A new design of dynamic S-Box based on two chaotic maps.

Ghada Zaïbi\(^{(a)}\), Fabrice Peyrard\(^{(b)}\), Abdennaceur Kachouri\(^{(a)}\), Danièle Fournier-Prunaret\(^{(c)}\), Mounir Samet\(^{(a)}\)

\(^{(a)}\)GE dept. National school of engineering of Sfax, ENIS, Sfax, Tunisia  
ghada.zaibi@ieee.org, abdennaceur.kachouri@enis.rnu.tn, mounir.samet@enis.rnu.tn  
\(^{(b)}\)IRIT CNRS, Toulouse, France  
Fabrice.Peyrard@irit.fr  
\(^{(c)}\)LATTIS, INSA, Toulouse, France  
Daniele.Fournier@insa-toulouse.fr

Abstract— The use of chaotic maps to construct dynamic S-Box has been increasingly studied. The aim of this paper is to propose a new design of dynamic chaotic S-Box that enhances the security criteria of the block ciphers. In fact, our S-Box is based on the combination of two chaotic maps: one and three dimensional piecewise linear maps. The randomness of these maps was verified by using the NIST test package. However, the conversion method seems to be crucial to keep the randomness and the uniformity of the binary sequences. The security analysis shows that the dynamic S-Boxes based on two chaotic maps have the lowest linear approximation probability \(L_0=0.0625\), the closest value to 0.5 of the strict avalanche criteria \(SAC=0.4998\) and also an equiprobable input/output distribution \(DF=10/256\).

Keywords-component: S-Box; chaos; NIST test; AES; security

I. INTRODUCTION

Recently, there has been a spate of interest in using chaos in cryptography and telecommunications area. The properties of the chaotic maps such as sensibility to initial conditions appear useful to enhance the diffusion and confusion properties of block ciphers. The only nonlinear component of the conventional block ciphers is the substitution table: S-Box.

So, the idea is to use these chaotic maps to construct dynamic S-Boxes.

Many researchers have dealt with this field of interest and have proposed many approaches for obtaining S-Boxes based on chaos \([1], [2], [3], [4], [5] and [6]\).

In this paper, the NIST test package is presented and the random like behavior of the resulting chaotic sequences is studied to check their efficiency in the S-Box design.

Two methods to construct dynamic S-Boxes are presented and different security criteria such as nonlinearity, the equiprobable input/output XOR distribution and strict avalanche criterion were evaluated and compared with other existing chaotic S-Box designs.

This paper is structured in the following way:

In section II, we present the NIST test suite and the adopted ways to detect randomness of chaotic binary sequences. We propose, in Section III, our Chaotic S-Box design and a comparison of all security criteria is given. Finally a conclusion is drawn.

II. STATISTICAL TEST SUITE OF THE NIST

The NIST (National Institute of Standards and Technology) tests is a statistical package of 16 tests developed to detect the randomness of binary sequences. These sequences may be the output of cryptographic random or pseudorandom number generators \([7], [8]\). The latest version (version 2.0b, August 25, 2008, available online \([12]\)) contains only 15 tests: the Frequency (Monobit) Test, the Frequency Test within a Block, the Cumulative Sums (Cusums) Test, the Runs Test, the test for the Longest-Run-of-Ones in a Block, the Binary Matrix Rank Test, the Discrete Fourier Transform (Spectral FFT) Test, the Non-overlapping Template Matching Test, the Overlapping Template Matching Test, the Maurer’s “Universal Statistical” Test, the Approximate Entropy Test, the Random Excursions Test, the Random Excursions Variant Test, the Serial Test and finally the Linear Complexity Test.

There is no order to apply the NIST tests, but the frequency test must be applied first, because it provides the most basic evidence of the non randomness which is the non uniformity. Each test gives a resulting P-Value. The P-value represents the probability that a perfect random generator would have generated a sequence less random than the sequence that was tested \([7]\). A significance level \(\alpha\) is fixed for the next simulations: \(\alpha=0.01\). This level indicates that about 1% of the sequences are expected to fail. A sequence pass the test if the P-value is >0.01, and if P-Value is <0.01 the sequence is non random.

