"POLITEHNICA" UNIVERSITY OF BUCHAREST

ETTI-B DOCTORAL SCHOOL

Cryptographic Methods and Techniques using Chaos Theory and Statistics

Dezvoltarea de metode şi tehnici criptografice folosind teoria haosului şi statistica

by **Ing. Corina MACOVEI**

RESUME

THESIS COMMITTEE

President	**Prof. Dr. Ing. Gheorghe BREZEANU**	from	**Universitatea "Politehnica" Bucureşti**
Coordinator	**Prof. Dr. Ing. Adriana VLAD**	from	**Universitatea "Politehnica" Bucureşti**
Reviewer	**Prof. Dr. Ing. Alexandru ŞERBĂNESCU**	from	**Academia Tehnică Militară Bucureşti**
Reviewer	**Prof. Dr. Ing. Mihai CIUC**	from	**Universitatea "Politehnica" Bucureşti**
Reviewer	**Prof. Dr. Ing. Victor-Adrian GRIGORAŞ**	from	**Universitatea Tehnică "Gheorghe Asachi" Iaşi**

Bucharest 2020

Contents

Chapter 1

Introduction

The thesis *Cryptographic methods and techniques using chaos theory and statistics* investigates a current topic - private communication - with the help of chaos theory and statistics.

Deterministic chaos is a phenomenon that at first glance seems random, but this behavior can be exactly determined if we know with infinite precision the initial state of the system that generates it. Because the accuracy of the initial state of the generating system cannot be infinite, we can say that a correct prediction of chaotic systems behavior can be made within certain limits. On the other hand, all chaotic systems are governed by laws of motion, so we can conclude that there is order in the apparent chaos. The term chaos comes from our difficulties to detect regularities in this type of dynamic. Despite its determinism, this system produces choices that surprise our minds. Hence, the name deterministic chaos.

A practical example of deterministic chaos is the butterfly effect, discovered by the meteorologist Edward Norton Lorenz, [1]. This phenomenon, in which the beating of the wings of a butterfly here, may cause a hurricane in another part of the world, demonstrates the dependence of the phenomenon on the initial conditions. The meteorologist also studied the weather forecast. During an experiment he interrupted the computer simulation, and resumed it after a pause, with 3 decimals, instead of 6 decimals previously used by the machine he was working on. To ensure the continuity of the results obtained, Lorenz resumed the calculation of the last few dozen points. He

noticed with surprise that these points no longer coincided with those calculated in the previous simulation. Here is how a truncated knowledge of the initial state of the generating system can lead to the divergence of the initially neighboring trajectories.

One of the most popular and simplest chaotic system is the logistic function, based on which the evolution of the population can be modeled, as shown by the biologist Robert May. The logistic function will serve to exemplify some methods and techniques developed by this work. Another interesting and very current example is the spread of the COVID-19 virus worldwide, which can be compared to the butterfly effect, the speed of its spread finding the world unprepared. A particularly useful future study for everyone would be to investigate the chaotic system on which the spread of global diseases and pandemics could be mapped. The work of Steven Strogatz and his collaborators is a good starting point for this future research.

The need to encrypt messages has amplified with the trend of digitization of all activities in society. Cryptography is an ancient science, the first records dating from the era of Emperor Julius Caesar. However, the research and development of cryptographic algorithms began relatively late in 1970. Then this field developed rapidly. The basic principle of encryption is to modify the original message so that it can only be decrypted by the recipient. This should lead to a very short decryption time for the recipient and a very long time for any other entity that would attempt to decrypt the message. The first forms of encryption were permutations and substitutions. The performance of computers and their low costs have facilitated the development of new algorithms.

Deterministic chaos is used as a generator of disorder in cryptographic algorithms. Basically, we hide the message in a chaotic stream of data. At the reception, in order to extract the message, it is necessary to know the value of the parameters and the initial conditions of the chaotic system. Thus, deterministic chaos provides a favorable context for its use in cryptographic applications.

Chapter 2

Autocorrelation function's relevance for the statistical independence sampling distance

A property of high-performance cryptosystems is that after analyzing the data produced by the cryptosystem, a relationship between the analyzed samples is not revealed. They need to be decorelated, the lighter version of the statistical independence, useful in practice, where there is often not enough information. In this regard, Chapter 2 presents a mean test on the autocorrelation function, customized on two of the simplest chaotic systems used in cryptography: the tent function and the logistic function. These results were disseminated in two papers indexed in the ISI Web of Science, in 2019 [2], respectively 2020 [3].

