A new pseudorandom bits generator based on a 2D-chaotic system and diffusion property

Omar Z. Akif[1], Sura Mazin Ali[2], Rasha Subhi Ali[3], Alaa Kadhim Farhan[4]
[1]Department of Computer Science, College of Education for Pure Science (Ibn al-Haitham), University of Baghdad, Iraq
[2]Political Science College, Al Mustansiriyah University, Iraq
[3]Ministry of Culture, Baghdad, Iraq
[4]Department of Computer Sciences, University of Technology, Baghdad, Iraq

Article Info

Article history:

Received May 8, 2020
Revised Dec 5, 2020
Accepted Mar 17, 2021

Keywords:

2-D Chaotic map
Diffusion property
Mouse movement
PRNG

ABSTRACT

A remarkable correlation between chaotic systems and cryptography has been established with sensitivity to initial states, unpredictability, and complex behaviors. In one development, stages of a chaotic stream cipher are applied to a discrete chaotic dynamic system for the generation of pseudorandom bits. Some of these generators are based on 1D chaotic map and others on 2D ones. In the current study, a pseudorandom bit generator (PRBG) based on a new 2D chaotic logistic map is proposed that runs side-by-side and commences from random independent initial states. The structure of the proposed model consists of the three components of a mouse input device, the proposed 2D chaotic system, and an initial permutation (IP) table. Statistical tests of the generated sequence of bits are investigated by applying five evaluations as well as the ACF and NIST. The results of five standard tests of randomness have been illustrated and overcome a value of 0.160 in frequency test. While the run test presents the pass value t0=4.769 and t1=2.929. Likewise, poker test and serial test the outcomes was passed with 3.520 for poker test, and 4.720 for serial test. Finally, autocorrelation test passed in all shift numbers from 1 to 10.

Corresponding Author:

Omar Z. Akif
Department of Computer
College of Education for Pure Science (Ibn al-Haitham)
University of Baghdad
Baghdad, Iraq
Email: omar.z.a@ihcoedu.uobaghdad.edu.iq

1. INTRODUCTION

The most significant concern with any current digital data is the privacy and security, for if they are breached, this will lead to compromise in their confidentiality and integrity [1]. Also, sharing on the data communication network is one of the main security challenge [2]. Consequently, the main goal of security is to protect the vital information from illegal interception, tampering and distribution over the networks without authorised access [3], [4]. Ensuring that information is secure has become very important in several fields, such as medical imaging, internet communication, military communication, telemedicine, and multimedia systems. Therefore, increasing the awareness in a cryptography field is very important to protect the information [5]. Cryptosystems rely significantly on the factors of complexity, confusion and diffusion, which is why a number of changes have been made to such systems to enhance these factors [6]. An optical key distribution quantum cryptography based on the BB84 protocol is experimentally and quantitatively analyzed in the present study. To this end, a secret quantum key is generated, followed by testing of the

BB84 protocol security through the introduction of Eve in the system. The synchronization of cubits between Alice and Bob is disrupted by Eve, resulting in bit errors. To make quantum cryptography more secure, a one-time pad method and chaotic signal emitted by a semiconductor laser with optical feedback are employed. System security can be maximized through a combination of the quantum key and chaotic signal [7]. Recently, there are several applications of chaotic systems have been mentioned and executed by researchers [8].

Chaotic systems present numerous important attributes, such as sensitivity to primary conditions and topological transitivity, which are associated with certain specifications, like cryptography confusion and diffusion. Consequently, a correlation has been established between chaotic systems and information encryption [9]. A chaotic system in encryption has become favored by increasing numbers of researchers, with various techniques having been used [10]. The notion of chaos synchronization was first introduced by Pecora and Carroll in 1990 [11]. In recent years, different chaos-based cryptosystems have been proposed due to the close relationship between chaos and cryptosystems [12].

These cryptosystems use chaotic maps to generate pseudorandom number generators (PRNGs), which can increase the randomness of any designed system in which they are utilised [13]. Most chaotic maps have the challenge of a limited range for keys, especially with 1D chaotic maps, which provide weak security [14]. Discrete-time chaotic maps require spatial discretization for digital realizations, which results in dynamical degradation [15]. A non-linear deterministic system of high complexity can be addressed based on chaotic theory. Improved security can be achieved by creating a hyper chaotic system through the integration of a number of chaotic techniques [16].

