A novel hybrid mechanism for generation of pseudo-random sequences for data protection purposes

Manuel José Maldonado[1], José Luciano Maldonado[2]
[1]Programa de Doctorado en Ciencias Aplicadas
Universidad de Los Andes
Núcleo La Hechicera, Mérida, 5115
Venezuela

[2]Instituto de Estadística Aplicada y Computación, IEAC
Universidad de Los Andes
Núcleo Liria, FACES, Mérida, 5115
Venezuela

Abstract— In this paper, a mechanism for the generation of pseudo-random numbers with a very high level of randomness is proposed. The design of this mechanism is based on geometric structures simulated by software, specifically rotating cylinders that are activated, also, with random inputs produced by means of events of the electronic system on which it operates. Due to the random quality, its outputs can be used to implement robust systems for data protection.

Keywords—pseudo-random numbers, generation of pseudo-random numbers, randomness indexes, pseudo-random number production engines.

I. Introduction

PSEUDO-RANDOM number production engines are of vital importance for disciplines such as simulation, game theory and cryptography, among others. However, most of these engines are the product of mathematical procedures that do not sufficiently take into account the uncertainty factor. On the other hand, it is known that the greater the uncertainty about the knowledge of a system, the greater the randomness will be linked to its level of predictability, and this fact is very relevant in the field of information protection, specifically cryptography [1],[3],[4],[5]. But other categories of pseudo-random generator engines are widely used today, for applications where a high random level of number generation is desirable. There are the purely random methods, which operates under a continuous feeding of random inputs provides by physical phenomena that are not predictable in their occurrence nor their magnitude. This kind of engines serves for simulation and game theory process, and due their nature these are not replicable for this reason not suitable in cryptography or information protection processes. In the last years, a new category of pseudo-random generator engines is emerging: the hybrid methods or engines, which consist in the merge of the two kinds previously described. These are engines that receives inputs from unexpected events of the real world, and with a mathematical treatment of those inputs proceeds to generates the pseudo random numbers.

As previously described, it is clear that the future of information protection does not belong to deterministic methods or algorithms nor to the purely random. Additionally, the power of computers is constantly increasing, which requires that the protection of information has to advance at an even higher rate. For example, the combinatorial strength offered by quantum computers is already foreseen, with enough QBITS, to break in relatively short times the current block-symmetric and key-dependent methods, such as AES, Serpent, Twofish, and also those asymmetric algorithms such as RSA, Gamal and elliptic curve algorithms, ECC. On the other hand, pure random methods, for information protection processes, have the drawback that randomness does not allow replication, therefore the inverse function in the decryption process is impossible.

So, knowing the present of the generation of pseudo-random numbers, this paper proposes the development of a hybrid method for the generation of secuences of pseudo-random numbers that can be used in a wide variety of areas, from engineering to mathematical processes of simulation and, specifically, in disciplines as demanding as those related to information protection.

II. The generation of random numbers

There is a vast variety of methods to generate numbers with the qualification of “random”, that is, numbers with acceptable levels of randomness [4],[6],[8],[11],[12]. These methods can be classified into mathematical methods, purely phenomenological methods and hybrid methods [2],[4],[10].

Mathematical methods are those that use mathematical formulas or algorithms to generate random numbers [4],[6],[8]. These methods, although achieve an acceptable uniformity in the frequency of appearance of the values in a certain range, by presenting periodic repetitions in their content, and their sequences being known and being associated with the value of a generating seed, are not eligible for purposes of information protection, that is, these methods are not suitable for cryptography [13], since their level of indeterminacy is quite deficient.

Regarding the purely phenomenological methods, these are based on the study of randomness in unexpected physical events, for example, quantum ones, over which there is no control or total knowledge. In this type of method, it is necessary to study the output of the system to evaluate, using randomness test protocols, whether physical events of random behavior are really being faced, or whether the intervention of one or more variables biases the generation, preventing a uniform distribution of the output values. Although these methods serve to capture truly random data sequences, those are useless for information protection applications, since they cannot be replicated in decryption [8],[9],[11]. But serve, perfectly, for simulation applications and games of chance.