The first system investigated in this paper is the tent function, described by (2.1), where p is the control parameter and x_0 is the initial condition. The discrete time moment is denoted, here, k.

$$x_{k+1} = \begin{cases} \frac{x_k}{p}, 0 \leq x_k \leq p \\ \frac{1-x_k}{1-p}, p \leq x_k \leq 1 \end{cases} \tag{2.1}$$

Fig. 2.1 shows two trajectories of the tent function, for two initial conditions randomly chosen in $(0, 1)$. Keeping the value of the parameter p constant and choosing $N = 10000$ initial conditions x_{0j} from a uniform random distribution in $(0, 1)$, was built a random process, j varies from 1 to N. In Fig 2.1 are represented two samples.

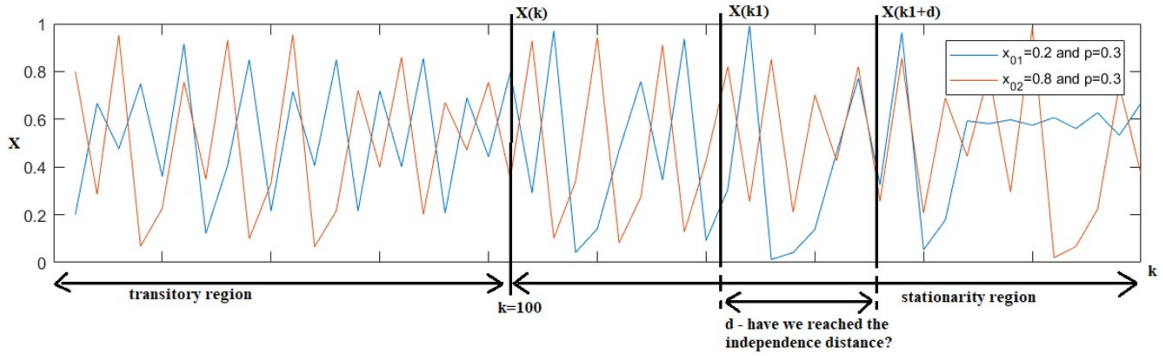


Figure 2.1: The temporal evolution of the system (2.1) for the same parameter $p = 0.3$ and two initial conditions, $x_{01} = 0.2$ and $x_{02} = 0.8$.

Returning to the random process consisting of the N curves for N different initial conditions, in order to investigate its statistical properties we need to be in the stationary region, and the random process to be ergodic. If the control parameter is chosen correctly, and the system is in the stationary region, the random process generated by them is ergodic.

Since we choose the initial conditions from a uniform distribution in (0,1), sampling the N curves at iteration 1, we will have a probability density function corresponding to this type of distribution. During the transient time of the random process, the probability density function will vary from a uniform to that of the investigated process. At successive moments $k_1, k_1 + 1, \ldots$ this probability density will remain constant, considering that at k_1 it enters the stationary region. The tent function is a particular case of chaotic signal, for which the probability density function is uniform, but this will not be the case of the second chaotic function used as an example in this chapter, the logistic function.

Ergodicity is another statistic property relevant to our study. This assures us that the information obtained from the analysis of a long temporal trajectory of tens, hundreds of thousands of iterations, will be present in the study of any variable obtained by sampling the random process, once in the stationary region.

Once stationary, we can raise the issue of statistical independence. Therefore, we look for that distance between the time moments k_1 and k_{1+d} which allows us to obtain two statistically independent random variables. Existing correlation tests in

the literature before 2006, such as those using the Spearman [4] correlation coefficient or the Pearson [5] degree of correlation, solved the problem only for Gaussian random variables. In that case, the decorrelation of the data implies statistical independence. Nevertheless, the random processes generated by chaotic systems do not follow a Gaussian distribution. Thus, the decorrelation attested by the tests [4, 5] would not lead to the implicit decision of statistical independence between the two random variables investigated.

The paper [6] solved this situation by applying random variable transformations on the empirical distributions resulting from the random variables obtained by sampling the random process studied at the compared iterations, k_1 and k_{1+d}. The test from [6] (Badea-Vlad) is a statistical independence test applicable to any law of probability, not just Gaussian law, and has been used to evaluate discrete-time systems, such as the tent map or the three-dimensional system Rössler. The paper [7] evaluates the performance of this test and completes it for situations of interest, in 2016, automating the decision of the test proposed by [6], which was initially visually evaluated to decide whether, after transformations, the two random variables investigated are jointly-Gaussian, allowing the decision of decorrelation to be equivalent to that of statistical independence. The article [8] investigates statistical independence in the context of the tent map (2.1). In [9, 10] the discussion moves to the logistic map. The research in [10] answers the question: "Does a small variation of the logistic map parameter affect the study of the random process generated?", an essential answer for our study.