PRNGs developed from a chaotic map can be characterized according to three methods: (a) the extraction of bits in a direct way from orbits; (b) the production of bits based on the interval of the position of the orbit; and (c) the equalization of the chaotic orbits [17]. Since the chaotic orbits provide knowledge about the chaotic system, when based on a 1D chaotic map, obtaining a secure key becomes possible for an intruder. Hence, 2D chaotic systems have been proposed for preventing the disclosure of data encrypted by PRNGs based on chaotic maps [18]. Chaotic systems are identified by a high sensitivity to the initial parameters and additional characteristics, such as periodicity, mixing, and high complexity [19]. A minor deviation in their initial states can result in a considerable change in behavior. Hence, to maintain a high level of security, it is necessary to combine multiple 1D chaotic maps to increase the cryptosystem's complexity. In this study, a new PRNG is created based on three phases, including the movement coordinates of a mouse input device, the output of a 2D chaotic map after being seeded with the output of the first phase for producing multiple pseudorandom sequences, and the initial permutation (IP) table (DFS) algorithm generated based on the output of these prior two phases.

2. RELATED WORK

In recent times, the use of the mathematical notion of chaos has proliferated within cryptography due to the ability of chaotic functions to display non-linear dynamics and arbitrary behavior. A close correlation exists between a number of characteristics of chaos (e.g. hypersensitivity to initial parameters and conditions, extreme mixing capacity, etc.) and the properties of confusion and diffusion, which are critical to cryptographic algorithms (e.g. encryption strategies) [20].

Integration of chaotic synchronization and the RSA algorithm into a double encryption has been devised. To achieve this, in the present paper, a design methodology for secure communication based on neural networks (NNs) in multiple time-delay chaotic (MTDC) systems is put forth. Constituting an asymmetric encryption, the RSA algorithm is advantageous primarily because factorizing two large prime numbers is incredibly challenging. Hence, processing this algorithm is more time-consuming in cases where the key is significantly lengthier. For the present work, a shorter key is employed so that the RSA algorithm can be processed faster, but by doing so the cryptosystem will be less secure. Double encryption, integrating chaotic synchronization and the RSA algorithm, is the solution employed in this study to improve cryptosystem security. Furthermore, to address the shortcomings of conventional GA, an IGA is suggested, which can facilitate the synthesis of a fuzzy controller in achieving exponential synchronization as well as in reducing the disturbance attenuation level, thus optimizing $X\infty$ performance [21].

The Lorenz and Lü chaotic systems are employed in this study to devise a field-programmable gate array (FPGA) pseudo random number generator (PRNG). Four distinct 3D chaotic attractors can be created based on these two chaotic systems. Specifically, the Lorenz chaotic system is the basis for one attractor, while the Lü chaotic system is that for the other three. An effective hardwired shifting and multiplexing model can mediate the reconfiguration of the output attractor associated with the suggested PRNG in real-time operation. Moreover, the suggested PRNG is incorporated in an FPGA cascaded encryption processor capable of input data ciphering 1-4 times sequentially, thereby taking advantage of the proposed reconfiguration feature. Every ciphering task involves setting the PRNG to a new configuration and its initialization is based on a portion of the encryption key. The encryption key size can differ depending on how many ciphering operations are necessary. VHDL is used for designing the suggested PRNG, while its synthesis is performed on Xilinx via the FPGA device XC5VLX50T and its analysis is based on MATLAB and the NIST statistical suite. The proportion of slices of FPGA used by the proposed PRNG is just 1.4%. Additionally, the PRNG's operating frequency can reach 78 MHz and the system also performs effectively on every NIST statistical test [22].

For the present study, it is proposed that the arbitrary behavior of certain chaotic maps can be enhanced via two integrated chaotic systems (ICS), which can perform cascade and non-linear combinations as well as producing novel structures by changing operations into three basic 1D chaotic maps. These systems can generate a number of chaotic maps of greater complexity. These differ from current maps in that they possess more sophisticated properties, such as broader chaotic ranges and behaviors of greater complexity. Furthermore, an ICS-based image encryption system is proposed to show how effective this system is. This approach displays good qualities regarding image encryption, as attested by the outcomes of simulation on various image types and detailed security analysis [23].

Image segmentation and multiple diffusion models are utilized constituting a new method of chaotic encryption. The initial step is determining the original value and control parameter associated with the chaotic map by generating an arbitrary key. The second step is application of the SHA-512 hash algorithm to produce a plain image-based hash array to serve as secret keys. The third step is image separation into sub-blocks and selection of a diffusion model on the basis of the parts values of the previously created array. The enhanced security and resilience to pervasive attacks of the suggested approach are confirmed by the outcomes of simulation and security analysis [24].

A novel four-dimensional chaotic system comprising four multiplier terms and four simple terms is put forward. Compared to other 4D systems, the proposed one is structurally and topologically dissimilar and generates two equilibrium points, with one being at the source. Equilibrium points and related stabilities, the power spectrum, a bifurcation diagram and a Poincaré map are the basis for the assessment of the key properties of the proposed system. Moreover, system trajectories are run to the zero equilibrium based on an optimal controller underpinned by the Riccati equation. Matlab and Simulink are employed for simulation of the novel system dynamics [25].

