Performance and Statistical Analysis of Chaotic Random Bit Generator

S. M. Seen Mohamed Aliar Maraikkayar*, K. Rajakumar, R. Tamilselvi, M. Parisa Beham, A. Sabah Afroze
Department of CSE, Sethu Institute of Technology, Virudhunagar, Tamil Nadu, India.

* Corresponding author. Tel: 8883925991; email: seenimohamedali@sethu.ac.in
Manuscript submitted January 3, 2019; accepted March 16, 2019.

Abstract: Medical Information transmitted through the internet and secured against any attacks is an international challenging fear. In the present scenario, a fabulous relation emerges between chaos and cryptography. Various features of chaotic systems such as initial state sensitivity, ergodicity, mixing properties and complexity in the structure produce deterministic pseudo randomness in the input data. Chaotic Random Bit Generator (CRBG) makes the bit sequence unpredictable by an intruder, in the field of medical research. In current years, mixture of chaos-based cryptosystems have been projected. To be used in medical field, a CRBG may require in meeting stronger desires than for any other applications. Motivated by all these issues, a novel chaotic random bit generator is proposed based on two different chaotic based logistic maps in parallel and with preliminary self-determining initial conditions. The random bit sequence which is chaotic in character is created by predicting the outputs of both the chaotic logistic maps. Also it is projected to put forward dissimilar tests by stressing some of its alluring arithmetic features, which make it an ideal preference for the expected random bit generation. Lastly, the results of all the statistical tests generated bit sequences, is tested under all the most powerful NIST suit tests for the prediction of randomness: The tests validate the exact expected uniqueness expected of real random sequences.

Key Words: Random bit generator, chaotic map, NIST tests, randomness.

1. Introduction

Very speedy growth in the communication era, especially the emerging networks have lengthened the data broadcasting area, which in turn present novel confront for defensiving the information from unlawful snooping. This rooted to explore the knowledge in challenges in the region of cryptography to accomplish the sturdy drive of new sheltered cryptographic methods. Due to the existence of this nearby relationship between chaos and cryptography, plenty chaos-based cryptosystems have been put forward since 1990. Chaos-based cryptosystems diverges in to analog and digital types. Analog chaos-based cryptosystems are occupying the organization and coordination of chaos. In Digital chaos-based cryptosystems, chaotic dissimilar dynamical systems are executed in incomplete computing accuracy.

Digital chaos-based cryptosystems are explained through slab ciphers based on and/or knock over iterations of chaotic maps, block ciphers based on baffled encompassing functions, stream ciphers forward employing chaos-based arbitrary bit generators etc. [1]. A random bit creator is a mechanism or a protocol which produces a continuous algebraically autonomous and unbiased double digits. In realistic situation, a
design based device or software programme is used to take benefit of the accepted basis of uncertainty and formulate a bit succession free from biases and the association is a complicated task. In such scenario, the predicament can be eradicated by replacing a haphazard bit creator with a mystified arbitrary bit generator. [2]. Encryption quality mainly depends on the success of the random bit generator developed. There are quite a lot of methods used for authenticating the arbitrарiness of the proposed CRBG. The various standards used for testing the sequences are: (i) National Institute of Standards and Technology (NIST) suite of arithmetical methods, (ii) The DIEHARD suite of numerical tests, (iii) The Crypt-XS suite of geometric method, and (iv) The Donald Knuth’s arithmetical tests. NIST algebraic methods group wrap a sufficient number of practically autonomous numerical methods, which recognize any discrepancy in the uncertainty. Hence for analyzing the unpredictability nature of the CRBG, the NIST test suite are used [3]. Yu et al [4] proposed a dual image encryption algorithm based on standard Data Encryption algorithm (S-DES) and chaotic map. The plain text and cryptograph block were both 8 bits, execution speed was very fast but could not meet the security requirements. So the chaotic map is combined with the S-DES system to enhance the security by using the sensibility in the original value and randomness in the chaotic map. Their result showed that the encryption took one second for encrypting the image. The computing speed is quick and escapes from the exhaustion attacks. But the algorithm does not deal with any parameters related to the security level. Also it does not show any results with S-DES alone. Liu et al. [5] explained a new chaotic scrambling algorithm. Their algorithm involved the dynamic twice interval division and had larger key space. The method involved the one dimensional logistic maps to generate the chaotic sequence of real numbers. The number of iterations should also be less to speed up the operation. This is the drawback of their algorithm. The organization of the paper is as follows: Section 1 deals with the introduction of the random generators, concept of logistic map and the statistical tests used for the validation of randomness of the sequences. Section 2 explains the proposed methodology based on the Chaotic Random Bit Generator and the uniqueness of the proposed work. Section 3 explains the statistical tests and their procedure in evaluating the randomness of the sequences. Section 4 involves the testing strategy to validate the sequences in randomness and the Statistical Analysis of the Randomness of the Chaotic Sequences. Section 4 involves the results and discussion of the sample sequence randomness and their validation. Section 5 concludes the unique features of the proposed Random Bit generator and the validation of the generated sequences.