The answer is, as was intuited, that the random process is determined by the value of the chaotic system parameter, for the logistic map, denoted by R in the relation (2.2).

$$x_{k+1} = R \cdot x_k(1 - x_k) \tag{2.2}$$

Fig. 2.2 gives an idea of how different the trajectories of the logistic map (2.2) are depending on its parameter. The question we ask ourselves is: how much does a slight deviation of the control parameter affect our study?

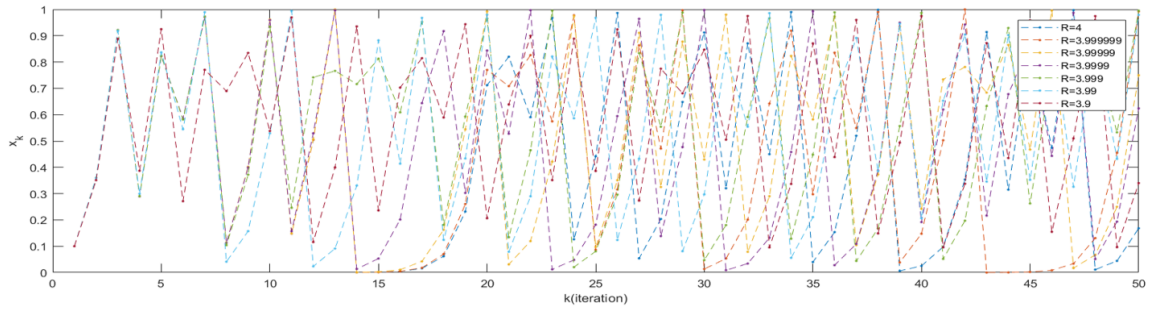


Figure 2.2: Multiple trajectories of the logistic map for different $R = 4(1 - \epsilon)$ and the same initial condition $x_0 = 0.1$.

The autocorrelation function shows how correlated is a signal with a copy of itself at a later time. In our study, the statistical autocorrelation is investigated. The random variables whose autocorrelation is studied are obtained by sampling the 10k of the previously mentioned trajectories, representing the random process given by the chosen control parameter.

The aim of the study is to determine the sampling distance from which the data become uncorrelated. In this sense we introduce the autocorrelation function defined for the random process X corresponding to the chaotic function, in the relation (2.3) and the experimental autocorrelation function (2.4), where $x_i(k_1)$ and $x_i(k_1 + d)$ are the values of the iterations k_1, respectively $k_1 + d$ of the trajectory i, with i varying from 1 to N, N being the number of trajectories used to simulate the random process. It is important to mention, k_1 must be chosen after the time interval corresponding to the transient time, therefore, in the stationary region.

$$R_X(d) = E[X(k_1)X(k_2)] = E[X(k_1)X(k_1 + d)] \tag{2.3}$$

$$R_{exp,X}(d) = \frac{1}{N}\sum_{i=1}^{N} x_i(k_1)x_i(k_1 + d) \tag{2.4}$$

In this chapter we also performed an analogy between two tent map variants with different domains of definition, equal by a transformation of a random variable, $Y = 2X - 1$. This equivalence leads to a simple calculation of the autocorrelation function, the results obtained on one system can be translated for the other system.

On the left Fig. 2.3 are three autocorrelation functions corresponding to the tent map, calculated for several control parameters. In this case, it is observed that the autocorrelation function tends to the value 0.25. In the case of the logistic map, on the right Fig. 2.3 are graphically represented the autocorrelation functions for various values of the parameter R, $R = 4$ and two values very close to 4 ($R = 3.999999$ and R =3.9999) (top right), as well as two more distant values ($R = 3.99$ and $R = 3.78$) (bottom right). At first glance we notice that for $R = 4$ and d large enough the autocorrelation function tends to the value 0.25. Similarly, for values very close to 4, it is difficult to visually notice a difference. For R more different from the reference value $R = 4$, the autocorrelation function no longer has a mean value of 0.25, but is completely different, about 0.28 for $R = 3.99$ and 0.41 for $R = 3.78$.