The present study is concerned with the creation of a communication channel prototype incorporating a chaotic cryptographic algorithm with a cipher feedback mode. Owing to its capacity for fast processing and reduced delay necessary for the voice channel, FPGA is used for implementation. Specifically, one Spartan-3 FPGA board is employed as a transmitter for encryption, while another Spartan-3 FPGA board is used as a receiver for decryption. The application of the encryption-decryption cycle for asynchronous communication effectively ensures the security of the voice channel, as revealed by empirical testing. Meanwhile, the non-encrypted channel is associated with the average MSE value of 0.3513 V2, average delay value of 202_s, and the average THD-N value of 17.52%. By contrast, the encrypted channel is associated with the average MSE value of 0.3794 V2, average delay value of 202 _s, and the average THD-N value of 20.45%. Hence, it is possible to restore the initial information transmitted via the encrypted channel at a level of quality not much different from the non-encrypted one [26].

3. METHODOLOGY

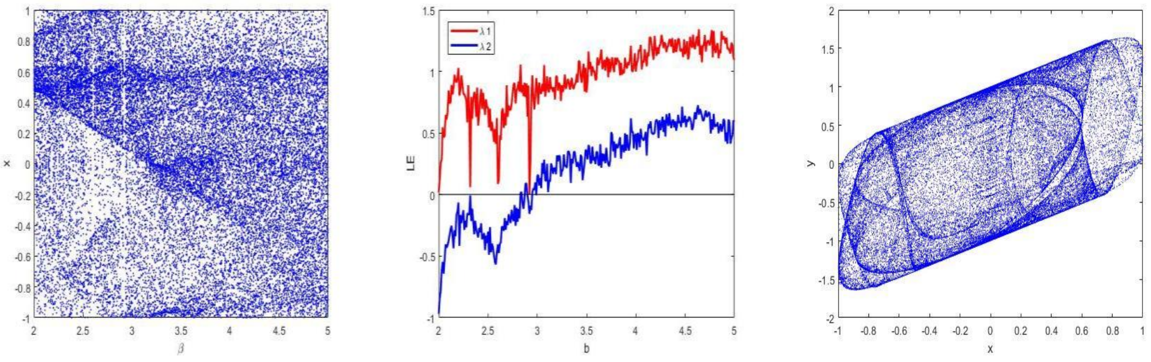
3.1. The new 2D chaotic map

The new 2D-chaotic system equations are given as (2):

$$x_{n+1} = sin(bx_n)sin\left(\frac{a}{y_n}\right) \quad (1)$$

$$y_{n+1} = ax_{n+1} + sin(by_n) \quad (2)$$

Where a and b are the control parameters that have been applied on the proposed system with a=0.8 and b between 2 and 5 (b=[2,5]). Therefore, if the value of b has been replaced by another value less than 2 and greater than 5, then the randomness and chaos will not be achieved. The proposed map is derived from Hénon and Sine maps, with the latter being modulated in the former. The result is then used to enhance the nonlinearity and randomness of the Sine map. Based on the Jacobean matrix to find the equilibrium points of (1) and (2), we found that it is completely unstable. The performance evaluation of Equations demonstrate the chancily of the proposed map. Based on the LE, we found in (1) and (2) are hyper chaotic with high sensitivity to its initial values (x,y) and control parameters (a,b), as illustrated in Figures 1(a), 1(b) and 1(c).



(a) (b) (c)

Figure 1. Illustrating the new hybrid 2D chaotic system, (a) The phase plane of, (b) The Lyapunov exponent of, (c) The bifurcation diagram (1)

4. PRNG VIA MOUSE MOVEMENT

True random number generators (TRNGs) and pseudorandom number generators (PRNGs) are the two major classes that can be used to obtain arbitrary sequences. More specifically, arbitrariness is generated by TRNGs based on a non-deterministic entropy source and specific post-processing functions, whereas multiple "pseudorandom" numbers are produced by PRNGs based on the so-called seed input and deterministic algorithms. PRNGs are more suitable for applications that necessitate numerous random numbers, because their speed generally exceeds that of TRNGs [27].

An additional operation before generating the pseudorandom numbers is required, which can be performed by a variety of approaches of a non-deterministic source, such as mechanical or electronic noise. These approaches need more devices to guarantee that their applications are not universal. A mouse input device is utilised to select an object on a computer monitor, which is moved by the user's hand. The x- and y-axes are two dimensions for any movement of the mouse on the screen that corresponds to the coordinates of its pointer at a particular location. Digital movement is considered to be reflected by the ensuing mouse cursor movement and an application programming interface (API) is designed to provide the distinct values of the mouse pointer with respect to the x- and y-axes at any point on the screen [17].