In relation to the hybrid methods [10],[14], those combine data from the physical world with numerical methods to, in real time, obtain an output probability distribution very close to the uniform one. Consequently, these methods produce sequences of numbers with random characteristics. Usually, these sequences are not replicable.

III. Proposed Hybrid Method, the cylinder system

The proposed hybrid model has the peculiarity that the sequences of numbers it produces can be replicated. These types of models are useful in information protection processes, in addition to their applicability in game theory and simulation. In the model that will be described, which will be called the Cylinder System from now on, event values of the physical world are captured to trigger the production of numbers with uniform distribution and a very good level of randomness.

Data from the physical world comes from events such as the clock time of the computer system used, processor cycles, process identifier, mobile device accelerometers, GUIDs, a keystroke, the position of the mouse on the monitor, among other physical events. Values for each event are selected from within the range of values for that event, and arithmetic operations are performed on those values to create larger numbers. If the values of the physical random variables are well distributed, the resulting large numbers will tend to retain the distribution characteristics of the originally selected numbers. Admittedly, even, if the randomness of large numbers achieves high levels, these numbers are really pseudo-random.

Once the large numbers are achieved, those are introduced into a set of three-dimensional structures in the form of cylinders, hence the name of the Model's Cylinder System. Rotations of such structures are then simulated, and sequences of these large numbers at positions produced by alignments of those structures are captured, i.e. read.

This system of rotational structures is implemented in such a way that:

a. Once the large numbers have been placed in the rotational structures, both to start the rotations and to read the numbers, random initialization variables must also be generated. These variables will indicate the direction of rotation of each structure (clockwise or counterclockwise), its speed of rotation and the initial positions for reading.

b. The reading of numbers in the rotational components is done following predefined alignments for each rotational structure. Each read number is modulated to the required range in the call of the function that requests the generation of a random number.

c. Unexpected sequence breaks occur, that is, unexpected jumps, implemented by software, when certain alignments or states of the rotational structures are detected, so that, in such circumstances, a new reading position is established for each structure, also changing the movement parameters of that structure (speed and direction).

The Cylinder System consists of a configuration of N cylinders, one inside the other. Each cylinder, in turn, is made up of M bands, and each band contains L large numbers. So that the operation of the Cylinder System can be understood, we will assume the particular case in which a system of 5 cylinders is built, with 5 bands each and 50,000 large numbers in each band. In any case, the dimensions of the Cylinder System are adopted following RAM memory availability criteria, in most of the computers and mobile devices used today. That is, Cylinder Systems can be implemented as large or as small depending on the availability of RAM memory.

To build the Cylinder System, each of the cylinders can be represented as a two-dimensional array (M rows or bands by L columns or cells), as can be seen in Fig. 1. It is assumed that column 1 (a b), is adjacent to the last column L (c d).

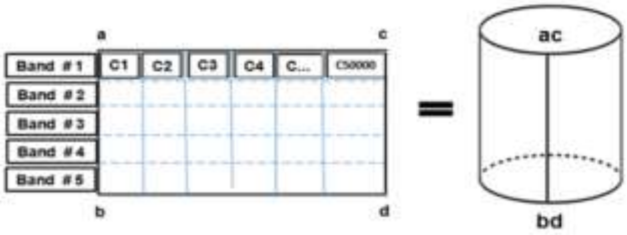

Figure 1. General architecture of a Cylinder System

Next, the creation and operation of the Cylinder System is explained. Two sets of numbers are used for this:

1. Permanent numbers set: It is a collection of sets of numbers called DNA. These sets of numbers are generated through physical events resulting from the user's interaction with their computer or mobile. This DNA is stored hidden in a binary file, and is protected so that only the user of the computer has access to it. Based on this DNA, the Cylinder System is built.
2. Modifier numbers set: These are five numbers that are included in the call to the function that builds the Cylinder System. These five numbers are also obtained through physical events resulting from the user's interaction with their computer or mobile, that is, these are unpredictable numbers from the user's system. The modifying numbers are different for each Cylinder System that is implemented, and their function is to modify the content of the DNA with which the Cylinder System is built on any time.

