УДК 681.586
DOI: 10.18469/ikt.2024.22.3.11

ГЕНЕРАЦИЯ ТРЕХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ СЛУЧАЙНЫХ ЧИСЕЛ ДЛЯ КРИПТОГРАФИЧЕСКИХ СИСТЕМ ЗАЩИТЫ ИНФОРМАЦИИ

Шакурский М.В., Козырева Н.И., Макаров И.С.
Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ
E-mail: m.shakurskiy@psuti.ru, n.kozyreva@psuti.ru, i.makarov@psuti.ru

Статья посвящена разработке генератора псевдослучайных чисел. Известны решения, использующие в своих структурах обратную связь. Такие решения подходят для криптографии, так как обеспечивается высокая вычислительная сложность поиска предыдущих значений случайной величины. Тем не менее, использование обратной связи в структурной схеме алгоритма приводит к возникновению стохастической связи между соседними значениями генерируемой последовательности. В основе предлагаемого генератора лежит идея использования трехканальной системы с взаимной связью, где каждый канал представляет собой генератор, формирующий выходной сигнал на основе выходных сигналов двух других генераторов. Это позволило получить три независимые последовательности псевдослучайных чисел с существенно ослабленной связью между предыдущим и последующим значениями. За счет такого подхода значительно увеличивается период повторения генерируемых чисел, так как для повторения генерации требуется одновременное появление на выходах трех генераторов начальных значений. Вероятность такого события уменьшается на много порядков. Генераторы работают параллельно, поэтому быстродействие системы не снижается.

Ключевые слова: *генератор псевдослучайных чисел, взаимная связь, остаток от деления, период повторения, структурная реализация, программная реализация*

Введение

Генерация случайных сигналов представляет собой достаточно востребованное направление в различных областях научных исследований и техники [1; 2; 3]. В области криптографической защиты информации значительное внимание уделяется генераторам псевдослучайных чисел (ГПСЧ), в основе которых лежит алгоритм, реализующий математическую модель генерации [4–7]. Известные подходы имеют ряд недостатков, касающихся статистических характеристик генерируемой последовательности и ее длительности. То есть, использование ГПСЧ в качестве гамма-последовательности является достаточно уязвимым решением, так как по небольшому отрезку известных значений может быть восстановлен весь ряд случайных чисел [8]. Таким образом, актуальной является задача разработки такого генератора случайных чисел, восстановление полного ряда которого по известному ряду значений невозможно или крайне затруднительно.

Задача генерации случайных чисел

В общем случае ГПСЧ представляет собой функцию, входом которой является набор параметров, определяющих генерируемый ряд. Рассмотрим алгоритм Блюм-Блюма-Шуба [9], где для генерации чисел используется функция с обратной связью. Входным значением является результат решения функции на предыдущей итерации. В данном алгоритме повторение генерируемого ряда произойдет тогда, когда в результате работы генератора повторится начальное значение ряда. При этом восстановление предыдущего значения итерации осложняется использованием функции остатка от деления нацело, что требует решение задачи факторизации. Таким образом, данный алгоритм обладает хорошими криптографическими свойствами и большим периодом повторения. Одно из возможных решений по увеличению периода повторения описано в работе [10], где предлагается формирование двух последовательностей случайных чисел, где каждый из генераторов использует на входе результаты работы обеих генераторов на предыдущей итерации. К достоинствам предложенного решения следует отнести меньшую вероятность повторения ряда, так как теперь, для повторения ряда чисел требуется совпадение начальных значений двух генераторов и возможность генерации двух последовательностей случайных чисел. Однако данное решение не обеспечивает устойчивость к потенциальному решению задачи факторизации и не исключает взаимную стохастическую связь значений между соседними значениями, так как результат работы функции на одной итерации является аргументом функции на следующей итерации.

Генерация трех последовательностей случайных чисел

В целях совершенствования существующих решений предлагается реализация генератора трех независимых последовательностей незави-

симых случайных чисел. Такое решение позволит снизить взаимную стохастическую связь между генерируемыми значениями и усложнить вычисление значений предыдущей итерации.