As the user moves the cursor over the monitor, a variety of values are obtained and processed. Hence, when the user moving the mouse pointer across a monitor, the various numbers will be generating. Repeating the cursor movement in the same pattern is difficult for the user, so the numbers generated from this technique can be considered a random sequence and an additional device is no longer needed. The approach offers a low cost, suitable, robust, and efficient technique for generating random bits [28].

5. INITIAL PERMUTATION TABLE

The initial permutation table is used in the data encryption standard (DES) with input including 64 numbered bits (1 through to 64). This table specifies the input permutation on a 64-bit block and transposes the input block by moving bit 58 of the plain-text to bit index 1. Then, bit 50 is moved to bit index 2, and this same pattern is repeated. The IP is then used to increase the diffusion of the generated random bits [29].

6. THE PROPOSED SCHEME

With the proposed method, a web application is developed to include the three components of the motion of the mouse, the 2D chaotic system, and the IP table. First, we collect the mouse cursor coordinates during its movement over the screen for a certain interval in order to generate a pre-established range of values. Repetition indicates that the same sequences will be continuously composed with the same seed, which is considered a limitation. Security is increased, if the manipulation is performed on those coordinates. However, repeating the movement of the mouse over the same area does not lead to obtaining the same previously generated coordinates, which indicates that its action is entirely random.

When these numbers are sufficient for seeding the 2D chaotic map, the generator will be prepared for creating random sequences for utilisation in cryptographic applications. Then, the generated numbers are subjected to the IP table to increase the complexity. These process phases are illustrated in Figures 2 and 3.

Since the space of the generated key and the length of the period are the maximum, the produced keys have robust statistical properties, which are anticipated for any pseudorandom binary sequences used for cryptographic purposes. The randomness evaluation is performed through five statistical tests. Figure 2 illustrates the general structure of the proposed system. While Figure 3 explained the operating principle of the second part.

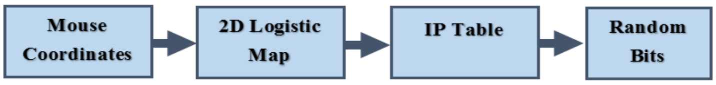


Figure 2. The general structure of the proposed method

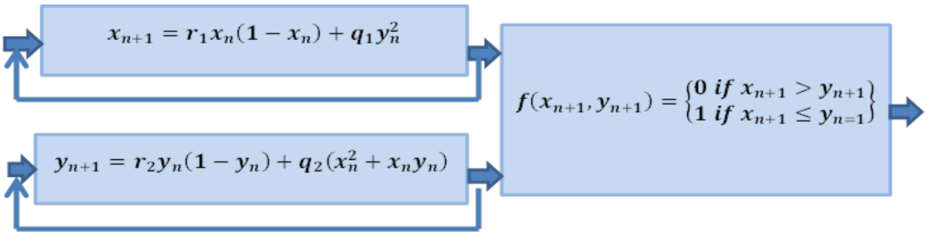

$x_{n+1} = r_1 x_n(1 - x_n) + q_1 y_n^2$
$y_{n+1} = r_2 y_n(1 - y_n) + q_2(x_n^2 + x_n y_n)$
$f(x_{n+1}, y_{n+1}) = \begin{Bmatrix} 0 \; if \; x_{n+1} > y_{n+1} \\ 1 \; if \; x_{n+1} \leq y_{n=1} \end{Bmatrix}$

Figure 3. The operating principle of the second part

Movement of the mouse cursor for a certain amount of time to produce sufficient coordinates (*x*-axis and *y*-axis) is the requirement associated with the initial step of the suggested random bits generator. Then, these coordinates are utilised as the initial conditions for the chaotic generator seeds. To this end, two sequences of an equal length *m* will be produced, one for the *x*-axis (referred to as *Xm*), and the second for the *y*-axis (referred to as *Ym*), a step programmed with JavaScript methods. To enhance security, the diffusion is increased by using the IP of the DES. The complete proposed method is implemented based on algorithms 1 and 2.

```
Algorithm 1: Mouse Motion
Input: X- Y-axes
Output: Two sequences (seedX[m], seedY[m])
Begin
Set X-old = 0, Y-old = 0;
Repeat
 $X_{new}$[m] = new X-axis;
$Y_{new}$ [m]= new Y-axis;
If ($X_{new}$[m] <>$X_{old}$) AND ($Y_{new}$[m] <>$Y_{old}$) Then
 m=m +1;
        $X_{new}$[m] = new X-axis;
        $Y_{new}$[m] = new Y-axis;
        $X_{old}$ = $X_{new}$[m];
        $Y_{old}$ = $Y_{new}$[m];
End If
Until Stopping Criteria Satisfied
For m = 1 To$m_{max}$Do /*numbers should be between [0,1] */
seedX[m] = $X_{new}$[m] × real factor<1
seedY[m] = $Y_{new}$[m]× real factor<1
End for
End
```

```
Algorithm 2: 2D-Chaotic PRBG
Input:seedX[m], seedY[m], MaxIteration, Start_Value
Output: Output
Begin
SetKeySize = 255;
For I = 1 To m Do
A= seedX[i];
 B = seedY[i];
For n = 1 ToMax_IterationDo
$X[n] = R_1A (1 - A) + Q_1B^2$
$Y[n] = R_2B (1 - B) + Q_2 (A^2 + AB)$
If (X[n] > Y[n]) Then
Rout[n] = 1;
Else
 Rout[n] = 0;
End If
 A = X[n];
                 B = Y[n];
End For n
End For i
End
```