DNA can be made up of an arbitrary number of sets of numbers. For the construction of the Cylinder System, used as an example in this paper, three sets of numbers are used that make up the DNA:

1. A set of 25,000 numbers with range 1 to 255.
2. Four numbers with values between 250 to $2^{16} – 1$.
3. Five numbers with values between 250 to $2^{32} – 1$.

By jointly manipulating the ADN and the modifying numbers, by means of basic arithmetic operations, each cell of a new Cylinder System is filled with numbers of dimension greater than 10^6. In the described Cylinder System, the value of these numbers, usually, ranges between 10^6 and 10^9, although these can sometimes reach values close to 2^{31}.

The modifier numbers are defined in the range given by a 32-bit integer, and thanks to these, a big collection of Cylinder Systems can be obtained from the same DNA, where each Cylinder System is really a different generator of pseudo-random numbers. This is a particular property of the Cylinder System, which allows a great variety of content, so each time it is initialized it will present a different content and, therefore, the random sequences it provides are different each time.

In Fig. 2, you can see a Cylinder System in detail. On the left, the outer cylinder can be seen with its five bands loaded with numbers (decimal digits were used in each cell for a better appreciation). On the right side of the figure, the Cylinder System appears in perspective, showing the internal cylinders with their bands and numbers in the cells.

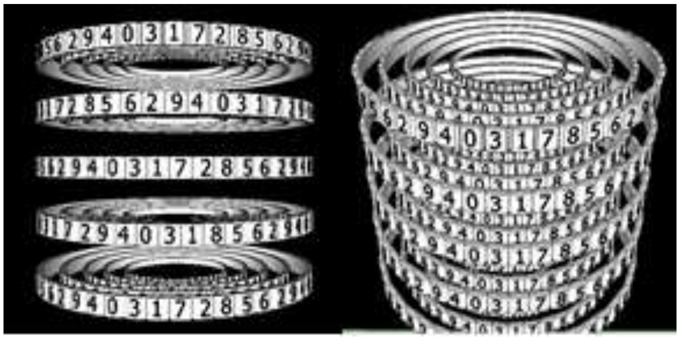

Figure 2. Details of a Cylinder System

Next, the modifying numbers are manipulated to calculate the necessary data for the rotation of the bands of each cylinder and to calculate the cell, of each band, where the reading in the Cylinder System will begin. The data calculated is:

1. Direction of rotation of each band (clockwise or counterclockwise). Since there are 5 cylinders with 5 bands each, 25 rotation direction data is required.
2. Rotation speed of each band. There are three scroll speeds: slow, medium, and high. In this case, too, 25 rotational speed data is required.
3. Initial reading site for each band, to initiate the process that will produce a pseudo-random number each time it is requested. Likewise, 25 data is required.

In each request for a pseudo-random number, the corresponding cell is read, as determined by manipulating the modifying numbers. Then, that band is rotated according to the pre-established direction and speed, also, by the modifying numbers. Subsequently, the Cylinder System positions itself in the next band to be read, depending on whether it is being read, either band by band of the same cylinder, which would correspond to a vertical reading as shown in Fig. 3, or a reading horizontal, in which the same bands of all the cylinders can be read, as can be seen in Fig. 4.