Рассмотрим математическую модель генератора трех последовательностей случайных чисел. Для этого построим выражения, описывающие генерацию каждого из рядов случайных чисел, являющихся функциями двух, сгенерированных на прошлой итерации значений:

$$\left.\begin{aligned} x_{i+1} &= f(y_i, z_i) \\ y_{i+1} &= f(x_i, z_i) \\ z_{i+1} &= f(x_i, y_i) \end{aligned}\right\}. \qquad (1)$$

Заметим, что функция генерации каждого из значений имеет аргументом результат генерации двух других функций, что значительно снижает стохастическую связь между соседними генерируемыми значениями в каждом ряду генерируемых значений. Построим функции генерации:

$$\left.\begin{aligned} x_{i+1} &= \{a_1 y_i + b_1 z_i + c_1 y_i / z_i\} \\ y_{i+1} &= \{a_2 z_i + b_2 x_i + c_2 z_i / x_i\} \\ z_{i+1} &= \{a_3 x_i + b_3 y_i + c_3 x_i / y_i\} \end{aligned}\right\}. \qquad (2)$$

Коэффициенты *a, b,* с уравнений (2) подбираются с целью получения оптимальных статистических характеристик генерируемых последовательностей. Начальные значения *x*1, *y*1, *z*1 являются ключом. Изменение начальных значений приводит к генерации новых последовательностей.

Сформируем на основе выражений (1) структурную схему ГПСЧ (рисунок 1). Структурная схема содержит три идентичных функциональных блока 1, 2 и 3 для генерации значений *x, y* и *z* соответственно, с тремя входами и двумя выходами. На первые входы поступают начальные значения (значения ключей). Вторые и третьи входы соединены с первыми выходами двух других блоков.

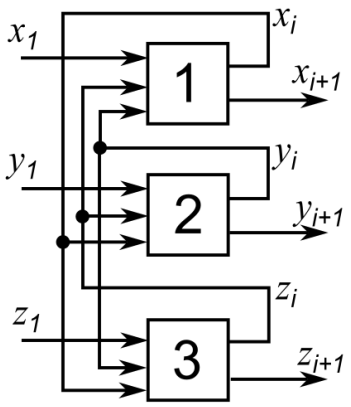

Рисунок 1. Структурная схема ГПСЧ

При этом структура каждого из трех блоков (рисунок 1) имеет одинаковый вид, приведенный на рисунке 2. Заметим, что обозначения входов и выходов на рисунке 2 даны для первого функционального блока, однако, изменение блока приведет лишь к изменению входных значений в соответствии с рисунком 1.

Каждый функциональный блок (рисунок 1) содержит блок 1 памяти коэффициентов (БП) *a, b,* с (рисунок 2), блоки 2, 3, 4 умножения, к первым входам которых подключены выходы блока 4, а ко вторым входам подключены входы функционального блока, блок 5 деления, к первому входу которого подключен второй вход функционального