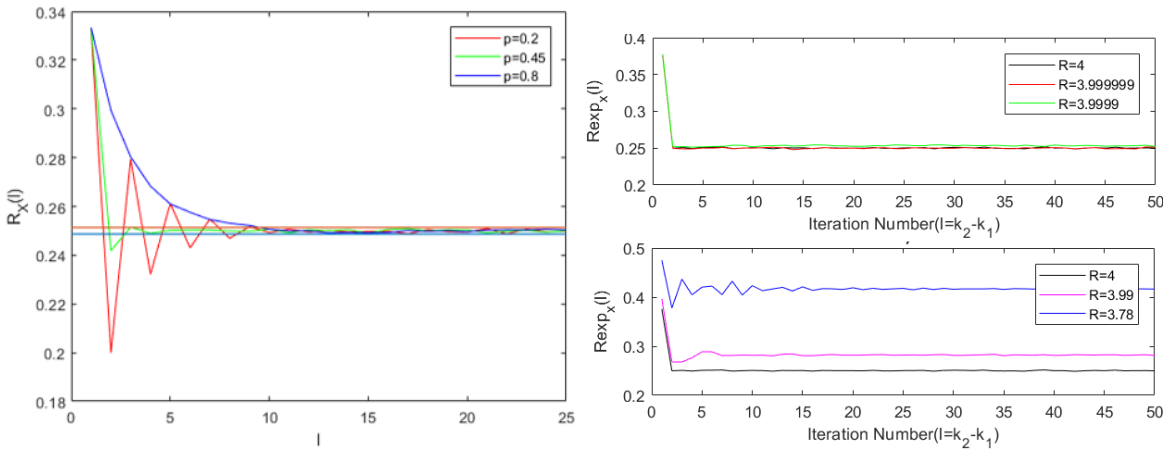


Figure 2.3: The experimental autocorrelation function associated with tent map (left) and logistic map (right).

The value of the autocorrelation function for the parameter $R = 4$ was easily determined, for this particular situation the statistical distribution of the logistic map being known, the mean and the dispersion are at our disposal.

Therefore, the relation (2.5) shows the mean of the random variable ξ, in the specified range, with ε being the confidence interval, corresponding to the $\alpha/2$ quartile characteristic to the Gaussian law. $N = 10000$ represents the number of curves considered for the study, and σ corresponds to the dispersion of the r. v. values ξ.

$$R_{exp,X}(d) \in [\mu_\xi - \varepsilon; \mu_\xi + \varepsilon]; \varepsilon = Z_{\alpha/2}\frac{\sigma_\xi}{\sqrt{N}} \tag{2.5}$$

Thus, we can apply the statistical test that will tell us if we are in the situation of decorrelation of the investigated data, the two random variables at times k_1 and $k_1 + d$.

The test has two hypotheses:

- H_0 - the expected value of the autocorrelation function has been reached (the distance d could correspond to the statistical independence between $X(k_1)$ and $X(k_2)$);

- H_1 - we are below the searched value of the autocorrelation that could ensure statistical independence ($X(k_1)$ and $X(k_2)$ are not statistically independent).

In order to highlight the results obtained, we applied a Monte Carlo analysis on a group of autocorrelation functions, an analysis that also highlighted practical computational aspects, such as precision in calculation and degradation of trajectories. Several autocorrelation functions were calculated on groups of $N = 10^4$ or $N = 10^5$ trajectories of the tent map and logistic map. 500 such autocorrelation functions were calculated and the acceptance percentage of the mean test was analyzed. Our question was: how many autocorrelation functions are in the test acceptance range for a given d?

For the tent map the results showed that for $l > 10 - 15$ iterations in almost 95% of cases the test is passed for any control parameter, and in the case of the logistic map it was observed that for the control parameter $R = 4$ (the only parameter for which we know the mean and the theoretical dispersion and we can calculate the mean test), in almost 95% of cases the test is passed for $l > 5$, as in Fig. 2.4.