The NIST has adopted two ways to interpret the resulting P-Value:

- Proportion of sequences passing the test: We can compute the proportion of sequence that pass a test by using the resulting P-value. This proportion is equal to the number of sequences, having P-Value>significance level, divided by the total number of tested sequences.

- The distribution of P-Value: The review of the uniformity may be ensured by two methods, the first one is the use of a histogram. P-values are distributed on 10 sub-intervals between 0 and 1. The second one is to compute the
Chi-square \( \chi^2 = \sum_{i=1}^{10} \frac{(F_i - s/10)^2}{s/10} \), where \( F_i \) is the number of P-values in sub-interval \( i \), and \( s \) is the sample size. A new P-value is calculated such that P-value = \( \text{igamc}(9/2, \chi^2/2) \). If P-value \( \geq 0.0001 \), then we can conclude that the sequence is uniformly distributed.

In order to test the chaotic maps, we follow these steps:
1) Convert real chaotic outputs to binary values.
2) Generate sequences of a minimum of 1 million bits.
3) Apply the tests to each sequence and analyse the resulting P-Values.

Two methods are used to convert real output into binary values:

1. The IEEE single precision floating point format: each real number is represented with 32 bits: 1 signed bit, 8 exponent bits and 23 bits as mantissa.
2. Round each real number to the nearest integer. And since the real output are between 0 and 1, we obtain binary sequence.

A. Statistical tests applied on 1D map

The PWLCM (Piecewise Linear Chaotic Map) is given by this expression:

\[
X(t+1) = F_p(X(t)) = \begin{cases} 
X(t)/p & 0 \leq X(t) < p \\
(X(t) - p)/(0.5 - p) & p \leq X(t) < 0.5 \\
(1 - X(t) - p)/(0.5 - p) & 0.5 \leq X(t) < 1 - p \\
(1 - X(t))/p & 1 - p \leq X(t) \leq 1 
\end{cases}
\]

Where \( 0 < p < 1/2 \). This map is represented in Fig. 1.

The next results are presented in two parts: test of sequences generated by the round method and test of sequences generated by IEEE single precision format.

1) Conversion with round

The NIST test suite is applied to 100 binary sequences of 1 million bits minimal length. These sequences are constructed by varying initial conditions (100 initial conditions).

First of all, we must be ensured that all the sequences pass the frequency test, so we compute the P-values of the first test for different initial conditions (Fig. 2). The red line indicate the significance level \( \alpha = 0.01 \).

Among 100 only 6 sequences fail the first test, so the following initial conditions do not be used to generate binary sequences \( x_1=[0.13, 0.15, 0.17, 0.5, 0.87, 1] \).

Indeed, to obtain a good random number generator, as regards its statistical properties, we must avoid some initial conditions. This chaotic map will be used as seed for the 3D map that we propose to employ in the S-Box construction.

The proportions of sequences passing the tests are given by Fig. 3. The total number of sequences is 50 which is related to the initial condition \( x_0=[0.5, 1] \). The average of this proportion is 0.94 which means the success of 47 sequences among 50. All calculated proportions are acceptable and we can conclude that there is no deviation of randomness for this map.
the second method to study the uniformity of the sequences. Fig. 5 represents the P-values computed via an application of the Chi-squared function. We notice that the sequences are not uniformly distributed when we applied tests 2 and 14.

For this binary representation, we notice a deviation from randomness in test 14 and a non uniform distribution of sequences in tests 2 and 14. Consequently, we choose the first binary representation to save the randomness and uniformity of the chaotic map.

The P-values plotted in Fig. 6 vary between 0.1822 and 0.9713, so all the sequences pass the first test. We also compute the proportion of sequences passing the tests (Fig. 7), and we observe that the data have failed 8 tests. We observe the P-value distribution of the sequences, and we conclude that the P-values are uniformly distributed only for the frequency and the linear complexity tests. This binary representation must be avoided for this 3-dimensional map because it causes a deviation from randomness.