7. STATISTICAL AND AFC

In this section, the statistical evaluation includes five tests. These tests typically utilised to determine if the sequences have the specific characteristics of a truly random sequence. In the case it successfully passes all five criteria, it is still not a guarantee that it was created by a random bit generator. The suggested generator presented here does provide truly random sequences in which all values gathered from ACF computations are close to zero. All the results obtained from the auto-correlation test are provided and explained in Tables 1 and 2.

Table 1. Standard five tests of randomness

Frequency	Runs test	Poker	Serial	Autocorrelation test
Pass value 0.160 with freedom degree “1” must be <=3.84	Pass value t0=4.769 with freedom degree “5” must be <=10.788 Pass value t1=2.929 with freedom degree “5” must be <=10.788	Pass value 3.520 with freedom degree “5” must be <=11.1	Pass value 4.720 with Freedom degree “3” must be <=7.81	Shitf no.1 >--> pass value 0.495 Shitf no.2 >--> pass value 0.041 Shitf no.3 >--> pass value 1.742 Shitf no.4 >--> pass value 2.042 Shitf no.5 >--> pass value 0.263 Shitf no.6 >--> pass value 0.170 Shitf no.7 >--> pass value 0.269 Shitf no.8 >--> pass value 0.391 Shitf no.9 >--> pass value 2.473 Shitf no.10 >--> pass value 0.044 With Freedom degree “1” must be <=3.84

Table 2. ACF results for the PRNG

Autocorrelation function, N/50000	
Time lag k	ACF(k)
1	-0.006761
2	0.00082
3	0.00776
4	0.00046
5	0.00002
6	0.00854
7	0.00168
8	0.00001
9	0.0050
10	-0.00004

8. NIST STATISTICAL TESTS

The NIST statistical test suite proposed in [30] is still widely used for testing randomness and involves 16 different types of tests. The significance level of each test in NIST has been set to 0.01. Hence, sequences pass a test, if the P value is greater than 0.01 and less than 1. In this study, a numerical experiment has been carried out. Specifically, 1000 groups of bit sequences have been generated with a length of 10^6 for each group and hence, 1000 groups have been chosen randomly for the initial condition. The means of the P values of each test are illustrated in Table 3. As the table shows, there is a significant difference between the two groups: the first group results being calculated based on the current study, whilst the second group pertains to the reference outcomes [31]. Additionally, according to the results illustrated in Table 2, the bit sequence of this research has passed all the tests successfully. Therefore, for the current study, it has been found that the sequence delivers good statistical performance and can achieve true randomness. In Table 3, a comparison with the reference outcomes [31] is provided. Clearly, the results of the current study demonstrate better means of the P_values than the same test results obtained from the reference [31].

Table 3. The comparison between this study and ref [31] according to means of P_value in the NIST test

Test Index	Means of P_value	Results	
	Proposal System	Ref[30]	
Approximate entropy	0.954	0.34973	Passed
Block frequency	0.528	0.33242	Passed
Cumulative sums	0.450	0.21495	Passed
FFT	0.666	0.41003	Passed
Frequency	0.663	0.42167	Passed
Linear complexity	0.376	0.51674	Passed
Random excursions	0.746	0.30147	Passed
Random excursions variant	0.452	0.29871	Passed
Longest runs of ones	0.744	0.25675	Passed
Overlapping template of all ones	0.412	0.31775	Passed
Rank	0.287	0.19306	Passed
Runs	0.654	0.24879	Passed
Serial	0.452	0.24198	Passed
Universal statistical	0.845	0.41029	Passed
Lempel-Ziv Compression Test	0.923	0.35497	Passed

9. RESULTS ANALYSIS

The proposed method has been developed for generating the random numbers. These numbers can be used as keys in different encryption methods and hence, this method has been used to manage and generate random keys. The main purpose of this research has been to develop a method to generate pseudorandom numbers based on mouse motion and the main research question is whether or not the random numbers that have been generated via a 2D chaotic system will be redundant or not. To address this question, initial numbers in the chaotic system are generated based on mouse motion, so that redundancy does not occur. To this end, normally, the direction of mouse motion will not repeat, because it is dependent on the direction of x,y and thus, the second mouse moving throughout *x,y* points will be generating a new one.