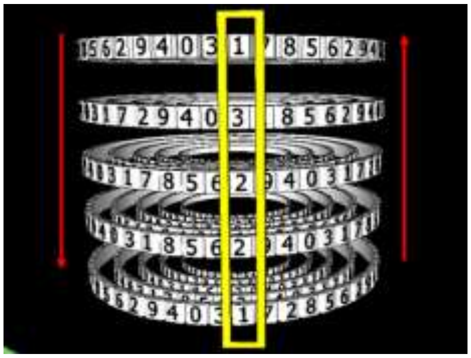

Figure 3. Vertical reading of the bands of a cylinder

Due to the fact that each band can turn in a different direction and speed than the other bands, and that the initial reading cell can also be different within the band, with respect to the initial cells of the other bands, it is very difficult for an alignment of reading positions to be repeated, therefore, it is really almost impossible for the same sequence of pseudo-random numbers to occur, even an of short length.

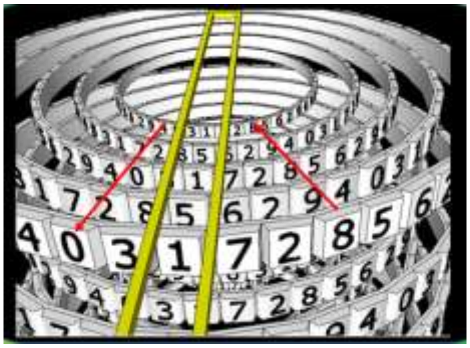

Figure 4. Horizontal reading of the bands of all cylinders

In the Cylinder System, we have a phenomenon that can be included in the "quantum" category, due to the unpredictability of its appearance. Let us remember that the term "quantum" is used to highlight a classical quality of certain phenomena described in physics, such as the uncertainty regarding their occurrence in time [7]. In our case, this phenomenon is related to modifier numbers, which due to their origin can be considered random. Thus, these modify the Cylinder System's DNA in an unexpected way each time a Cylinder System is built, resulting in the numbers that fill the cells having an unpredictable value. By extension, the pseudo-random sequences generated are unpredictable.

This kind of quantum jump was implemented in the Cylinder System, through a mechanism that, when detecting, in certain situations, coincident properties in the data read, modifies the operating variables of all the bands instantly. That is, at that instant, the speed of rotation, the direction of rotation and the reading sites of each band are modified, which could be described as a random jump to a new cell within each band, and from there each band will spin with redefined speed and direction, delivering unpredictable results.

Quantum jumps were implemented thinking of breaking, in an unexpected way, the sequence of production of pseudo-random numbers of the Cylinder System, and in this way achieving a closer approach to the desired property of these numbers, randomness.

IV. GENERATION OF PSEUDO-RANDOM SEQUENCES WITH THE CYLINDER SYSTEM

To generate a pseudo-random number, the Cylinder System read function is invoked. This function receives an integer as a parameter, and returns an integer between zero and the integer value given as a parameter minus 1. Therefore, to generate a pseudo-random sequence, as many calls are made as numbers are required to constitute that sequence.

When starting the reading process, the Cylinder System makes readings on the vertical alignments of the external cylinder, and moves on to the next concentric cylinder. Finishing the readings of the innermost concentric cylinder, start with the horizontal readings. To make the horizontal readings, it begins by reading the alignments made on bands number 1 of all the cylinders, and when these readings have been made, it goes to bands number 2, and so on until reaching the last bands (in the example described, the last bands are number 5). Thus a reading cycle is completed, and a new reading cycle is started. This process is repeated until the required sequence of pseudo-random numbers have been read.

Every time a number is read from any band, it scrolls in the predetermined direction and at the predetermined speed. Each bands moves as many places as its speed sets: high speed moves the band 3 places, medium speed moves the band 2 places, and slow speed moves the band a place. The next reading, on this band, will be done on the new reading alignment site. This is how the readings are produced, in a process that is restarted when certain conditions are detected in the reading positions of the bands, and that is the moment the quantum jumps are produced. This process stops when the request for the generation of a pseudo-random number to the Cylinder System ends.