![Figure 4. Proportion of sequences passing a test.](image)

![Figure 5. P-values calculated using Chi-square function.](image)

![Figure 6. Frequency test P-values vs initial conditions.](image)

![Figure 7. Proportion of sequences passing a test.](image)

### B. Statistical test applied on the 3D map

In this paper we use a piecewise affine map defined on the unit cube $[0; 1] \times [0; 1] \times [0; 1]$. This map is given by the following expressions [9]:

$$
\begin{align*}
    x(n) &= |2x(n-1) + y(n-1) + z(n-1) - 1| \\
    y(n) &= |2y(n-1) + x(n-1) + z(n-1) - 1| \\
    z(n) &= |2z(n-1) + y(n-1) + x(n-1) - 1|
\end{align*}
$$

(2)

Where the initial conditions are $x_0, y_0, z_0 \in [0,1]$. Authors in [9] have demonstrated that it is a chaotic map since it exhibits sensitive dependence on initial conditions and topological transitivity.

In order to test the randomness of this chaotic map, we convert real output on binary format by using the round function and the IEEE 754 single precision format (32 bits).

#### 1) Conversion with round

In this section we test 40 sequences generated by changing initial conditions, where:

$x_0=0.1 \cdot 0.01 \cdot 0.5$, $y_0=[0.3 \cdot 0.01 \cdot 0.7]$, $z_0=[0.45 \cdot 0.01 \cdot 0.85]$. We start by the frequency test.

<table>
<thead>
<tr>
<th>Tests</th>
<th>Proportion of sequences passing a test</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.001017</td>
</tr>
<tr>
<td>1</td>
<td>0.28758</td>
</tr>
<tr>
<td>2</td>
<td>0.44512</td>
</tr>
<tr>
<td>3</td>
<td>0.1822</td>
</tr>
<tr>
<td>4</td>
<td>0.9713</td>
</tr>
<tr>
<td>5</td>
<td>0.1822</td>
</tr>
<tr>
<td>6</td>
<td>0.9713</td>
</tr>
<tr>
<td>7</td>
<td>0.1822</td>
</tr>
<tr>
<td>8</td>
<td>0.9713</td>
</tr>
<tr>
<td>9</td>
<td>0.1822</td>
</tr>
<tr>
<td>10</td>
<td>0.9713</td>
</tr>
</tbody>
</table>

2) IEEE 754 single precision representation (32 bits)

For this binary representation the total number of tested sequences is 120. Each real output is converted to only 16 bits by omitting the 16 first bits including sign and exponents bits. We use 50 triplets of initial conditions, where:

$x_0=0.1 \cdot 0.01 \cdot 0.49$, $y_0=[0.3 \cdot 0.01 \cdot 0.69]$, $z_0=[0.45 \cdot 0.01 \cdot 0.84]$.

Fig. 8 shows the proportion of sequences passing the tests which fluctuates from 0.95 for Serial test to 1 for Universal test. The P-values computed by using the Chi-square application are plotted in Fig. 9. These values vary between 0.001017 for the FFT test and 0.922036 for the Rank test. The proportion of sequences, as well as the theoretical P-values shows that there is no deviation from randomness for this specific binary representation, and the sequences can be considered as good RNG.
The aim of our paper is to construct different chaotic maps like Logistic map [1], [2] and continuous baker map [4]. There are many methods proposed to create S-Boxes based on different chaotic maps like Logistic map [1], [2], Piecewise Linear Chaotic Map [3] and continuous baker map [4], [5]. The proposal of our paper is to construct a dynamic S-Box by combining two chaotic maps. Two methods are used to generate the $16 \times 16$ substitution table. The first one consists of:

- Arbitrarily choose the PWLCM initial condition.
- Iterate this chaotic map $n$ times. Each three output are gathered to form the initial conditions of the three dimensional chaotic map.
- Iterate the 3D map $N$ times, for each set of $(x_0, y_0, z_0)$ and construct a vector $v$ which contains the $N$th iteration of $(x_i, y_i, z_i)$, where $0 \leq i < N$.
- Randomly choose 256 values of the vector $v$.