That is, it will not go through the same previous *x,y* points. Because of this, every time when the mouse moves in a new direction, the numbers that have been generated based on the x,y points are increased too, which will make it extremely difficult to repeat the same set of x,y points that was generated before. Hence, the proposed system will work strongly due to the continually generated new different x,y points without any redundant numbers. Table 1 illustrates the results which are proving that the redundant in values of mouse motion is very hard. The overall outcomes indicate high success in the five tests. Furthermore, the value of the result in each step did not reach that of the threshold (less than the Max number which is illustrated in each test in Table 1). Therefore, all tests were thoroughly successful. Thereby, each test had progressed smoothly all the way to the end, because had any failure occurred during any step in the test, it would have meant that the test and the subsequent ones would have indefinitely stopped.

The second step in this research was to select a set of at least 150 x,y points generated in the first step based on mouse motion to be used as pseudorandom numbers in the chaotic system. This set of x,y points will lead to the system working at least 150 times (depending on the set of values generated by the mouse motion). The set of values are representing the pseudorandom numbers that will be generating irredundant keys. With reference to this, a slight change in the pseudorandom numbers will lead to numerous changes in other results (give different results) due to the mouse is going through at least 150 x,y points. Firstly, the first eight float digits that have been generated from the chaotic system based on the first x,y mouse motion are selected. Then, those eight digits are converted into the integer number, which is subsequently converted and represented as a binary figure. Finally, the digits are reordered based on the initial permutation (IP) table to obtain a new number. Going through the same procedure will lead to the generation of new eight digit numbers in each iteration. To sum up, Table 2 illustrates the measurements of the random numbers generated by IP. The analysis of the results shown in table 2 has led to an important consideration, which is the fact that the AFC results must be very close to 0 to give a significant indicator about the randomization. Moreover, table 3 illustrates the results of the NIST tests. Clearly, the proposed system outcomes are better than the others by double or more in 15 out of 16 tests. That is, only one test gave a lesser result than the compared study, which was the “Linear complexity test”, but it still passed.

10. CONCLUSION

An innovative design for a PRNG has been proposed in which the problem of repeated keys has been solved. As a non-deterministic source, the mouse input device is used to generate the initial random numbers, being an inexpensive, convenient, and universally accessible device. This approach for generating values is entirely unpredictable and uncorrelated. The second aspect of leveraging a 2D chaotic system increases the complexity and randomness of the produced key. Taking the advantages of chaotic behaviour and the simplicity of using the mouse are the primary motivations for this proposed method. Next, the maximum periods of a massive random binary sequence of numbers were generated. Five standard statistical tests and ACF were applied to the generated sequences to give a guarantee that the acceptable properties of the random binary sequences can be used efficiently in the designing of a cryptography system as well as to increase its robustness. All the values gathered from the ACF computations were close to zero. Furthermore, the NIST involved 16 different tests being applied, and the results when compared with another study show that the current research obtained substantially better results.

REFERENCES

[1] E. R. Arboleda, J. L. Balaba, and J. C. L. Espineli, “Chaotic rivest-shamir-adlerman algorithm with data encryption standard scheduling,” *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 6, no. 3, pp. 219-227, 2010, doi: 10.11591/eei.v6i3.627.

[2] O. Z. Akif, G. J. Rodgers and H. S. Al-Raweshidy, "Protecting a sensitive dataset using a time based password in big data," *2017 Computing Conference*, London, UK, 2017, pp. 871-879, doi: 10.1109/SAI.2017.8252197.

[3] S. A. Mahmood, K. A. Hussein, Y. N. Jurn, and E. A. Albahrani, “Parallelizable cipher of color image based on two-dimensional chaotic system,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 1, pp. 101-111, 2019, doi: 10.11591/ijeecs.v18.i1.pp101-111.

[4] Y. Naseer, D. Shah, and T. Shah, “A novel approach to improve multimedia security utilizing 3D mixed chaotic map,” *Microprocessors and Microsystems*, vol. 65, pp. 1-6, 2019, doi: 10.1016/j.micpro.2018.12.003.