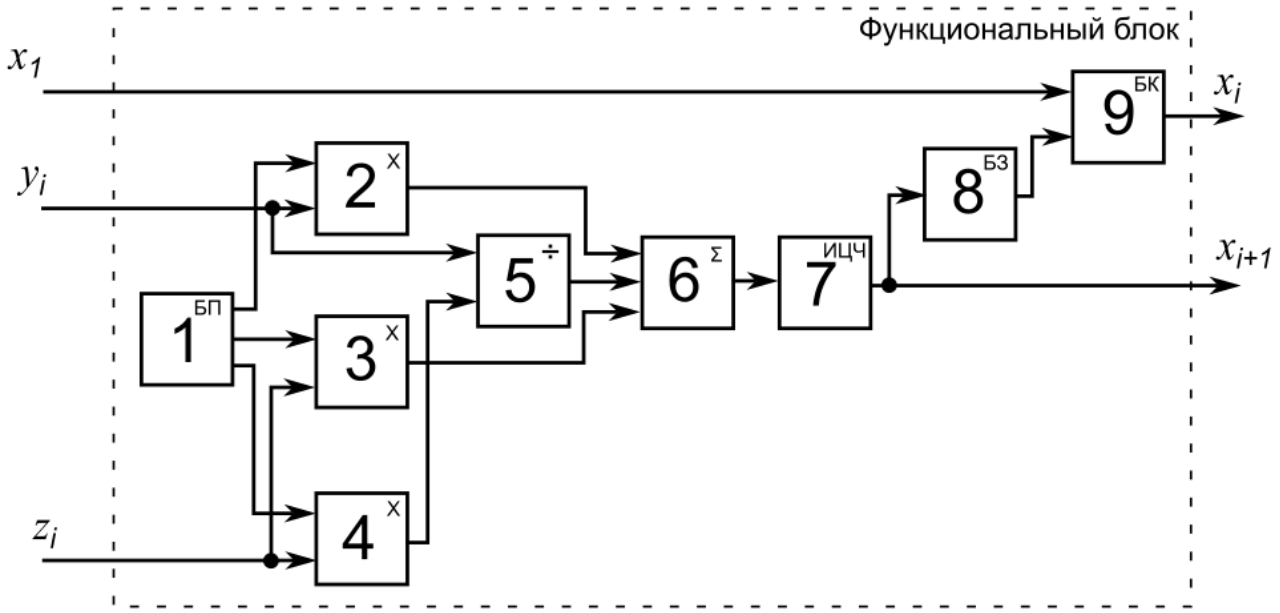

Рисунок 2. Структурная схема блоков ГПСЧ

блока, а ко второму входу подключен выход блока 4, блок 6 суммирования, ко входам которого подключены выходы блоков 2, 3, 5, а выход которого подключен к входу блока 7 с исключением целой части (ИЦЧ), выход блока 7 подключен ко входу блока 8 задержки (БЗ) и является вторым выходом функционального блока, блок 9 коммутации, первый вход которого подключен к первому входу функционального блока, второй вход подключен к выходу блока 8, а выход является первым выходом функционального блока. Устройство работает следующим образом. Каждый из трех функциональных блоков работает идентично. На первом шаге блоки 9 коммутации передают на первые выходы функциональных блоков начальные значения (значения ключей) с первых входов, которые по обратной связи передаются на соответствующие входы функциональных блоков. Начиная со второго шага работы устройства, блоки 9 коммутации передают на первые выходы функциональных блоков значения с блоков 8 задержки на один отсчет. Работа функционального блока основана на выражениях (2). Умножители 2, 3 и 4 формируют три слагаемых, причем значения коэффициентов a, b и c берутся, соответственно, с первого, второго и третьего выходов блока 1 памяти. Блок 5 реализует деление в третьем слагаемом, блок 6 осуществляет сложение трех слагаемых. Блок 7 исключает целую составляющую из полученного в блоке 6 значения, после чего результат передается на второй выход функционального блока и на блок 8 задержки, для последующего использования результата на следующем шаге.

Выходной сигнал, каждого из трех функциональных блоков представляет собой последовательность случайных чисел, определяемую коэффициентами a, b и c и начальными значениями (ключами).

Компьютерное моделирование

Моделирование генератора выполнено на языке программирования Python. Определим массивы рядов псевдослучайных чисел, задавшись произвольными начальными значениями. Листинг кода:

```
a1=0.65; a2=0.5; a3=0.55
b1=0.6; b2=0.55; b3=0.7
c1=4; c2=3.6; c3=4.3
n=100000
x=np.zeros([n],dtype=float)
y=np.zeros([n],dtype=float)
z=np.zeros([n],dtype=float)
x[0]=0.6; y[0]=0.55; z[0]=0.65
```

Величина n определяет количество чисел в массиве каждого ряда.

Формирование цикла вычисления случайных чисел в соответствии с выражениями реализуется следующим кодом:

```
for i in range(n-1):
    x[i+1]=(a1*y[i]+b1*z[i]+c1*y[i]/z[i])%1
    y[i+1]=(a2*x[i]+b2*z[i]+c2*z[i]/x[i])%1
    z[i+1]=(a3*x[i]+b3*y[i]+c3*x[i]/y[i])%1
```

В результате получим три массива случайных чисел. Визуализация полученных чисел представлена на рисунке 3 в виде точечных графиков. Жирными точками показаны значения массива x, маленькими точками значения двух других массивов. Результат приведен для первых 1000 отсчетов каждого массива.

Проведем анализ полученных значений. Проанализируем распределение значений. Для этого построим гистограмму значений, поделив все значения на две «корзины». Результат для трех каналов приведен на рисунке 4. Приведены распределения значений по двум корзинам для трех массивов значений.