The basic procedure of the second method is as follows:

- Arbitrarily choose the PWLCM initial condition.
- Iterate this chaotic map $n$ times. Each three output are gathered to form the initial conditions of the three dimensional chaotic map.
- Iterate the 3D map $N$ times, for each set of $(x_0, y_0, z_0)$. Then we construct a vector $v$ which contains the $N$th iteration of $(x_i, y_i, z_i)$, where $0 \leq i < N$.
- Print each element of $v$ in hexadecimal format and convert the result to integers from 0 to 255.
- Finally, organize the resulting table to a $16 \times 16$ matrix.

As we can see, the only difference between the two methods is in how to obtain the 256 integers from the real chaotic output. To construct S-Boxes based on only one map, we just omit the second step and change the first one by arbitrarily choosing the corresponding map initial condition. For the next results we set $n=258$ and $N=550$. 

IV. SECURITY ANALYSIS COMPARISON

The design of cryptographically “good” S-Boxes is based on essential criteria which are:

- Nonlinearity,
- Strict Avalanche Criteria,
- Equiprobable input/output Xor distribution.

A. Nonlinearity

Linear cryptanalysis studies linear approximations of the cryptosystem. The purpose is to construct linear equations between input plaintext and output cipher text and enumerate all linear approximations of the S-Box in a linear approximation table. Otherwise, the nonlinearity is measured by calculating the linear probability approximation ($L_p$) [1]:

$$L_p = \frac{\max_{a,a'} \left( \# \{ x \in X : x \cdot a = (f(x) \cdot b) \} \right)^2 }{2^{n-1}}$$

Where $a, b \in \{1,2,...,2n-1\}$, $x.a$ is the parity of the binary product of $x$ and $a$.

Decreasing $L_p$ leads to increasing linear attack complexity. In fact, we calculate $L_p$ for six different S-Boxes constructed by using: the 3-dimensional map, the PWLCM and the two maps combination. The Table I collect these results for the two proposed methods. We remark that the S-Box based on the two maps combination has the lowest $L_p$ by using the first method, so it is immune against linear attacks. This value is equal to Jakimoski’s result [1] and better than results in [4], [5] and [6] where the computed values are respectively 0.0706, 0.0706 and 0.088135.
TABLE I. LINEAR PROBABILITY APPROXIMATION

<table>
<thead>
<tr>
<th>Chaotic map</th>
<th>Lp</th>
<th>Lp’</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D</td>
<td>0.0625</td>
<td>0.0881</td>
</tr>
<tr>
<td>x0=0.8; y0=0.5; z0=0.1;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWLCM</td>
<td>0.0791</td>
<td>0.0705</td>
</tr>
<tr>
<td>p=0.15; x0=0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 maps combination</td>
<td>0.0625</td>
<td>0.0705</td>
</tr>
<tr>
<td>p=0.15; x0=0.7;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 and 2. Are the first and second methods.

B. Strict Avalanche Criterion (SAC)

A function satisfies the strict avalanche criterion (SAC) if for one input bit complemented, each output bit changes with a probability of one half [10].

We can define this criterion as follow:

The function \( f(x) \) is a Boolean function, \( \delta \) is the changed bits sum such as:

\[
\delta = \sum_{x \in \mathbb{Z}_2^n} f(x) \oplus f(x \oplus c_i)
\]

Where \( x \) and \( c_i \) are two \( n \) bits vectors which differ only in one bit i.

The Boolean function \( f(x) \) accomplishes SAC criterion if and only if \( \delta = 2^{n-1} \) for all i, \( 0 \leq i \leq n - 1 \).

According to [10], we should calculate a dependence matrix and extract its mean value. If we obtain a value close to 0.5, then the cryptographic transformation satisfies the SAC criterion.

By using the second method, the S-Boxes based on the PWLCM and the two maps combination have the closest values to 0.5 (Table II). Therefore, the two S-Boxes satisfy SAC criterion.