The readings produce pseudo-random numbers, since those are generated under pre-established conditions when creating the cylinders, but the complexity of the Cylinder System's operation means that the numbers produced present a high level of randomness.

In test protocols of up to 1,000,000,000 requests for pseudo-random numbers in the range 0-255, an excellent level of randomness was observed. One of the properties that contributes most to the achievement of this result is the fact that the positions of the bands are not realigned periodically, but through quantum jumps that are unexpected.

The measurement of the level of randomness of the Cylinder System is described in another paper, by the same authors of this paper, which is being submitted to arbitration in another Scientific Journal. In that paper, a robust method is proposed to measure the randomness of pseudo-random number generators. The randomness of the Cylinder System turned out to be of a high level, of the order of 0.97 on a scale of 0.0 to 1.0.

V. MODULATION IN THE CYLINDER SYSTEM

Given the characteristics of the numbers that are stored in the Cylinder System, that is, large numbers that usually exceed the ranges within which a requested pseudo-random number must be delivered, it is necessary to modulate the output. The modulation process is explained below:

Let k and v to be two positive integers between 10^6 and 10^9, if k is the content of the cell to be read and v is the integer parameter of the function that invokes the generation of a number with the Cylinder System, so i=k%v is the pseudo-random integer returned and corresponds to the modulus of dividing k by v. As can be deduced, reading k will produce up to k different responses depending on the value of v. This implies that, when a new reading of the Cylinder System is made again in the cell that contains k, but with a new value v, the response will be different, even generating that response from the same cell.

Additionally, the rotation parameters of each band cause that after the Cylinder System begins to rotate, it is unlikely that a new alignment will occur in the same place for all the bands, due to the occurrence of quantum jumps, which causes read sites to change unexpectedly, thus generating different numbers on each read. Anyway, it is possible, albeit very remotely, that the rotation and position parameters are repeated, but in that case the new parameter v will determine the answer. So, in the System of Cylinders, pseudo-random number sequences with successive repetitions are very unlikely to occur.

VI. SCOPE OF THE CYLINDER SYSTEM

Although the proposed pseudorandom number generator, mainly, was created with the aim of being used in information protection applications, it can also be used in any type of application where the use of pseudorandom number sequences is required. Fig. 5 illustrates how the Cylinder System would be used for information protection applications, and Fig. 6 illustrates how it would be used in cases where information decryption is not required. The difference between these two ways of using the Cylinder System is that in the first case, there is a mechanism to recover the original values of the encrypted data, while in the second case this mechanism is not necessary. This difference is shown in the feedback block of those two figures.

In any case, a scheme is presented here that avoids the possible repetition of values returned by the Cylinder System and, therefore, the possible periodicity is eliminated. This is a very relevant feature of the Cylinder System because the most, if not all, of the current pseudo-random generation systems of a mathematical nature have not been able to solve the problem of periodicity.

Once the cells in the Cylinder System are filled, these remain unchanged during a pseudo-random number generation session. This property may be useful for cryptographic processes, because the Cylinder System allows a pseudo-random production with high randomness indices and with the possibility, in the receiver, to decrypt the data.

It is possible, too, to vary the content of the cells of the Cylinder System, through mathematical processes that involve the parameters that intervenes in the reading of a particular cell. Such parameters are the value obtained by the modulation of a reading just made, the reading position, the speed and the direction of the band involved, and other parameters included in the reading process

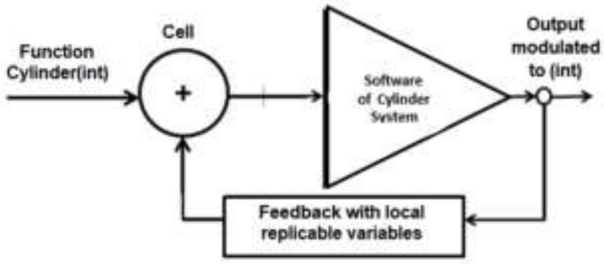

Figure 5. The replicable Continuous Feedback Cylinder System

Let ***Pos*** the lecture position recently reading, speed the rotation speed of the involved band and ***Dir***, the rotation direction of that band. It is possible to modify the content of a cell recently reading using (1):

$$NVC = CV + Pos * ((speed + 1)\%256 + I)\%337 + Dir * 39 \tag{1}$$

Where:

I: Resulting value of the modulation.
CV: Current value of the cell.
NCV: New current value of the cell.