2. Proposed Methodology

Based on the above surveys for random bit generation, a novel chaotic map based on the logistic map is defined for the creation of arbitrary sequences. The proposed chaotic random bit generator is shown in Fig. 1. The logistic map is a much uncomplicated geometric duplication frequently used to reveal the escalation of biological populations. Because of its geometric straightforwardness, this mock-up persists to be helpful test bed for innovative thoughts in bedlam theory as well as importance of confusion in cryptography. A CRBG is developed which depends on two logistic maps, early from arbitrary independent preliminary situation $(X_0, Y_0 \in (0,1)$ and $X_0 \neq Y_0)$.

$$X_{n+1} = \lambda X_n (1 - X_n) \quad (1)$$

$$Y_{n+1} = \lambda Y_n (1 - Y_n) \quad (2)$$

The outputs of the logistic maps are compared with their respective median in the decision devices. Because an important observation is made that the median is the most suitable statistical characteristic that may split the domain into two equally filled sub-domains and in this way randomness is achieved. Because of this statement, the criterion for the final output of the logistic map is as follows:
Here median1 denotes the median of the values generated by the logistic map 1. Median2 denotes the median of the values generated by the logistic map 2. \( a_k \) and \( b_k \) are the outputs from the two decision devices. The sequences are grouped as 10 bits in all the sequences. First 10 bits represent the first sequence (0-9), then from 10-19 bits as the second sequences and so on.

The hexadecimal equivalent values for the corresponding 10 bits are calculated for both the 10 bit sequence values. The final bit sequence is generated by sending the outputs of the decision devices in hexadecimal value to the logical device which performs logical Exclusive–OR (EXOR) operation between two sequences. The generated random sequences used as a key for encryption and the encrypted image is not attacked by any statistical or differential attacks, as the level of the encryption quality and entropy gets increased with the increase in the number of blocks. The generated sequences randomness level is increased when compared with our previous work [6]. The chaotic sequence generated by the Random Generator is tested under various statistical tests and the randomness is evaluated.

3. **Statistical Tests**

The NIST test suite is an algebraic wrap up encompassing of 16 different tests. The random sequences which are generated by either software algorithms or hardware designs are based on cryptographic random based generators. NIST suite tests are broadly classified into two groups, (i) Non-parameterized tests and (ii) Parameterized tests [3].

### 3.1. Non-parameterized Tests

#### 3.1.1 Frequency (monobit) test

The goal of this non-parameterized test is to locate the percentage of quantity of zeroes and ones in the whole arbitrary values.

1. The input sequence is considered as \( ε \).
2. The test statistic \( S_{obs} \) is calculated as
   \[
   S_{obs} = \left| \frac{S_n}{\sqrt{n}} \right|
   \]
3. Next, the P-value is calculated using
   \[
   P = \frac{erfc\left( \frac{S_{obs}}{\sqrt{2}} \right)}{2} \tag{4}
   \]
where \( erfc \) is the complementary error function

4. Decision Rule based on the 1% stage defines that if the calculated P-value is < 0.01, then the conclusion is that the generated sequence is non-random.

