agrees with the Euclidean dimension $D = 4$, $D_2/D = 0.83$. The latter ratio value can serve as a characteristic of the uniformity degree of the full volume filling of the PS with cycle points. As the analysis showed, the studied cycle with the initial vector $R_0(8,6,7,1)$ corresponds to a non-periodic PRS of length $N = 14030$ with a generated numbers distribution close to uniform.

The linear section of graph (curve 2), obtained for the points set of the trajectory basin and cycle in the F-2 algorithm phase space, has a somewhat smaller slope, which corresponds to the correlation dimension value about $D_2 = 3.0$. Curve 3 in Fig. 5.2 corresponds to logarithm of correlation integral for a pseudo-random cycle with initial conditions $R_0(1,6,6,7)$ of the tested algorithm F-3. Graphs 1 and 3 of the function $\log C(l) = f(\log(l))$ in **Fig. 5.4** almost exactly repeat each other and have an extended rectilinear section with a slope $D_2 = 3.3$, which and makes it possible to obtain a quantitative estimate of the filling uniformity of the space with DS states points on pseudo-random cycles. Note that PRSs with good statistical and correlation properties, especially when the delay increases more than 5 correspond to algorithms F-1 and F-3.

For the F-4 algorithm cycle with initial radius vector $R_0(7,14,6,15)$, the period $T = 613$, the

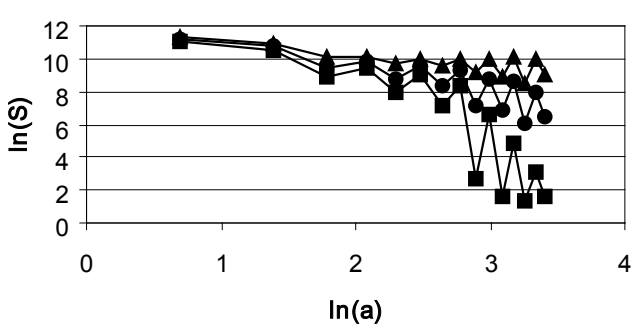


Fig. 5.3. *Fractal signatures of pairwise mappings: square – mapping* $(x_n; x_{n+1})$, *circle* – $(x_n; x_{n+2})$, *triangle* – $(x_n; x_{n+9})$.

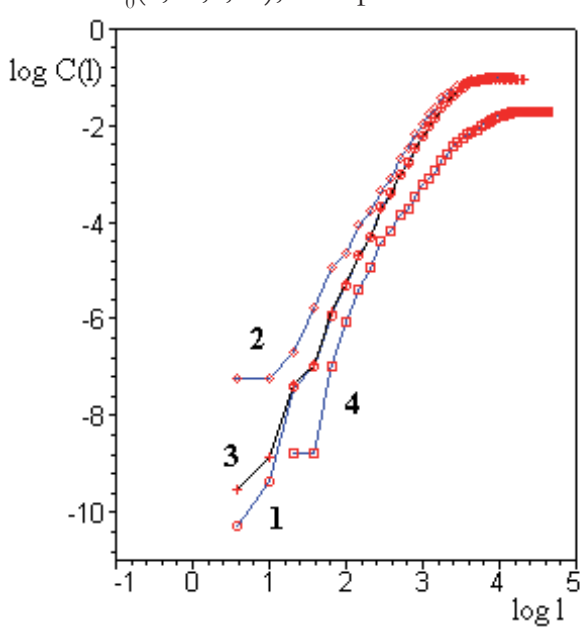


Fig. 5.4. *Dependence of* $\log_2 C(l)$ *on* $\log_2 l$ *for chaotic algorithms with delay and various delay parameters.*

dependence $\log C(l) = f(\log(l))$ (curve 4 in Fig. 5.2) does not have a clearly defined straight section. This means that correlation integral has significant deviations from law $C(l) \sim 1 - D$ and, therefore, this pseudo-random cycle points are located unevenly in PS.

Fractal analysis can be applied not only to a chaotic points set in a multidimensional PS, but also to a one-dimensional set of PRS implementation numbers. The correlation dimension definition by the standard method, applied to a one-dimensional (Euclidean dimension $D = 1$) chaotic array of $N = 1000$ PRS numbers, formed by the algorithms F-1, F-2, F-3, F-4 with different delay parameters, gave the following results. For all tested algorithms, the correlation dimension value is within $D_2 = D_2/D = 0.91 \div 0.96$, including for the random number generator RND of the Maple mathematical software package. The obtained ratio D_2/D values indicate a fairly good filling uniformity definition domain with the generated numbers. This is also confirmed by analysis of one-dimensional probability distribution of the numbers in the sequence.

It can be seen from given data that studied chaotic algorithms, as well as certified random number generator RND, demonstrate a sufficiently high structural quality of the generated sequences. When the distribution function of the generated numbers $p(x)$ and the correlation coefficient change, the proposed method for estimating degree of structural complexity effectively captures the corresponding change in the PRS statistical properties.

Thus, it is shown that calculation of fractal characteristics of recurrent chaotic sequences and their graphical images makes it possible to quantify effectiveness of mixing mechanism and degree of statistical connection between the chaotic sequence members, which ultimately determines complexity of the chaotic generating algorithm

6. STATISTICAL CHARACTERISTICS OF PSEUDO-RANDOM SIGNALS GENERATED BY DISCRETE ALGORITHMS WITH DELAY

The most complete information about statistical properties of discrete sequences is provided by the distributions analysis of pairwise conditional probabilities of the sequence members $p(i + j, x_n | i, x_k)$, $j = 1, 2, 3, \ldots; k, i = 1, 2, 3, \ldots, M$. The pairwise conditional probability is the generating probability number x_n at the $(i+j)$-th algorithm step, if the number x_k was obtained at the i-th step. In this case, the definition domain of the discrete algorithm is an arbitrary closed integer interval [M1, M2], M = M2 – M1 + 1, $x_n \in$ [M1, M2].