In (1), where 337 and 39 are, can go any values that are not multiples of 10 or exceed the 350 value (so as not to approach the numerical limit of the type of data that make up the cells of the Cylinder System). This is because large numbers are defined as unsigned integers in the range 10^6 to 10^9, and we don't want them to exceed 2^{31} value.

The replicable feedback process, shown in Fig. 5 and by (1), it is feasible for cryptographic [13], simulation, game theory and other applications. But if a higher degree of randomness is desired, the Cylinder System can adopt the scheme shown in Fig. 6, where it works with a feedback scheme based on truly random variables produced by physical events.

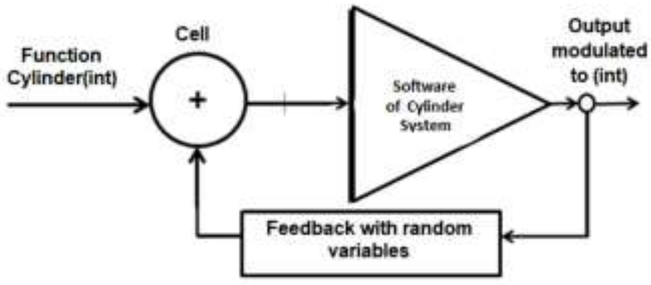

Figure 6. The Cylinder System with continuous random feedback

The Cylinder System with random continuous feedback, represented by Fig. 6, is not applicable to cryptographic processes, since the receiver would be unable to replicate the Cylinder System that originates the encryption, this scheme being ideal for simulation processes, theory of games, and other applications that do not require replicating pseudo-random sequences.

VII. SYTEM ADDITIONAL FEATURES OF THE CYLINDER SYSTEM

As the range of values each cell varies between 10^6 and 10^9, approximately, for the example described, the Cylinder System can produce approximately $(10^9 - 10^6)^{1,250,000}$, that is, $7.249073*10^{11,249,456}$ different pseudo-random sequences, with the described DNA and proposed architecture for the Cylinder System of this example.

On the other hand, because the same modifier numbers produce the same number sequences with the same DNA, the process is replicable, making it useful in cryptographic processes.

The content of the random sequences for each DNA, which can be shared by two users of any cryptographic method that uses the Cylinder System, is different from the content of the sequences of another pair of users of the same cryptographic method, since their DNAs would be different. In the Cylinder System of the described example, it is possible to encrypt a file, using a specific DNA, in approximately $8.0 * 10^{\mathbf{61}}$ different ways.

If you want to know how many different Cylinder Systems can be built, with the following three sets of numbers that make up the DNA used for this example:

- 25,000 numbers between 1 and 255: If 255 values can be assigned to each cell, for 25,000 cells, we have, approximately, 3.195294*$10^{60,163}$ permutations of all values in the range 1 to 255, for 25,000 cells.
- Five numbers between 250 and 2^{16} – 1: Each number is in a range of 250 to 65.285 and those five numbers produce, approximately, 1.208833588708967*10^{24} permutations of all values.
- Four numbers between 250 and 2^{32} – 1: Each number is in a range of 250 to 4,294,967,295 and those four numbers produce, approximately, 3.4028236*10^{38} permutations of all values.

It is found that, approximately, 3.195294*$10^{60,163}$ *1.208833587089*10^{24}*3.402823666*10^{38}= 1.3143674256* $10^{60, 226}$ different Cylinder Systems can be built.