#### 3.1.2. Runs test

The survival of total figure of runs in the series is sketched by this test. Run is an endless series of apt bits.
(1) The pre-test proportion is denoted as $\pi$ of ones in the input binary sequence and is computed as

$$\pi = \frac{\sum \epsilon_i}{n}$$  \hspace{1cm} (5)

(2) The next procedure is, if $\pi - 1/2 \geq \tau - 1$, where $\tau = 2/\sqrt{n}$, then there is no need for the runs test. If the test is not apt, then the P-value is well thought-out as 0.0000.

(3) The test statistic $V_n (obse)$ estimated as:

$$V_n (obs) = \sum_{k=1}^{n} r(k) + 1$$  \hspace{1cm} (6)

where $r(k) = 0$ if $\epsilon_k = \epsilon_{k+1}$, and $r(k) = 1$ otherwise.

(4) Then the P-value is computed as:

$$erfc \left( \frac{|V_n (obs) - 2n\pi (1-\pi)|}{2\sqrt{2n\pi(1-\pi)}} \right)$$  \hspace{1cm} (7)

where $erfc$ is the complementary error function in which the values are estimated from predefined complementary error function table.

### 3.1.3. Discrete fourier transform (spectral) test

This test estimates the acme elevation in the Discrete Fourier Transform of the estimated sequence.

(1) The zeros and ones of the input sequence are considered as $\epsilon$.

(2) Discrete Fourier Transform (DFT) is applied on to the generated sequence to produce a value represented as $S, S = DFT(X)$.

(3) Then $M$ is estimated as $M = \text{modulus} \left( S' \right) \equiv |S'|$, where $S'$ is the sub group consisting of the initial $n/2$ elements in $S$, and the modulus role fabricates a succession of acme heights.

(4) Estimate the $T$ value

$$T = \left( \log \frac{1}{0.05} \right)^*$$  \hspace{1cm} (8)

$n$ is considered to be the 95 % of peak height threshold value.

(5) Estimate $N_0$, where

$$N_0 = 0.95 \times n/2.$$  \hspace{1cm} (9)

$N_0$ is the customary hypothetical (95%) number of peaks (under the presumption of arbitrariness) that are a lesser than amount of $T$.

(6) Then, Compute $N_1$, which is the actual pragmatic integer of acme in $M$ that are lesser than amount of $T$.

(7) Then, $d$ value is calculated as:

$$d = \left( \frac{(N_1 - N_0)}{\sqrt{n(0.95)(0.05)/4}} \right)$$  \hspace{1cm} (10)

(8) Then $P$-value is estimated based on

$$erfc \left( \frac{|d|}{\sqrt{2}} \right)$$  \hspace{1cm} (11)

(9) Decision Rule based on the 1% stage defines that if the calculated $P$-value is < 0.01, then the conclusion is that the generated sequence is non-random.
3.2. Parameterized Tests

3.2.1. Frequency test within a block

The test starts with the severance of $M$-bit slabs and finding number of ones in the proposed block.

1. The input sequence is divided as $N$,

$$N = \left\lfloor \frac{n}{M} \right\rfloor$$

$n$ – No. of non-overlapping blocks.

2. The quantity $\pi_i$ of ones in every $M$-bit block is estimated as

$$\pi_i = \frac{\sum_{j=1}^{M} e((i-1)M + j)}{M}$$

for $1 \leq i \leq N$.

3. The $\chi^2$ statistic $\chi^2(\text{obs})$ is calculated as:

$$\chi^2(\text{obs}) = 4M \sum_{i=1}^{N} (\pi_i - 1/2)^2$$

4. $P$-value = $\text{igamc}(N/2, \chi^2(\text{obs})/2)$ is computed, where $\text{igamc}$ is the incomplete gamma function and the values are intended from the existing equations. Same decision rule is followed.

3.2.2. Serial test

The serial test predicts the happening of all credible overlying $m$-bit sub groups across the whole sequence.

1. An augmented sequence $\varepsilon'$ is explained by appending the initial $m-1$ bits to the concluding value of the sequence for distinct values of $n$.

2. The occurrence of all promising overlying $m$-bit slabs, all probable overlying ($m-1$)-bit blocks and all probable overlying ($m-2$)-bit slabs are estimated.