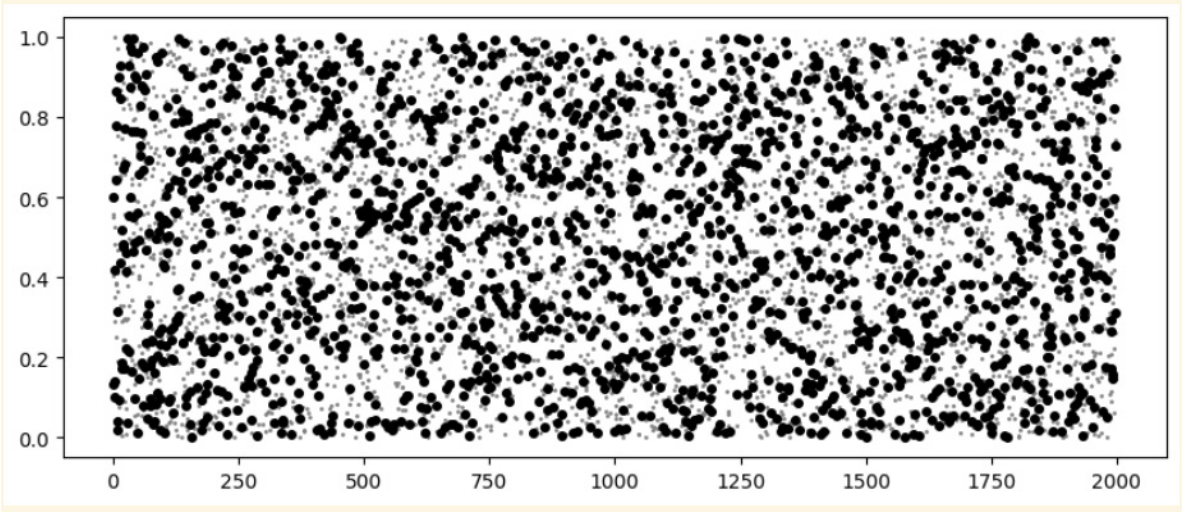

Рисунок 3. Визуализация полученных значений случайных чисел

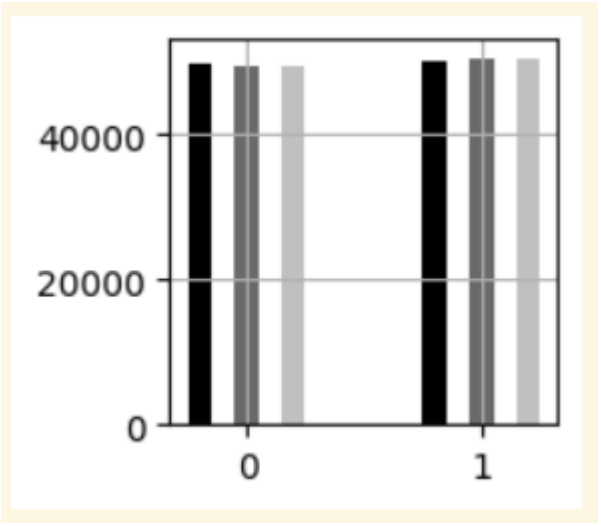

Рисунок 4. Распределение значений по двум «корзинам»

На рисунке 5 приведено распределение значений по десяти «корзинам», то есть 0..0,1, 0,1..0,2 и т.д. Видно, что значения равномерно распределяются во всем диапазоне возможных значений. Эксперименты с большим количеством корзин привели к аналогичному результату – к равномерному распределению значений.

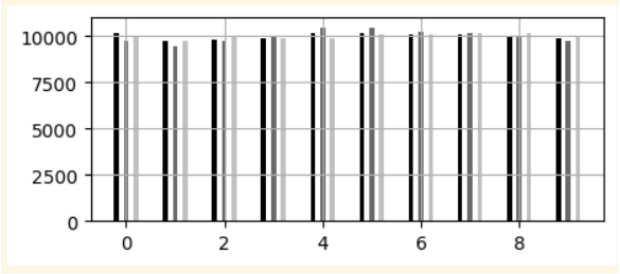

Рисунок 5. Распределение значений по десяти «корзинам»

Таким образом, генератор дает значения, равномерно распределенные во всем диапазоне от нуля до единицы.

Рассчитаем статистические характеристики полученных массивов.

Математическое ожидание по выходам:

x: *0,5009276269407209*

y: *0,503412071585185*

z: *0,5026476352807453*

Дисперсия по каналам по выходам:

x: *0,08294137402589338*

y: *0,08162625720105295*

z: *0,08320368092229136*

Стандартное отклонение:

x: *0,28799544098109153*

y: *0,2857030927397408*

z: *0,2884504826175393*

Проанализируем статистическую связь между массивами случайных значений. Для этого найдем ковариацию и коэффициент корреляции массивов попарно.