If conditional probabilities distribution for any j practically coincides with uniform distribution, $p(i+j, x_n | i, x_k) \approx 1/M$, $j = 1, 2, 3, \ldots$ with an arbitrary choice i. At the same time, if the probability distribution of generated numbers $p(x_n)$ is close to uniform, then value probability x_n is practically also equal to $1/M$. Thus, the transition probabilities to state x_n at the j-th step coincide with this value probability at this step, regardless of the sequence values at the previous algorithm steps, which is typical for random sequences in independent trials. More over, the pseudo-random sequence generated by such an algorithm will be close in its probabilistic characteristics to sequence of independent equiprobable numbers from the interval $[M_1, M_2]$ [17]. In the latter case, the given sequence can be expected to have best statistical properties. The establishment of such a fact emphasizes studying importance the conditional probabilities distributions for a priori judgments about generated pseudo-random sequences quality

To characterize the conditional distributions $p(x_{i+j} | x_i)$, the location form of points (x_{i+j}, x_i) on the plane for mapping $x_{i+j} = f(x_i)$ given by the discrete algorithm is of great importance for corresponding values $j = 1, 2, 3, \ldots$ and $i = 1, 2, 3, \ldots, N$ [18]. Obtaining the points scatter

(x_{i+j}, x_i) and visualizing them on screen does not require large computational resources compared to direct conditional probabilities calculation, although the this scatter nature does not directly give the shape of the conditional probability distribution. The scatter visualization indicates the regularity degree of these distributions, the functional relationships presence, the forbidden transitions existence, and even entire forbidden zones, which inevitably affects the correlation and other statistical properties of the sequence.

Consider the simplest formula of algorithm with delay: $x_n = x_{n-1} + x_{n-Nz}$, where Nz is delay parameter, supplemented by operation of returning to interval $[M_1, M_2]$ in case the newly obtained value x_n is thrown out of its limits. In this algorithm, mixing, chaotization of formed process is carried out by adding a random value of the sequence retarded member and by clipping resulting numbers sum on the boundary of definition domain M_2.

For numerical simulation, the following algorithm values parameters were taken: $M_1 = 1$, $M_2 = 255$, Nz = 16. The analyzed algorithm, with the selected parameter values and given initial conditions, which are a 16 random numbers set (delay vector), form a pseudo-random sequence with a probability distribution close to uniform distribution $p(x) = 1/M$. The difference from uniform distribution is characterized by total discrepancy between occurrence frequencies observed in numerical experiment in the generated sequence of numbers x_m and the value $1/M$:

$$\Delta p_\Sigma = \sum_{m=1}^{M} \left| n(x_m)/N - 1/M \right|,$$

where $n(x_m)$ is the numbers number xm in sequence of N members. This total discrepancy numerically coincides with relative average difference from uniform law $\Delta p_{av.rel.}$. The largest relative difference between the occurrence frequency of observed numbers and the value of $1/M$: $\Delta p_{max.rel.} = [1/(1/M)]\,|\,n(x_m)/N - 1/M\,|_{max}$.

Table 3

Algo-rithm	$p(x)$		$p(x_{i+j}\|x_i)$		
	N = 210 000		N = 52 000 000, k = 6		
	$\Delta p_{av.rel.}$	$\Delta p_{max.rel.}$	j	$\Delta p_{av.rel.}$	$\Delta p_{max.rel.}$
1	0.03	0.10	1	0.03	0.12
			16	0.031	0.12
2	0.03	0.10	1	0.997	1.12
			2	0.5	1.04
			16	0.03	0.10
RND	0.03	0.15	-	-	-

The obtained estimates of probability and conditional probability distributions for sequences generated by algorithms analyzed in this paper in comparison with standard RND generator are grouped in **Table 3**:

Two-dimensional distribution of points pairs (x_{i+j}, x_i), where $i = 1, 2, \ldots, N$ is a uniformly filled area of randomly scattered points, for any $j = 1, 2, 3, \ldots, 16, \ldots, 32$. Such nature of uniform, complete and random filling of numerical interval $[M_1, M_2]$ with points (x_{i+j}, x_i) indicates the process chaotization under study.

More accurate information about transition probabilities to each of the x_n values is given by the construction distributions conditional probabilities. For a visual representation in **Fig. 6.1** histograms of transition frequencies are constructed only for 6 generated numbers values $x_k = (k - 1)\cdot 50 + 1$, $k = 1, 2, ..., 6$ with a step $j = 1$. Frequencies histograms are calculated by implementing sequence from $5.2\cdot 10^7$ members. Values $\Delta p_{av.rel.}$ and $\Delta p_{max.rel.}$, characterizing

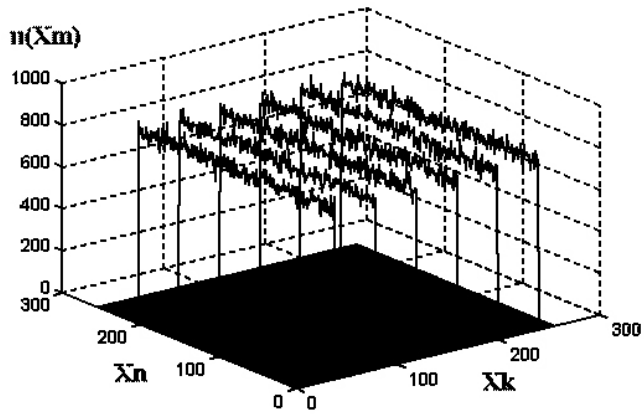


Fig. 6.1. *Transition frequency histograms for 6 values of generated numbers $x_k = (k - 1)\cdot 50 + 1$, $k = 1, 2, \ldots, 6$ with step $j = 1$.*

differences from uniform distribution of one of the histograms (x_k = 251) for j = 1 and 16, are given in Table 3. From the data obtained, it follows that for algorithm No. 1 of the distribution of conditional probabilities for any j practically coincide with the uniform distribution.