VIII. CYLINDER SYSTEM TESTS

In the DNA testing stage, up to 1,000,000,000 sets of 25,000 numbers in the range 1-255 were generated. These 25,000 numbers are the main component of the DNA for the Cylinder System.

The frequency distribution of the numbers within the range 1-255 presented an excellent uniformity, which produced very high quality DNAs, this in turn allowed the construction of Cylinder Systems that generated sequences with very good randomness indexes [9].

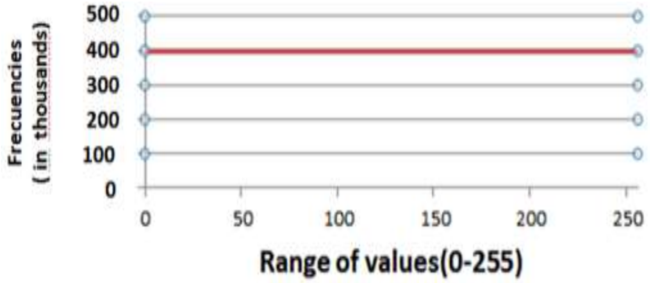

Figure 7. Frequency distribution for 1,000,000,000 data.

The graphic in Fig. 7 shows the cumulative frequency distribution for 1,000,000,000 data, between 0-255, that were produced by a Cylinder System built with replicable continuous feedback DNA. It can be seen in this graphic that, even in this case, (replicable feedback) the uniformity of the data is excellent.

IX. DISCUSSION

It is well known that, virtually, to all existing pseudorandom number generators can have their outputs predicted, after repeatedly observing the sequences of numbers they generate. For this reason, we consider that the use of random numbers produced by purely mathematical procedures, as a way of guaranteeing unpredictability, is totally wrong. Furthermore, it is also known that most current pseudorandom number generators fail statistical tests of randomness. Therefore, as a way to overcome these problems, the Cylinder System was proposed, which, after being evaluated in a vast amount of tests, has shown to be very efficient, since, as described in the preceding sections, it presents a very high randomness index (of the order of 0.97 on a scale of 0.0 to 1.0), and it has the particularity that it is almost impossible to observe that it repeats some sequence of numbers, not even in millions of tests. That is, with the System of Cylinders proposed here, it can be affirmed that there is a way to generate sequences of pseudorandom numbers, that guarantees not only that the sequences are practically never repeated, but also that this system can be perfectly integrated in any encryption system, so its application in the protection of information is also guaranteed.

On the other hand, in view of the continuous growth of the calculation potential of machines, what is coming for the future is to ensure, even more, the management of information. In this sense, the data structures of the Cylinder System can be scaled to a higher level, where more than three dimensions are dealt with, in which case it will no longer be possible to speak of Cylinder Systems, but of other multidimensional architectures. But, for now, the developed Cylinder System perfectly supports any attempt to be broken, both by current machines and by those that will be developed in the near future.

X. CONCLUSIONS

An adequate protection of the information involves creating encryption with models that imply the maximum possible indeterminacy, and the Cylinder System has a great capacity to make its production resemble truly random sequences.

The Cylinder System poses such a challenge to codebreakers that, even for quantum computers, it constitutes an insurmountable barrier, since it confronts them with a scheme of total indeterminacy.

The number of elementary particles in the universe is 2.2* 10^{80}, while the number of different cylinders that can be created, only with the Cylinder System of the example described, is approximately 1.2894880171011*$10^{60,226}$, that is, many can be created more sets of different cylinders than there are elementary particles in the universe, which leaves no doubt about the high randomness of the pseudo-random sequences generated by the proposed Hybrid Mechanism. And the number of pseudo-random sequences that these Cylinder Systems can produce is approximately 7.249073*$10^{11,249,456}$.

The number of different Cylinder Systems that can be built based on DNAs, also different, is only limited by the imagination and the availability of RAM memory to store those architectures.