[5] T. Sadeeq, A. K. Farhan, and S. A. Hassan, “A Proposed Public Key Encryption Based on Hybrid Chaotic Maps *QALAAI ZANIST SCIENTIFIC JOURNAL*, vol. 2, no. 2, pp. 64-71, 2017, doi: 10.25212/lfu.qzj.2.2.08

[6] A. T. Sadiq, E. Mostafa, Yasser F. Mahmoud, A. B. Majeed, “Using Chaotic Maps To Enhance Rsa Public Key Cryptography,” 2018.

[7] M. H. Al Hasani and K. A. Al Naimee, “Impact security enhancement in chaotic quantum cryptography,” *Optics & Laser Technology*, vol. 119, p. 105575, 2019, doi: 10.1016/j.optlastec.2019.105575.

[8] S. Vaidyanathan, A. Sambas, S. Zhang, Y. Zeng, M. A. Mohamed, and M. Mamat, “A new two-scroll chaotic system with two nonlinearities: dynamical analysis and circuit simulation,” *TELKOMNIKA (Telecommunication, Computing, Electronics and Control*, vol. 17, no. 5, pp. 2465-2474, 2019, doi: 10.12928/TELKOMNIKA.v17i5.10650.

[9] Chunyan Han, “An image encryption algorithm based on modified logistic chaotic map,” *Optik*, vol. 181, no. December 2018, pp. 779-785, 2019, doi: 10.1016/j.ijleo.2018.12.178.

[10] W. Feng, Y. G. He, H. M. Li, and C. L. Li, “Cryptanalysis of the integrated chaotic systems based image encryption algorithm,” *Optik*, vol. 186, pp. 449-457, 2019, doi: 10.1016/j.ijleo.2018.12.103.

[11] Z. Tang and S. Yu, "Design and realization of digital image encryption and decryption based on multi-wing butterfly chaotic attractors," *2012 5th International Congress on Image and Signal Processing*, Chongqing, China, 2012, pp. 1143-1147, doi: 10.1109/CISP.2012.6469744.

[12] L. Liu and S. Miao, “A new simple one-dimensional chaotic map and its application for image encryption,” *Multimedia Tools and Applications*, vol. 77, no. 16, pp. 21445-21462, 2018, doi: 10.1007/s11042-017-5594-9.

[13] Y. Zhao, C. Gao, J. Liu, and S. Dong, “A Self-perturbed Pseudo-random Sequence Generator Based on Hyperchaos,” *Chaos, Solitons Fractals X*, vol. 4, p. 100023, 2020, doi: 10.1016/j.csfx.2020.100023.

[14] W. A. Hussein, N. M. G. Al-Saidi and H. Natiq, "A New 2D Hénon-Logistic Map for Producing Hyperchaotic Behavior," *2018 Third Scientific Conference of Electrical Engineering (SCEE)*, Baghdad, Iraq, 2018, pp. 265-269, doi: 10.1109/SCEE.2018.8684083.

[15] İ. Öztürk and R. Kılıç, “A novel method for producing pseudo random numbers from differential equation-based chaotic systems,” *Nonlinear Dynamics*, vol. 80, no. 3, pp. 1147-1157, 2015, doi: 10.1007/s11071-015-1932-5.

[16] K. R. Radhika and M. K. Nalini, "Biometric Image Encryption Using DNA Sequences and Chaotic Systems," *2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, Bangalore, 2017, pp. 164-168, doi: 10.1109/ICRAECT.2017.56.

[17] Kadhim F. and H. Emad M., “Mouse Movement with 3D Chaotic Logistic Maps to Generate Random Numbers,” *Diyala Journal for Pure Sciences*, vol. 13, no. 3, pp. 24-39, 2017, doi: 10.24237/djps.1303.268B.

[18] H. A. Abdullah and H. N. Abdullah, “FPGA implementation of color image encryption using a new chaotic map,” *Indones. J Electr Eng Comput Sci*, vol. 13, no. 1, pp. 129-137, 2019, doi: 0.11591/ijeecs.v13.i1.pp129-137.

[19] O. Datcu, C. Macovei, and R. Hobincu, “Chaos based cryptographic pseudo-random number generator template with dynamic state change,” *Applied Sciences*, vol. 10, no. 2, p. 451, 2020, doi: 10.3390/app10020451.

[20] S. Dhall, S. K. Pal, and K. Sharma, “A chaos-based probabilistic block cipher for image encryption,” *Journal of King Saud University - Computer and Information Sciences*, 2018.

[21] F. H. Hsiao, “Chaotic synchronization cryptosystems combined with RSA encryption algorithm,” *Fuzzy Sets Syst Fuzzy Sets and Systems*, vol. 342, pp. 109-137, 2018, doi: 10.1016/j.fss.2017.10.016.