After a corresponding change in algorithm, in which one-dimensional distribution remains uniform, and distributions functions of conditional probability $p(x_{i+1}|x_i) = 0$ (j = 1) through one value depending on the parity of number x_i on previous step. The distribution pattern of points on the plane in this case is regular (**Fig. 6.2**). This is confirmed by quantitative differences in $\Delta p_{av.rel.}$ and $\Delta p_{max\ rel.}$ from a uniform distribution (see Table 3).

The sequences correlation characteristics generated by algorithms 1 and 2 were evaluated based on analysis of clipped and non-clipped 100 segments of size N = 128 and 1024 symbols, sequentially generated by algorithms without any selection. Despite significant differences in distributions form $p(x_{i+1}|x_i)$, lateral outliers value of auto- and cross-correlation functions relative to level $1/\sqrt{N}$ for both algorithms is approximately the same and is: (1.3÷3.8) for ACF N = 128, (1.5÷4.3) for CCF N = 128, (2.2÷4.8) for ACF and CCF N = 1024. These data indicate that the algorithms considered above features had little effect on correlation properties of both the sequences themselves and the results of their clipping. At the same time, a numerical

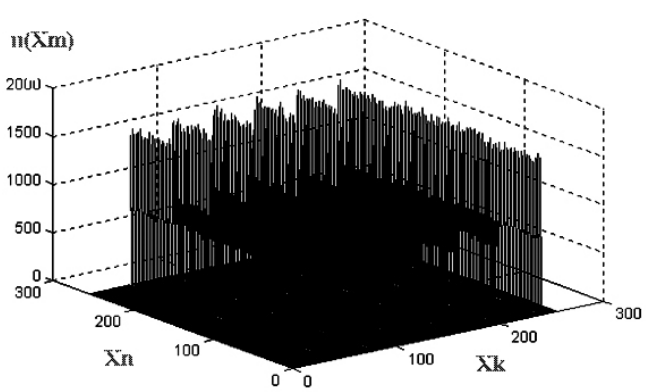


Fig. 6.2. *Distribution of conditional probabilities.*

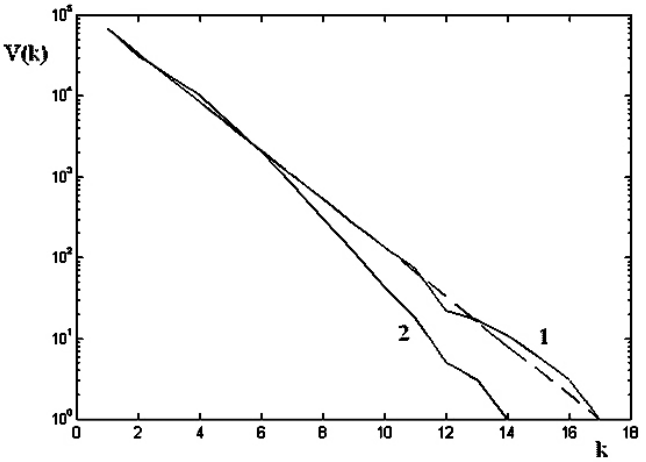


Fig. 6.3. *Block structure of sequences generated by algorithm No. 2.*

experiment on block structure over the length of a clipped sequence of $2.7 \cdot 10^5$ members showed that if algorithm No. 1 generates sequences with a block structure close to the law $p(k) = 1/2^k$ before blocks of size k = 17-18, then sequences block structure generated by algorithm No. 2 deviates significantly from this law (**Fig. 6.3**), which is directly related to uneven distribution of transition probabilities even at one step j = 1.

The volume of the signals system generated by algorithms was estimated by selecting from generated clipped sequence of balanced codes with a length of 128, 256 and 512 symbols with following correlation properties: lateral outliers of the aperiodic autocorrelation function do not exceed $R_{max} = 2.26/\sqrt{N}_{code}$, and the outliers of aperiodic crosscorrelation functions over entire array of selected codes are less than or equal to $R_{max} = 3.39/\sqrt{N}_{code}$. Correlation functions were calculated using the formula for balanced sequences [4].

As numerical experiment showed, the uneven conditional distributions and the forbidden transitions existence affected the select codes speed into signal system as sequence duration N increased (**Fig. 6.4**). In this figure, all curves corresponding to algorithm No. 2 are significantly lower than those of algorithm No. 1.