The proposed pseudo-random generator is low cost, since it can be implemented in any computer (laptop or desktop) and does not require sophisticated physical resources such as generators based on quantum devices.

The proposed generator overcomes the limitations of pseudo-random generators based on mathematical models, the purely phenomenological models, and non-replicable output hybrids.

The hybrid mechanism for generation of pseudo-random sequences described, is the first geometric model composed of rotational structures and fed with truly random variables. For the state of the art, it has no limitations, quite the contrary, only advantages, since in addition to guaranteeing non-repeating pseudo-random sequences, since it does not start from a seed, it can be integrated into any cryptography system, for which it also guarantees protection of information. On the other hand, its memory and speed requirements are not an issue for today's hardware platforms. All these characteristics guarantee a long life for this proposal, however, to ensure its existence alongside the development of computers, this proposal can scale to a higher level through the use, in the future, of multidimensional data structures. so you can scale from a cylinder system to a multidimensional system.

REFERENCES

[1] A. Scott. The Quest for Randomness, American Scientist, May - June 2015. Volume 102, Number. Page 170.

[2] The Importance of True Randomness in Criptography. Authen Tec.com. http://www.authentec.com December 14 2010.

[3] J. Katz and Y. Lindell. Introduction to Modern Cryptography CRC PRESS 2015. London New York Washington, D.C. august 31 2007, Chapter 10 333 – 375.

[4] Á. Cervantes Generación de números pseudoaleatorios eficientes en microcomputadoras, Tesis doctoral, UNAM, México, 2006.

[5] B. Schneier. Applied Cryptography, Chapter 1, Second Edition, John Wiley & Sons, New York City, New York, USA, 1996.

[6] S. García. Generadores de números aleatorios en criptografía. Universidad Politécnica de Madrid, España, 1994.

[7] Equipo de Microsoft Quantum y Microsoft Research. La criptografía en la era de las computadoras cuánticas. News Center Microsoft Latinoamérica. Marzo 5, 2020.

[8] A. Gallego. Generador de números aleatorios. Departamento de Ingeniería Eléctrica y Electrónica. Universitat Rovira e Virgil. Septiembre 2011.

[9] A. Godínez et al. Pruebas estadísticas de generadores de secuencias pseudoaleatorias para aplicaciones criptográficas. 1er Congreso Iberoamericano de Instrumentación y Ciencias Aplicadas San Francisco Campeche. Campeche, México. Volumen: ISSN 2395-8499. 2014.

[10] O. Skliar, R. Monge, V. Medina, S. Gapper, G. Oviedo. A Hybrid Random Number Generator (HRNG). Rev. Mat vol.18 n.2 San José. Dec. 2011.

[11] J. Haw, S. Assad, A. Lance, N. Ng, V. Sharma, P. Lam. Maximization of Extractable Randomness in a Quantum Random-Number Generator. Symul. Phys. Rev. Applied 3, 054004 – 11 May 2015.

[12] T. Unkašević, Z. Banjac, and M. Milosavljević. A Generic Model of the Pseudo-Random Generator Based on Permutations Suitable for Security Solutions in Computationally-Constrained Environments. Sensors Dec 2019.

[13] D. Hutchinson. Randomness in Cryptography: Theory Meets Practice. Information Security Group Department of Mathematics Royal Holloway, University of London, 2018.

[14] B. Sunar, W. Martin, D. Stinson. A provably secure true random number generator with built-in tolerance to active attacks. IEEE Trans. Comput. 56(1), 109–119 (2007)

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

-Manuel José Maldonado is the creator of the Cylinder System approach for data protection purposes, who carried out the programming in C++ language and wrote the initial version of the paper and participated in its revision and modification.
-José Luciano Maldonado participated in the evaluation of the Cylinder System approach, in the review and editing of the paper, and worked on the final presentation.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

This research did not receive funding from any organization.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)