[22] Rezk, A. H. Madian, A. G. Radwan, and A. M. Soliman, “Reconfigurable chaotic pseudo random number generator based on FPGA,” *AEU - International Journal of Electronics and Communications*, vol. 98, pp. 174-180, 2019, doi: 10.1016/j.aeue.2018.10.024.

[23] R. Lan, J. He, S. Wang, T. Gu, and X. Luo, “Integrated chaotic systems for image encryption,” *Signal Processing*, vol. 147, pp. 133-145, 2018, doi: 10.1016/j.sigpro.2018.01.026.

[24] M. Wang, X. Wang, Y. Zhang, and Z. Gao, “A novel chaotic encryption scheme based on image segmentation and multiple diffusion models,” *Optics & Laser Technology*, vol. 108, pp. 558-573, 2018, doi: 10.1016/j.optlastec.2018.07.052.

[25] M. Ababneh, “A new four-dimensional chaotic attractor,” *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 1849-1854, 2018, doi: 10.1016/j.asej.2016.08.020.

[26] M. A. Riyadi, M. R. A. Khafid, N. Pandapotan and T. Prakoso, "A Secure Voice Channel using Chaotic Cryptography Algorithm," *2018 International Conference on Electrical Engineering and Computer Science (ICECOS)*, Pangkal, Indonesia, 2018, pp. 141-146, doi: 10.1109/ICECOS.2018.8605229.

[27] Q. Zhou, X. Liao, K. wo Wong, Y. Hu, and D. Xiao, “True random number generator based on mouse movement and chaotic hash function,” *Information Sciences.*, vol. 179, no. 19, pp. 3442-3450, 2009, doi: 10.1016/j.ins.2009.06.005.

[28] A. Mostafa, N. F. Soliman, M. Abdalluh and F. E. Abd El-samie, "Speech encryption using two dimensional chaotic maps," 2015 11th International Computer Engineering Conference (ICENCO), Cairo, Egypt, 2015, pp. 235-240, doi: 10.1109/ICENCO.2015.7416354.

[29] A. Rukhin, J. Soto, and J. Nechvatal, “A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications,” 2010.

[30] L. Dong, Z. Yong, L. Ji, and X. Han, “Study on the pass rate of NIST SP800-22 statistical test suite,” in

Proceedings-2014 10th International Conference on Computational Intelligence and Security, CIS 2014, 2015, pp. 402–404.

[31] X. Huang, L. Liu, X. Li, M. Yu, and Z. Wu, “A new two-dimensional mutual coupled logistic map and its application for pseudorandom number generator,” *Mathematical Problems in Enineering*, vol. 2019, 2019, doi: 10.1155/2019/7685359.

BIOGRAPHIES OF AUTHORS

Omar Z. Akif received an MSc. degree in Computer Science (Informatic), institute for post graduate studies, from the Iraqi commission for computers and informatics (Iraq) in 2005, and a Ph.D. degree in Electronic and Computer Engineering, College of Engineering, Design and Physical Sciences from Brunel University, London, United Kingdom in 2018. He has engaged in research including computer security and network security. He is currently a lecturer at the Department of Computer Science, College of Education for Pure Science (Ibn al-Haitham), University of Baghdad.

Sura Mazin Ali Political Science College, Al Mustansiriyah University, Baghdad, Iraq. She has a BSc. in Software Engineering from Al Rafidin College (2001) and an MSc. in Computer Science (2006). She has had several papers published on Data Security & Artificial Intelligence. At present, she is the Director of the Computer Laboratory at her college.

Rasha S. Ali received a BSc. degree in Artificial Intelligence from the Department of Computer Sciences, University of Technology, in 2008, an MSc. degree in computer science (data mining) from the Department of Computer Sciences, College of Science, Baghdad University, in 2013, and a Ph.D. degree in computer science (secure retrieval from big encrypted data) from the Department of Computer Sciences, University of Technology, in 2017. In 2017, she joined the Department of Computer Sciences, Al-Turath University College, whilst in 2018, she joined Al-Nisour University College as an academic staff member. She has been the author of numerous technical articles since 2013. Her research interests include intelligent systems, privacy and security, management of big data, data mining and database applications, chaos theory, and cloud computing.

Alaa K. Farhan is Professor in the Department of Computer Sciences, University of Technology-Baghdad-Iraq. He completed his Bachelor of Computer Science and Master of Science degrees in information security at the Department of computer Sciences-University of Technology, Baghdad, Iraq, in 2003, and 2005, respectively. He received his Ph.D. degree in information security from the University of Technology, Baghdad, Iraq, in 2009. In 2005, he joined the Department of Computer Sciences, University of Technology, as an academic staff member. Assist. Prof. Dr. Alaa is the author of numerous technical papers since 2008, with his research interests including: cryptography, programming languages, chaos theory and cloud computing.