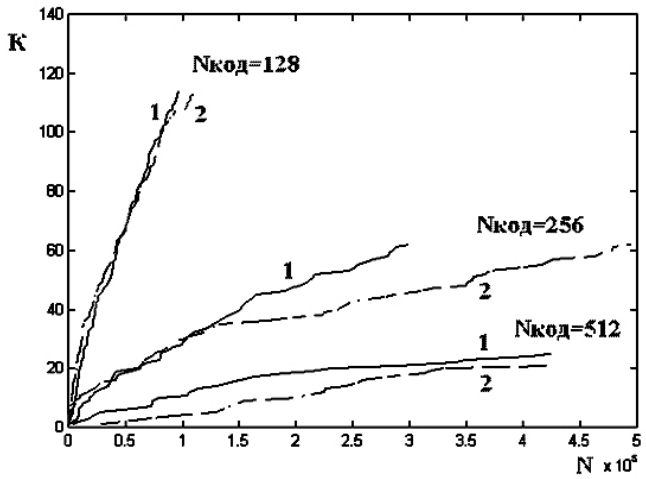


Fig. 6.4. *The number of balanced codes with lengths of 128, 256 and 512 characters.*

7. ANALYSIS METHOD FOR CODING PSEUDO-RANDOM ALGORITHMS BASED ON CODE GROUPS DISTRIBUTION

A promising direction in the modern radio communication systems development of is the broadband noise-like signals use [4,11]. Broadband communication systems have a advantages number over traditional narrowband systems. These advantages are associated with an increase in the users number, dense filling of the frequency range, and the need to ensure reliable confidential information transmission.

The spectrum expansion of transmitted signal, necessary for broadband systems operation, is achieved by using pseudo-random sequences generated by various special algorithms. These and other such sequences applications (redundant coding of information in digital communication channels, cryptography, Monte Carlo modeling) encourage search for new algorithms, including those with different characteristics and probabilities distribution functions. At the same time, search for new methods for statistical properties analyzing, the sequences they form, and estimating their proximity to an ideal random process continues [19].

A method for estimating statistical properties of digital multilevel pseudo-random sequences by statistics analyzing of code groups distribution in them from full code for selected pseudo-random signal base is considered. The this approach development is to find out intervals statistics between successive occurrences of identical code groups from the complete code. It is expected that this approach will allow, in particular, to analyze unknown algorithm structure that generates studied pseudo-random sequence. The specified intervals characteristics for sequences generated by various algorithms, including those used in standard software packages, are compared.

Statistical characteristics estimation of pseudo-random sequences generated by some algorithm and proximity to an ideal random sequence can be viewed as a forming process a steps series, the result of which at each step is completely independent of results at previous steps. Moreover, if the under study algorithm is defined on the integers set, for example, on the interval [1, M], where M-max is integer, then we can assume that ideal random sequence corresponding to these parameters will be formed as a result of throwing an M-dimensional die, on each face of which one of integers included in specified interval is plotted.

The approach allowing to analyze statistical structural features of pseudo-random integer sequences is considered. The investigated sequence is compared with a code group of a certain length and structure. The group length is sequence members number equal to chosen base of the pseudo-random signal (B), and group structure is a specific set of B integers from interval [1, M]. The total number of all different codes in this case is equal to elements number of the complete M^B code [20].

The described procedure makes it possible to construct the distribution functions of probabilities code groups coincidence coinciding with current implementation when repeatedly passing through the selected sequence section for code groups of different lengths and structures. Based on ideal random sequence model, one can obtain an closeness estimate of analyzed sequence to it for all relevant distribution functions. Such procedures have been performed for sequences generated by some algorithms. These sequences had indicated distributions, close to uniform, and differed little in variance from average variance for an ideal random sequence.

It has been established that for pseudo-random sequences generated by a number of algorithms (even close to an ideal random sequence), when analyzing intervals (K is interval length) between matches, when code group is shifted along pseudo-random sequence, some features are found. For the class of Fibonacci-type algorithms [5], close to uniform probability density distributions over all code groups for a given base were obtained.

However, when statistics analyzing of the certain code group occurrence (for $B = 1$), there was a gap in distribution $N_K = 0$, where N_K is the matches number. This interval (K) is equal to delay parameter, which is a analyzed algorithm characteristic for base $B = 1$. The distributions characteristic form of intervals (K) for 2 different code groups (11 and 19) (parameters of algorithm $M = 19$, delay Nz = 8) for $B = 1$ are shown in **Fig. 7.1***a,b*. For all other code groups, for a given B, this dependence has the same form as in Fig. 7.1*a*.

If the occurrence probability of a code group of length B for an ideal random sequence is $p = 1/M^B$, and T is members number of analyzed sequence, then occurrences average number of codes is T/M^B. Then expected matches number with a length B code group over an interval K in a length sequence T is:

$$N_K = (T / M^B) \cdot (1 - p)^{K-1} p. \quad (7.1)$$

On Fig. 7.1*a,b* this dependence $N_K = F(K)$ is presented for an ideal random process (curve 2). It has a character close to exponential.

For large signal bases ($B > 1$), the analyzed distribution of intervals has a more complex form: for interval K, which is equal to algorithm delay parameter Nz, there are code groups for which $N_K = 0$ and several codes for which N_K significantly exceeds expected average level corresponding to ideal random process model. Thus, the proposed method for statistics analyzing of intervals of appearance distribution of the of code groups makes it possible to decipher the structure of an unknown forming algorithm.