Let $v_{i_1...i_m}$ represents the happening of the $m$ bit design $i_1...i_m$ and $v_{i_1...i_{m-1}}$ denote the happening of the $(m-1)$-bit design $i_1...i_{m-1}$; and let $v_{i_1...i_{m-2}}$ denote the prevalence of the $(m-2)$-bit design $i_1...i_{m-2}$.

3. Compute:

$$\phi_m^2 = \frac{2^m}{n} \sum_{i_1...i_m} (v_{i_1...i_m} - \frac{n}{2^m})^2 = \frac{2^m}{n} \sum_{i_1...i_m} y_{i_1...i_m}^2 - n$$

$$\phi_{m-1}^2 = \frac{2^{m-1}}{n} \sum_{i_1...i_{m-1}} (v_{i_1...i_{m-1}} - \frac{n}{2^{m-1}})^2 = \frac{2^{m-1}}{n} \sum_{i_1...i_{m-1}} y_{i_1...i_{m-1}}^2 - n$$

$$\phi_{m-2}^2 = \frac{2^{m-2}}{n} \sum_{i_1...i_{m-2}} (v_{i_1...i_{m-2}} - \frac{n}{2^{m-2}})^2 = \frac{2^{m-2}}{n} \sum_{i_1...i_{m-2}} y_{i_1...i_{m-2}}^2 - n$$

4. Compute

$$\nabla \phi_m^2 = \phi_m^2 - \phi_{m-1}^2, \text{ and}$$

$$\nabla^2 \phi_m^2 = \phi_m^2 - 2\phi_{m-1}^2 + \phi_{m-2}^2$$

5. Compute

$$P\text{-value}_1 = \text{igamc}(2^{m-2}, \nabla \psi_m^2)$$

$$P\text{-value}_2 = \text{igamc}(2^{m-3}, \nabla^2 \psi_m^2)$$
3.2.3. Approximate entropy test

The test contrast the happening of overlying slabs of two uninterrupted/neighbouring lengths \((m\) and \((m+1)\)) against the conventional value for an arbitrary sequence.

1. \(n\) overlapping \(m\)-bit sequences are created by appending \(n\)-bit design to \(m-1\) bits from the beginning of the sequence to the finishing end of the sequence.

2. \(n\) overlapping blocks denotes the frequency count. Tally of the \(m\)-bit \(((m+1)\)-bit) values is represented as \(C_i^m\), where \(i\) is the \(m\)-bit value.

\[
C_i^m = \frac{\#i}{n} \quad \text{for each value of } i.
\] (18)

4. Compute

\[
\phi^{(m)} = \sum_{i=0}^{2^m-1} \pi_i \log \pi_i
\] (19)

5. The above steps are again repeated by changing \(m\) by \(m+1\).

6. The test statistics are computed by

\[
\chi^2 = 2n[\log 2 - \text{ApEn}(m)],
\] (20)

Same Decision rule is followed.

4. Statistical Analysis of the Randomness of the Chaotic Sequences

Among the various statistical tests, the NIST test depends on the hypothesis testing. In this proposed work, the test validate whether the generated sequence of zeroes and ones is a chance represented as unacceptable (null Hypothesis \(H_0\)). For each test, an applicable randomness value is chosen and used to resolve the acceptance or rejection of the \(H_0\) hypothesis. A significance value \((\alpha)\) be preferred for the tests and if \(P\)-value \(\geq \alpha\), then the unacceptable hypothesis is acknowledged, i.e., the succession is determined to be random. If \(P\)-value \(< \alpha\), then the unacceptable hypothesis is discarded, i.e., the succession is determined to be non-random. In practical scenario, the significance level \((\alpha)\) is selected in the intermission \([0.001, 0.01]\). The \(\alpha = 0.01\) designates that one would anticipate one sequence out of 100 sequences to be discarded. For the numerical experimentations on the developed CRBG 100 sequences are taken (sample size \(m = 100\)) and each progression has been initialized from a randomly selected seed, \((X_0, Y_0 \in (0, 1)\) and \(\lambda_1 = \lambda_2 = 4\)). The series of satisfactory proportion is computed by using the confidence interval defined by

\[
\hat{p} \pm 3\sqrt{\frac{\hat{p}(1-p)}{m}},
\] (21)

where \(m\) is the sample size and \(\hat{p} = 1 - \alpha\), which are \(m = 100\) and \(\hat{p} = 1 - \alpha = 0.99\) for the present analysis. So the series of satisfactory proportion is \([0.9303, 1.0497]\).