Ковариация:

x,*y*: *-0,00012238879458405428*

y,*z*: *-0,00013971923270529853*

z,*x*:- *0,00022766764314654492*

Коэффициенты корреляции:

x,*y*: *-0,001487430765140954*

y,*z*: *-0,0016953744638278099*

z,*x*:- *0,002740564866956251*

Проведем анализ в пределах каждого массива Сравним массив с самим собой, смещенным на один отсчет для установки стохастической связи между соседними отсчетами. В каждом генераторе на вход поступают зачения с двух других генераторов, что должно обеспечить отсутствие стохастической связи между соседними значениями. Для этого получим значения ковариации, сравнивая каждый массив с самим собой, смещенным на 1 отсчет. Ковариация в каналах при смещении:

x: -0,00044465370813492156

y: -0,00013004483682494856

z: -0,00023188457220133324

Видно, что связь между соседними значениями отсутствует.

Выводы

Исследование разработанного генератора трех последовательностей случайных чисел показало, что предлагаемое решение обладает рядом достоинств, а именно:

Генерируются три независимые последовательности независимых случайных чисел, определяемые коэффициентами алгоритма и начальными значениями, что может быть полезно для решения задач криптографического шифроания и пространственного моделирования.

Три массива генерируемых значений имеют равномерное распределение в диапазоне больше нуля и меньше единицы.

Возможно объединение массивов необходимым образом.

Литература

1. Сергиенко А.Б. Цифровая обработка сигналов. СПб.: БХВ-Питербург, 2011. 768 с.
2. Моделирование и анализ системы массового обслуживания общего вида с произвольными распределениями временных параметров системы / М.А. Буранова [и др.] // Инфокоммуникационные технологии. 2015. Т. 13, № 3. С. 252–258.

3. Kartashevsky V., Kozyreva N., Makarov I. The numerical method for analysis of arbitrary type queuing systems application // 2016 3rd International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S and T 2016). 2016. P. 109–111.
4. Barker E., Kelsey J. NIST Special Publication 800-90A. Revision 1. Recommendation for Random Number Generation Using Deterministic Random Bit Generators. National Institute of Standards and Technology, 2015. 110 p. URL: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-90ar1.pdf (дата обращения: 01.10.2024).
5. ГОСТ Р ИСО 28640-2012. Статистические методы: Генерация случайных чисел. М.: Стандартинформ, 2014. 36 с.
6. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications / A. Rukhin [et al.]. National Institute of Standards and Technology, 2010. 131 p. URL: https://math.uni.wroc.pl/~lorek/teaching/files/nistspecialpublication800-22r1a.pdf (дата обращения: 01.10.2024).
7. L'Ecuyer P., Simard R.J. TestU01: A C library for empirical testing of random number generators // ACM Transactions on Mathematical Software (TOMS). 2007. Vol. 33, no. 4. DOI: 10.1145/1268776.1268777
8. Молдовян Н.А., Молдовян А.А., Еремеев М.А. Криптография. От примитивов к синтезу алгоритмов. СПб.: БХВ-Петербург, 2004. 448 с.
9. Blum L., Blum M., Shub M. A simple unpredictable pseudo-random number generator // SIAM Journal on Computing. 1986. Vol. 15. P. 364–383.
10. Способ и устройство генерации последовательностей случайных чисел: патент 2716357. Российская Федерация. № 2018133894 / М.В. Шакурский (RU); заявл. 25.09.2018; опубл. 11.03.2020, бюл. № 8.

Дата поступления 11.11.2024
Дата принятия 11.12.2024

Шакурский Максим Викторович, д.т.н., заведующий кафедрой информационной безопасности (ИБ) Поволжского государственного университета телекоммуникаций и информатики (ПГУТИ). 443090, Российская Федерация, г. Самара, Московское шоссе, 77. Тел. +7 927 772-98-73. E-mail: m.shakurskiy@psuti.ru

Козырева Надежда Ивановна, к.т.н., доцент кафедры ИБ ПГУТИ. 443090, Российская Федерация, г. Самара, Московское шоссе, 77. Тел. +7 927 208-31-99. E-mail: n.kozyreva@psuti.ru

Макаров Игорь Сергеевич, к.т.н., заведующий кафедрой программной инженерии ПГУТИ. 443090, Российская Федерация, г. Самара, Московское шоссе, 77. Тел. +7 937 208-80-66. E-mail: i.makarov@psuti.ru