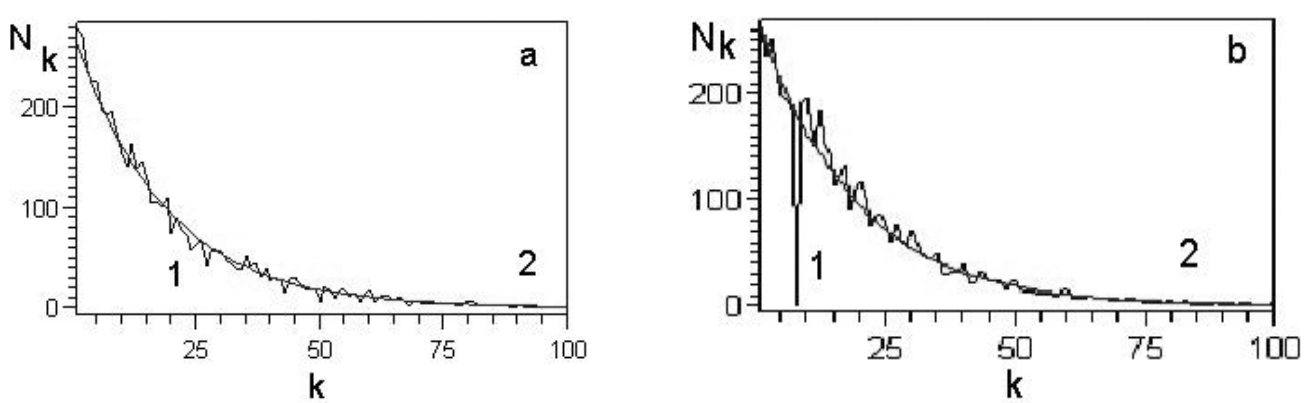


Fig. 7.1a,b. *Distribution over intervals between occurrences of identical code groups at B = 1 for a sequence generated by a Fibonacci-type algorithm (with parameters: M = 19, delay Nz = 8; a – curve 1 for code 11; b – curve 1 for code 19; curves 2 for a perfect random sequence.*

8. FILLING EFFICIENCY PHASE SPACE OF ENCODING DISCRETE ALGORITHM WITH DELAY

One of the promising ways to form a pseudo-random sequence of integers $\{x_n\}$ is algorithm with delay created on basis of processes modeling in ring self-oscillatory systems with dynamic chaos [13]. The algorithm discrete version is defined on integers set M of natural series from integer interval $[M_1, M_2]$ ($M_2 > M_1, M = M_2 - M_1 + 1$).

Along with M, the algorithm main parameter is delay parameter Nz, which determines number of delayed sequence members (x_{n-1}, x_{n-2}, x_{n-3}, ..., x_{n-Nz}) that must be known at each step to determine a new member x_n. The algorithm formula is supplemented with returning operation the number x_n to interval $[M_1, M_2]$ in case new value received is outside it.

The numbers x_{n-1}, x_{n-2}, x_{n-3}, …, x_{n-Nz} are generalized coordinates in Nz-dimensional phase space given dynamical system, and each specific set of them determines radius vector $R_n(x_{n-1}, x_{n-2}, \ldots, x_{n-Nz})$ and the corresponding system state point in this space. The total number of different delay vectors and system state points in phase space is equal to M^{Nz}. And each of these states can be accepted by system at least as initial conditions.

With algorithm parameters appropriate choice and initial conditions for changing state vector, i.e. transitions from one point of the phase space with coordinates R_i to another point with coordinates R_{i+1} are pseudorandom. But only as long as vector R_n takes on more and more new values. If the radius vector hits same point in the phase space again, due to the complete determinism of the algorithm, the system motion in phase space will repeat itself exactly, i.e. system enters a closed trajectory (cycle). This corresponds to occurrence of periodicity in sequence generated by algorithm.

Since, for given values of M and Nz, there are a finite number of different vectors in the phase space, sooner or later system will necessarily find itself on a cycle. The problem is to find longest realization N of pseudo-random sequence generated by algorithm that fills entire phase volume M^{Nz}.

The properties of algorithms study that generate pseudo-random sequences with a large period is an urgent task. One of most important parameters of such algorithms that characterize their quality is filling degree and structure of phase space.

Consider a discrete Fibonacci-type delay algorithm that generates a pseudo-random sequence with good statistical properties:

$$X_n = X_{n-1} + X_{n-Nz}, \quad X_n = X_n - M \text{ for } X_n > M. \tag{8.1}$$

The algorithm main parameter is delay parameter Nz, which determines number of memorized sequence members and phase space (PS) dimension. The algorithm is defined on a integers finite set of natural series from closed interval [1, M]. If newly calculated number of sequence is outside this interval, then a linear transformation of shift $\mathbf{x_n} \rightarrow \mathbf{x_n} \pm \mathbf{M}$ is performed, returning this number to the boundaries of definition domain. This transformation, in addition to functional action of Fibonacci-type algorithm itself, plays a significant role in chaotization mechanism of dynamic system under study behavior [7].

The number of system states points in phase space of algorithm is finite and equal to $\mathbf{M^{Nz}}$. The system movement in PS is carried out by jumping from one state to another. The system's motion trajectories occupy entire volume of PS, i.e. all possible states. Each such trajectory of system's motion follows its own closed cycle containing a system states limited number. The PS structure consists of a cycles finite set of different periods, the system behavior on which is pseudo-random. All cycles are located in a complex way in PS

entire volume. Thus, the PS of algorithm for Nz = 4 and $M = 17$ consists of five cycles with a periods of 73684, 3619, 2549, 2471, 529 and one singular point with coordinates (17, 17, 17, 17). The cycle choice is determined by setting a initial conditions set.

The system behavior pseudo-random nature on a cycle on an interval less than period is confirmed by the change dependence in distances in PS $\Delta R(n)$ between neighboring points on the cycle, shown in **Fig. 8.1** for algorithm with Nz = 3, $M = 13$. This distance at algorithm each step changes randomly, reaching values close to largest geometric PS dimensions.