5. Results and Discussion

A Sample Result of CRBG is considered as \([0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1]\).

5.1. Monobit Test

According to the test description, in the given sequence zeros and ones are converted in to -1 and +1 and added to produce S. Since the \(P\)-value is 1, the given sequence is random.

5.2. Runs Test

According to the test description, \(\pi=0.5\) and \(\pi-1/2\) is less than \(\tau\), where \(\tau = 2/\sqrt{n}\). \(V_{obs}=1\) and
\( P_{\text{valruntest}} = 0.5271 \). Hence the given sequence is random.

### 5.3. Discrete Fourier Transform Test

Calculated \( P \)-value is 0.4682. So the given sequence is random.

Frequency Test within a Block

According to the test description, \( \chi^2_{\text{obs}} = 4 \sum_{i=1}^{N} (\pi_i - 1/2)^2 = 3.6667 \) and \( P_{\text{valreq}} = 0.4957 \).

### 5.4. Serial Test

According to the test description, \( \psi^2_m = 4.40, \psi^2_{m-1} = 2.4 \) and \( \psi^2_{m-2} = 0, \nabla \psi^2_m = 2, \nabla^2 \psi^2_m = -4.0 \), \( P_{\text{val1}} = 0.8647 \) and \( P_{\text{val2}} = 0.9476 \).

### 5.5. Approximate Entropy Test

According to the test description, \( \varphi(m) = -1.7481, \Phi(m+1) = -1.1043, ApEn(m) = 0.6438, \chi^2 = 0.9874 \) and \( P_{\text{valent}} = 0.9954 \). If the computed \( P \)-value is < 0.01, then the sequence is concluded as non-random. 100 samples of the sequences are tested and their results for various tests are tabulated in Table 1, Table 2 and Table 3.

<table>
<thead>
<tr>
<th>S. no</th>
<th>Random Number Generated</th>
<th>Monobit Test P-val</th>
<th>Runs Test (P-val)</th>
<th>DFT Spectral Test (P-val)</th>
<th>Freq Test within a block (P-val)</th>
<th>Serial Test (P-val)</th>
<th>Approx Entropy Test (P-val)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00000001101</td>
<td>0.2059</td>
<td>0.8803</td>
<td>0.4682</td>
<td>0.1624</td>
<td>0.5940</td>
<td>0.1119</td>
</tr>
<tr>
<td>2</td>
<td>0000100100</td>
<td>0.0578</td>
<td>0.0753</td>
<td>0.4682</td>
<td>0.4957</td>
<td>0.4784</td>
<td>0.7071</td>
</tr>
<tr>
<td>3</td>
<td>1001000010</td>
<td>0.2059</td>
<td>0.1753</td>
<td>0.4682</td>
<td>0.9157</td>
<td>0.5940</td>
<td>0.8785</td>
</tr>
<tr>
<td>4</td>
<td>0000000010</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.5542</td>
</tr>
<tr>
<td>5</td>
<td>1000000000</td>
<td>0.0114</td>
<td>0.7253</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.5542</td>
</tr>
<tr>
<td>6</td>
<td>1000010000</td>
<td>0.2059</td>
<td>0.5469</td>
<td>0.4682</td>
<td>0.4959</td>
<td>0.0200</td>
<td>0.6404</td>
</tr>
<tr>
<td>7</td>
<td>0000001000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>8</td>
<td>0001000000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>9</td>
<td>0000000101</td>
<td>0.0578</td>
<td>0.4292</td>
<td>0.4682</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.3115</td>
</tr>
<tr>
<td>10</td>
<td>0000000100</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>11</td>
<td>1000000000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>12</td>
<td>0000010000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>13</td>
<td>0000100000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>14</td>
<td>0001000000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>15</td>
<td>0010000000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>16</td>
<td>0100000000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
<tr>
<td>17</td>
<td>0010000000</td>
<td>0.0114</td>
<td>0.0350</td>
<td>0.0295</td>
<td>0.1624</td>
<td>0.0200</td>
<td>0.6752</td>
</tr>
</tbody>
</table>
Table 2. Non-parameterized Tests Results