UDC 681.586
DOI: 10.18469/ikt.2024.22.3.11

GENERATION OF THREE RANDOM NUMBERS SEQUENCES FOR CRYPTOGRAPHIC INFORMATION SECURITY SYSTEMS

Shakurskiy M.V., Kozyreva N.I., Makarov I.S.
Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation
E-mail: m.shakurskiy@psuti.ru, n.kozyreva@psuti.ru, i.makarov@psuti.ru

The article is devoted to the development of a pseudorandom number generator. Somesolutions that use feedback in their structures are provided. Such solutions are suitable for cryptography, since high computational complexity of searching for previous values of a random variable is ensured. However, the use of feedback in the structural diagram of the algorithm leads to the emergence of a stochastic relationship between adjacent values of the generated sequence. The proposed generator is based on the idea of using a three-channel system with mutual communication function, where each channel is a generator that forms an output signal based on the output signals of two other generators. This made possible to obtain three independent sequences of pseudorandom numbers with a significantly weakened connection between the previous and subsequent values. Due to this approach, the repetition period of the generated

numbers significantly increases, since the repetition of generation requires simultaneous appearance of initial values at the outputs of three generators. The probability of such an event decreases by many orders of magnitude. Generators operate in parallel, so the system performance does not decrease.

Keywords: *pseudo-random number generator, interconnection, remainder of division, repetition period, structural implementation, software implementation*

Shakurskiy Maxim Viktorovich, Povolzhskiy State University of Telecommunications and Informatics, 77, Moskovskoe shosse, Samara, 443090, Russian Federation; Head of Information Security Department, Doctor of Technical Sciences. Tel. +7 927 772-98-73. E-mail: m.shakurskiy@psuti.ru

Kozyreva Nadezhda Ivanovna, Povolzhskiy State University of Telecommunications and Informatics, 77, Moskovskoe shosse, Samara, 443090, Russian Federation; Associate Professor of Information Security Department, PhD in Technical Sciences. Tel. +7 927 208-31-99. E-mail: n.kozyreva@psuti.ru

Makarov Igor Sergeevich, Povolzhskiy State University of Telecommunications and Informatics, 77, Moskovskoe shosse, Samara, 443090, Russian Federation; Head of Software Engineering Department, PhD in Technical Sciences. Tel. +7 937 208-80-66. E-mail: i.makarov@psuti.ru

References

1. Sergienko A.B. *Digital Signal Processing*. Saint Petersburg: BHV-Piterburg, 2011, 768 p. (In Russ.)
2. Buranova M.A. et al. Simulation and analysis of generalized queue system with arbitrary distribution of system parameters. *Infocommunicacionnye tehnologii*, 2015, vol. 13, no. 3, pp. 252–258. (In Russ.)
3. Kartashevsky V., Kozyreva N., Makarov I. The numerical method for analysis of arbitrary type queuing systems application. *2016 3rd International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S and T 2016)*, 2016, pp. 109–111.
4. Barker E., Kelsey J. *NIST Special Publication 800-90A. Revision 1. Recommendation for Random Number Generation Using Deterministic Random Bit Generators*. National Institute of Standards and Technology, 2015, 110 p. URL: https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-90ar1.pdf (accessed: 01.10.2024).
5. GOST R ISO 28640-2012. Statistical methods: Random number generation. Moscow: Standartinform, 2014, 36 p. (In Russ.)
6. Rukhin A. et al. *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. National Institute of Standards and Technology, 2010, 131 p. URL: https://math.uni.wroc.pl/~lorek/teaching/files/nistspecialpublication800-22r1a.pdf (дата обращения: 01.10.2024).
7. L'Ecuyer P., Simard R.J. TestU01: A C library for empirical testing of random number generators. *ACM Transactions on Mathematical Software (TOMS)*, 2007, vol. 33, no. 4. DOI: 10.1145/1268776.1268777
8. Moldovyan N.A., Moldovyan A.A., Ieremeev M.A. *Cryptography. From Primitives to Synthesis of Algorithms*. Saint Petersburg: BHV-Peterbutgh, 2004, 448 p. (In Russ.)
9. Blum L., Blum M., Shub M. A simple unpredictable pseudo-random number generator. *SIAM Journal on Computing*, 1986, vol. 15, pp. 364–383.
10. Method and device for generating sequences of random numbers: patent 2716357. Russian Federation. No. 2018133894 / M.V. Shakurskiy (RU); zayavl. 25.09.2018; opubl. 11.03.2020, byul. no. 8. (In Russ.)

Received 11.11.2024
Accepted 11.12.2024