The phase space under study of algorithm for Nz > 2 consists of one singular point with coordinates $(M, M, \ldots, M)$ and cycles family of different or same period. Each PS point belongs to only one specific cycle, while different cycles do not have a single common point.

The algorithm enters one or another cycle depending on the choice of the initial state vector. Until the cycle is closed, the state vector describes a pseudo-random process, by which corresponds to a non-periodic segment of a pseudo-random sequence generated by the algorithm of the appropriate size.

In the phase spaces ensemble of algorithms with different parameters M and

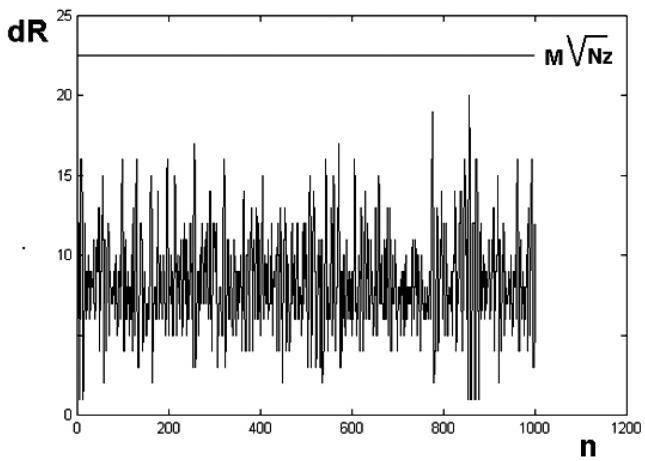


Fig. 8.1. *Distances in PS between neighboring points on cycle, Nz = 3, M = 13.*

Nz, both "short" cycles are observed, which periods T are much less than total number of points in phase space $\mathbf{M^{Nz}}$ ($T << \mathbf{M^{Nz}}$), and "long" cycles, whose periods are comparable to last value: $T \sim \mathbf{M^{Nz}}$. For even M, short cycles prevail in PS algorithm, and for odd M, short cycles do not exist at all, or they are presented in a small amount, occupying a small PS volume, which ensures existence of a long cycle. Thus, at odd M, longest cycles are observed. The such cycles period for certain values algorithm parameters can approach maximum possible value $\mathbf{T_{max}} = \mathbf{M^{Nz}}$.

A numerical experiment fixed case when PS contains only one long cycle and one isolated point: M = 2, Nz = 15, $\mathbf{T/M^{Nz}} = 1.0$. A close result was obtained at $M = 3$, Nz = 9, when period long cycle is $\mathbf{T/M^{Nz}} = 0.999$, and in addition to it and one isolated point, there is only one short fivecycle cycle in FS of system. All this confirms that value $\mathbf{T_{max}} = \mathbf{M^{Nz}}$ can serve as estimate of maximum non-periodic segment of sequence generated by algorithm. It should be borne in mind that this maximum period $\mathbf{T_{max}}$ can only be realized with certain ratios of the parameters M and Nz.

It is shown that by long cycles correspond to generated numbers distributions that are close to uniform. The change nature in the distribution functions of generated numbers with an increase in parameter Nz (for M = 255) is shown in **Fig. 8.2**. Here: $\Delta\mathbf{p_{av}}$, $\Delta\mathbf{p_{max}}$, $\boldsymbol{\sigma}$ – mean (1), maximum (2) relative modulo and root-mean-square (3) deviations of distributions from uniform (n = 210000). It can be seen that algorithm generates a sequence with an almost uniform distribution for Nz > 5. In this case, all conditional probabilities $\mathbf{p(x_i|x_j)}$ are also close to a uniform distribution. This means that pseudo-random sequence generated by this algorithm differs little in its probabilistic properties from sequence of independent equiprobable numbers from interval [1, M].

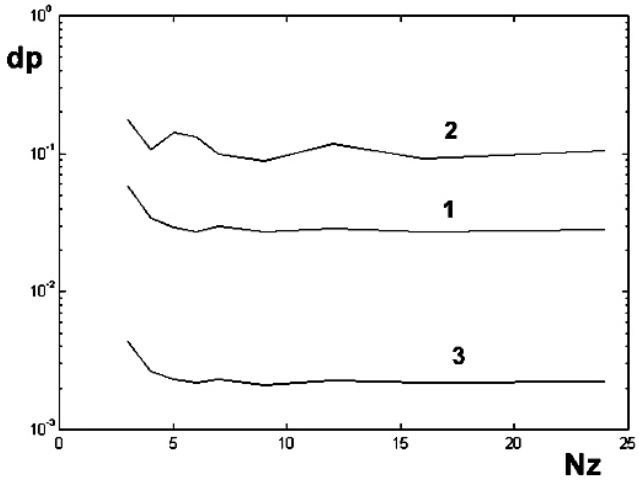


Fig. 8.2. *Difference of distribution from uniform depending on Nz (M = 255). 1 – Δp_{av}, 2 – Δp_{max}, 3 – σ.*

For large values of M and Nz, the structure study of system phase space by direct enumeration of its elements is very difficult. Therefore, study of system phase portrait was carried out at small values of phase space "volume", no more than $10^6 \div 10^7$. In this case, without limiting results obtained generality, in numerical analysis we will assume, as a rule, that $M_1 = 1$, $M_2 = M$.