In addition, these results are better than the ones obtained in [1], [3] and [4] which are respectively: 0.4972, 0.5125 and 0.4995.

TABLE II. STRICT AVALANCHE CRITERION

<table>
<thead>
<tr>
<th>Chaotic map</th>
<th>SAC1</th>
<th>SAC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D</td>
<td>0.5044</td>
<td>0.5051</td>
</tr>
<tr>
<td>x0=0.8; y0=0.5; z0=0.1;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWLCM</td>
<td>0.5049</td>
<td>0.5</td>
</tr>
<tr>
<td>p=0.15; x0=0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 maps combination</td>
<td>0.4993</td>
<td>0.4998</td>
</tr>
<tr>
<td>p=0.15; x0=0.7;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 and 2. Are the first and second methods.

C. Equiprobable input/output XOR distribution

Differential cryptanalysis uses the propagation predictability of the chosen plaintext differences in order to attribute probabilities and find the most probable key. This attack was introduced the first time by Biham and Shamir in [11].

To measure the complexity of the differential attack, we calculate for each S-Box the differential approximation probability of a map \( f \) (DP) which is given by the following expression:

\[
DP_f = \max_{\Delta x \neq 0, \Delta y \neq 0} \left( \frac{\#\{x \in X / f(x) \oplus f(x \oplus \Delta x) = \Delta y \} }{2^n} \right)
\]

Where \( x \) is the set of all possible input value, \( 2^n \) is its cardinality, \( \Delta x \) is the input differential and \( \Delta y \) is the output differential.

We calculate the occurrence frequency of the most probable output XOR for the S-Boxes generated respectively by the first and the second method. We notice that the most probable outputs of the first method are 6 and 8 which occurred 154 and 86 times respectively and 161 and 80 times respectively for the second one. Nevertheless, the DP of the S-Box based on the two maps combination is equal to 10/256 for both methods (table III).

This value is lower than the Jakimoski’s [1] and Chen’s [5] result which is 12/256 and equal to Tang’s [4] and Asim’s [3] results.

We can say that our S-Box has equiprobable input/output distribution.

TABLE III. DIFFERENTIAL APPROXIMATION PROBABILITY

<table>
<thead>
<tr>
<th>Chaotic map</th>
<th>DF1</th>
<th>DF2</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D</td>
<td>12/256</td>
<td>12/256</td>
</tr>
<tr>
<td>x0=0.8; y0=0.5; z0=0.1;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWLCM</td>
<td>10/256</td>
<td>12/256</td>
</tr>
<tr>
<td>p=0.15; x0=0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 maps combination</td>
<td>10/256</td>
<td>10/256</td>
</tr>
<tr>
<td>p=0.15; x0=0.7;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 and 2. Are the first and second methods.

V. CONCLUSION

In this paper we have studied the statistical properties of two chaotic maps. For this purpose the NIST test suite was applied and we have computed the P-values as well as proportion of sequences passing the tests.

We have verified that the two chosen maps have random like behavior in addition to their chaotic characteristic. Nevertheless, the choice of the conversion methods can disturb the randomness and the uniformity of sequences.

In fact, the IEEE 754 single precision format (32 bits) should be avoided and the round method is recommended when we need to use the PWLCM in cryptosystem. This is not the case for the 3-dimensional map where the IEEE 754 single precision format (32 bits) is more efficient. In conclusion, these chaotic maps seem to be suitable to design good S-Box.

In the two last sections, we have presented new methods to design chaotic dynamic S-Box and we studied their robustness against linear and differential attacks. So, we made a comparison of different chaotic S-Boxes for some security criteria such as: nonlinearity, equiprobable input/output distribution as well as the strict avalanche criterion (SAC).

This comparison shows that the S-Box based on the two map combination has: the lowest Lp by using the first method and the closest value to 0.5 of the strict avalanche criteria. This chaotic dynamic S-Box has also an equiprobable input/output distribution.
It will be very interesting to use this S-Box in radio environment based on WPAN like Zigbee technology, and more precisely in the AES CCM mode.

REFERENCES