<table>
<thead>
<tr>
<th>S. No</th>
<th>Statistical test</th>
<th>No. of sequences with P-Values ≥ 0.01 (Success) (m_p)</th>
<th>No. of sequences with P-Values &lt; 0.01 (Failure) (m_f)</th>
<th>Proportion of sequences passing the test (m_p / m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Frequency (monobit) test</td>
<td>97</td>
<td>3</td>
<td>0.97</td>
</tr>
<tr>
<td>2.</td>
<td>Runs test</td>
<td>94</td>
<td>6</td>
<td>0.96</td>
</tr>
<tr>
<td>3.</td>
<td>DFT (spectral) test</td>
<td>100</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3. Parameterized Tests Results

<table>
<thead>
<tr>
<th>S. No</th>
<th>Statistical test</th>
<th>No. of sequences with P-Values ≥ 0.01 (Success) (m_p)</th>
<th>No. of sequences with P-Values &lt; 0.01 (Failure) (m_f)</th>
<th>Proportion of sequences passing the test (m_p / m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Frequency test within a block</td>
<td>100</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2.</td>
<td>Serial tests</td>
<td>96</td>
<td>4</td>
<td>0.96</td>
</tr>
<tr>
<td>3.</td>
<td>Approximate entropy test</td>
<td>100</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

The various tests carried out to prove the randomness of the data clearly show that the computed proportion for each test lies inside the confidence interval. Hence, the tested binary sequences generated by the PRBG are random. The number of samples used for the test is 100 samples.

Among the tests carried out in non-parameterized test, runs test gives 96% success of the randomness and remaining two test such as frequency test and spectral test 97% and 100% result in randomness. The
comparative result analysis is shown in Fig. 2 and Fig. 3 for Non-parameterized tests and parameterized tests respectively. The two tests such as runs test and serial test failed in the same sequence, since both the tests depend on occurrence of overlapping m bit patterns. If the m bit has long runs of zeros, the test gives failure in random nature of bits. Due to this reason only, the same sequence failed in both the tests and give similar results. In parameterized test, serial test gives 96% result in randomness and 100% result in frequency test and approximate entropy test. Our previous method in non-parameterized test, runs test gives 94% success of the randomness and remaining two test such as frequency test and spectral test 96% and 98% result in randomness. In parameterized test, serial test gives 94% result in randomness and 96% result in frequency test and approximate entropy test. The result shows the better randomness of the CRBG with our previous work [6]. The experimental result shows that the generated sequences are truly random bits and pass through all the tests and produce high randomness. For simplification the length of the bit taken is 10 bits and significance level is 0.01. These random data are used in encryption methods to increase the security level.

Fig. 2. Randomness comparison of the generated sequences in non-parameterized tests.

Fig. 3. Randomness comparison of the generated sequences in parameterized tests.

6. Conclusion

This paper was concluded that the statistical testing procedure and various tests carried out to prove the randomness of the generated sequences. We introduced a design of a pseudo random bit generator (PRBG) based on two chaotic logistic maps from independent initial conditions in our previous work. The pseudo random bit sequence is obtained by evaluating the outputs of both the chaotic logistic maps. We have also tested meticulously the generated binary sequences using the NIST suite, consisting of various statistical
tests developed to perceive the specific features of anticipated random bit sequences. The results of statistical testing are cheering and showed that the proposed PRBG has ideal cryptographic properties and better used in the design of new stream ciphers.

References


S. M. Seeni Mohamed Aliar Maraikkayar is pursuing the M.E. in computer science and engineering, in Anna University Chennai, India. He did his bachelor’s in engineering at Sethu Institute of Technology and his master’s in business administration at Coventry University, London, UK. He has published papers in national and international journal and conferences. His research interest includes biomedical signal and image processing.