The numerical study results of phase space structure for different values of *M* and Nz (*M* = 2÷21, Nz = 2÷18, $M^{Nz} \le 10^7$) showed that algorithm phase space is a finite cycles set, except for cases Nz = 2 for odd *M*, when in addition to cycles in phase space of system there are points that do not belong to any cycle, but belong to "pools" of these cycles. Thus, if system is at one of these points, then after a certain steps number a system will reach corresponding cycle. In this case, system trajectories in phase space are separate closed cycles, number of which and value of their periods depend on system parameters. As an example, we can give spectra of periods (i.e., cycles periods existing in phase space, and their number) for case of Nz = 4 (**Tables 4, 5**).

Table 4

M	Spectrum of periods
2	1, 15
4	1, 15, 30 (8)
8	1, 15, 30 (8), 60 (64)
16	1, 15, 30(8), 60(64), 120(512)
6	1, 15(3), 30(3), 80, 90(12)
12	1, 15, 30(72), 80, 90(192), 240(5)
18	1, 15(6), 30(21), 80, 90(105), 240(27), 270(324)
10	1, 15(5), 30(10), 150(6), 312(2)
20	1, 15, 30(93), 150(918), 312(2), 1560(6)
14	1, 3(2), 15(7), 30(21), 210(168), 342(7)

It can be seen from data presented that as $M = 2^k$ (k = 1, 2, 3, 4), $M = 6^k$ (k = 1, 2, 3), and $M = 10^k$ (k = 1, 2) increase, the cycles spectrum is supplemented by one a new value with a large period, and the spectrum always contains cycles of multiplicity 1, 15, and 30 (*M* > 2). At even *M* > 2 and Nz = 3, cycle spectra always exhibit periods 1, 7, and 14. At odd values of *M*, such simple regularities are not observed in structure of cycle spectrum.

The results obtained allow us to draw following conclusions.

1. The system's motion trajectory occupies entire phase space volume, i.e. all possible states, which total number is equal to MNz.

2. The algorithm phase space for Nz > 2 consists of one singular point with coordinates R(M, M, …, M) and a cycles family of different or same period. Each phase space point belongs to only one specific cycle, while different cycles do not have a single common point.

Table 5

M	Spectrum of periods
3	1, 7, 29, 44
5	1, 8, 27 (2), 562
7	1, 9, 22, 427, 653, 1289
9	1, 7, 10, 20, 22, 24, 29, 44, 75, 134, 296, 767, 5132
11	1, 21, 24, 41, 101, 173, 250, 14030
13	1, 626, 2992, 3712, 5056, 7977, 8197
15	1, 27, 44, 176, 562, 828, 1637, 4702, 7764, 11405, 11484, 11881
17	1, 529, 2471, 2549, 3619, 73684
19	1, 4182, 4219, 5067, 5408, 5916, 28778, 75061
21	1, 1289, 2833, 5228, 5401, 25900, 58208, 88633

3. The system enters one or another cycle, depending on which phase space point the initial state vector falls into. Until loop closes, the state vector describes a pseudo-random process, i.e. to cycles correspond to pseudo-random sequence segments of corresponding size.

4. In the phase spaces ensemble of algorithms with different parameters M and Nz, both "short" cycles are observed, which period T is much less than points total number in phase space M^{Nz} ($T << M^{Nz}$), and "long" cycles, which periods comparable with last value: $T \sim M^{Nz}$. For even M values, short cycles predominate in algorithm phase space, and for odd M, short cycles do not exist at all, or they are presented in a small amount, occupying a small phase space region, which and ensures existence of a long cycle. Thus, at odd M, the longest cycles are observed. The such cycles period for certain (previously unknown) values of algorithm parameters can approach maximum possible value $T_{max} = M^{Nz}$. On **Fig. 8.3** shows numerical experiment results to determine three largest cycle periods (curves 1, 2, 3) for different M values for the same delay parameter Nz = 3 in comparison with points total number in phase space – M^{Nz} (curve 4) and with estimate of maximum period $T(M) = M^{0.645Nz}$ at Nz = 3 (curve 5).

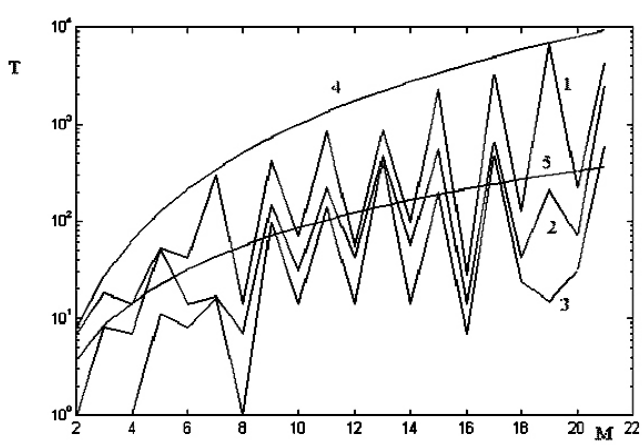


Fig. 8.3. *Cycle periods (curves 1, 2, 3) for different values of M with same delay parameter Nz = 3 in comparison with total number of phase space points – M^{Nz} (curve 4) and estimate of maximum period $T(M) = M^{0.645Nz}$ at Nz = 3 (curve 5).*

5. For an odd M value, all cycles, as a rule, have a different period, i.e. presented in singular. When M is even, same period cycles occur many times, although they are all different in terms of state vector accepted values.

6. For certain values of the algorithm parameters M and Nz, the period of a long cycle can be very close to the maximum possible value M^{Nz}: $T/M^{Nz} = 0.9 \div 1.0$. Moreover, experiment fixed case when phase space contains only one long cycle and one isolated point: M = 2, Nz = 15, $T/M^{Nz} = 1.0$. Almost the same result can be obtained at M = 3, Nz = 9: period of long cycle is $T/M^{Nz} = 0.999$, and in addition to it and one isolated point in system phase space there is only one 5-stroke short cycle.

All this testifies in favor of fact that value of $T_{max} = M^{Nz}$ can serve as an maximum period estimate of sequence generated by algorithm. It should be borne in mind that this maximum period T_{max} can only be realized with parameters certain ratios M and Nz. On Fig. 8.3 the boundary line $T(M) = M^{0.645Nz}$ (Nz = 3) is often exceeded not only by the longest period, but also by the periods of two more smaller cycles, therefore characteristic $T = M^{0.645Nz}$ should be considered as an average value for long cycles.

7. By long cycles correspond to distributions of generated numbers that are close to uniform. Thus, for a cycle with $T/M^{Nz} = 0.895$ (M = 13, Nz = 5), relative average difference on modulus from uniform distribution is 0.2% with a relative maximum of 0.45%, and root-mean-square deviation is 0.07%. It is these long cycles that can be used to generate pseudo-random sequences of long duration with a uniform probabilities distribution of

generated numbers. Based on monotonicity of the dependences considered, the results obtained can be considered valid for significantly large volumes and phase space dimensions

9. CONCLUSION

With an appropriate choice of parameters, the developed discrete algorithms form long non-periodic segments of pseudo-random sequences with a uniform probability distribution, which can be effectively used in cryptography, as well as when encoding information in telecommunication systems and computer networks [21].

REFERENCES

1. Knuth DE. *The art of computer programming. Volume 2: Seminumerical Algorithms.* Third edition, Addison–Wesley, 2007, 832 p.
2. Petrov AA. *Komp'yuternaya bezopasnost'. Kriptograficheskie metody zashchity* [Computer security. Cryptographic methods of protection]. Moscow, DMK Publ., 2000, 445 p.
3. Kuznetsov SP. *Dinamichesky khaos. Kurs lektsiy* [Dynamic chaos. Lecture course]. Moscow, Fizmatlit Publ., 2001, 295 p.
4. Varakin LE. *Sistemy svyazi s shumopodobnymi signalami* [Communication systems with noise-like signals]. Moscow, Radio i svyaz' Publ., 1985.
5. Bykov VV. *Tsifrovoe modelirovanie v statisticheskoy radiotekhnike* [Digital modeling in statistical radio engineering]. Moscow, Sovetskoe radio Publ., 1971, 328 p.
6. Hayes Brian. The Vibonacci Numbers. American Scientist: Computing Science. July-August 1999.
7. Shuster G. *Deterministic chaos. Introduction.* Moscow, Mir Publ.,1988, 240 p.
8. Bulinsky AV, Shiryaev AN. *Teoriya sluchaynykh prozessov* [Theory of random processes]. Moscow, Fizmatlit Publ., 2005.
9. Kemeny JG, Snell JL. *Finite Markov chains. The University Series in Undergraduate Mathematics.* Princeton, Van Nostrand, 1960.
10. Bharucha-Reid AT. *Elements of the Theory of Markov Processes and Their Applications.* New York, McGraw-Hill, 1960.
11. Rytov SM. *Vvedenie v statisticheskuyu radiofiziku* [Introduction to statistical radiophysics]. Moscow, Nauka Publ., 1966.
12. Gantmakher FR. *Teoriya matrits* [Matrix theory]. Moscow, Nauka Publ., 1966.
13. Belyaev RV, Vorontsov GM, Kolesov VV. Sluchaynye posledovatelnosti, formiruemye nelineynym algoritmom s zapazdyvaniem [Random sequences generated by a non-linear algorithm with delay]. *Radiotekhnika i electronika*, 2000, 45(12):954-960.
14. Potapov AA. *Fraktaly v radiofizike i radiolokatsii* [Fractals in radiophysics and radoilocation]. Moscow Logos Publ., 2002.
15. Malinetsky GG. *Khaos. Struktury. Komp'yuterny eksperiment. Vvedebie d nelineynuyu dinamiku* [Chaos. Structures. Computer experiment. Introduction to nonlinear dynamics]. Moscow, Editorial URSS Publ., 2002.
16. Ruell D, Takens D. *Comm. Math. Phys.*, 1971, 20(3):167.
17. Venttsel' ES. *Teoriya sluchaynykh protsessov i yeye inzhenernyye prilozheniya* [Theory of random processes and its engineering applications]. Moscow, Vysshaya shkola Publ., 2007, 479 p.
18. Kahaner D, Moler C, Nesh S. *Numerical Methods and Software.* Prentice-Hall, Inc. A Divisioon of Simon & Shuster Englewood Cliffs. NJ, 1989.
19. Bykov VV. Tsifrovoye modelirovaniye v statisticheskoy radiotekhnike [Digital modeling in statistical radio engineering]. Moscow, Sovetskoye radio Publ., 1971, 328 p.
20. Golenko DI. Modelirovaniye i statisticheskiy analiz psevdosluchaynykh chisel na elektronnykh vychislitel'nykh mashinakh [Modeling and statistical analysis of

pseudo-random numbers on electronic computers]. Moscow, Nauka Publ., 1965, 227 p.
21. Petrov AA. Komp'yuternaya bezopasnost'. Kriptograficheskiye metody zashchity [Computer security. Cryptographic methods of protection]. Moscow, DMK Publ., 2000, 445 p.