The sensitivity to the control parameter and how it is reflected in the mean test for values of the R parameter in a neighborhood of 4 for the logistic map was further studied. We considered the same acceptance interval as for $R = 4$. In each image in Fig. 2.5 500 autocorrelation functions are plotted for various values of the control parameter. In the top left figure, for R very close to 4, a percentage of 95% of the values are in the established range. In the bottom left figure we can see that only 68% of the values are in the desired range corresponding to $R = 4$. If we move away

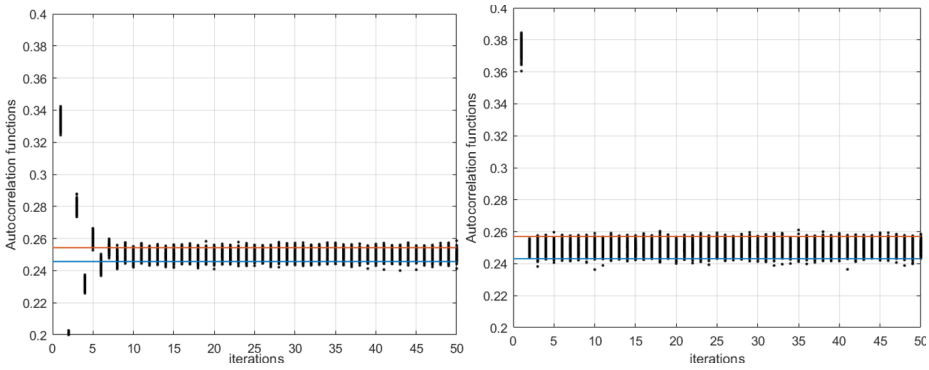


Figure 2.4: Monte Carlo analysis of the mean test on the autocorrelation function $N = 10000$; $L = 500$, up to $l = k_2 – k_1 = 50$. Left: tent map, $p = 0.2$. Right: logistic map, $R = 4$.

from $R = 4$, the autocorrelation function has a new mean value, as we can see in the figures on the right. Thus, the results from previous studies are reconfirmed, because for R very close to 4 the probability law seems to be the same as for $R = 4$, and when we move away we have a completely different mean and dispersion than those for $R = 4$.

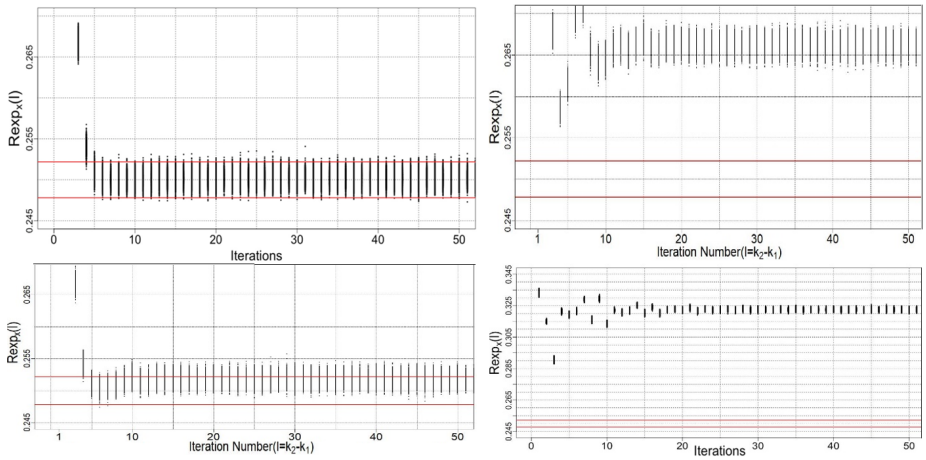


Figure 2.5: $N = 10^5$ different trajectories for the logistic map, 500 autocorrelation functions. Left: $R = 3.999999$ (up), $R = 3.9999$ (down) and right: $R = 3.99$ (up), $R = 3.78$ (down).

In our study we observed that the mean test is passed for d values lower than those obtained with the statistical independence test[6, 7]; this may show that the variables $X(k_1)$ and $X(k_2)$ are uncorrelated. For the tent map, in Table 2.1 the last column represents the decorrelation distance observed in the mean study on the

p	Statistical independence distance [6]	Decorrelation distance observed with the proposed test
0.25	25	8
0.45	15	4
0.55	15	3
0.75	35	7

Table 2.1: The relationship between $R_X(l)$ and the statistical independence distance for the tent map.

autocorrelation function. The second column represents the statistical independence distance evaluated in the previous articles [6, 7]. These values depend on the control parameter of the chaotic system. The study continued with a reassessment of the minimum statistical independence distance for the tent map and the logistic map, the results from the literature being reconfirmed.

The test is an important numerical tool that measures the relationship between the statistical independence assessments and the autocorrelation. Passing the test does not necessarily mean that the variables $X(k_1)$ and $X(k_2)$ are independent, but uncorrelated. Statistical independence must be assessed by specific methods. So the mean test does not replace the statistical independence test, but it is a support of the statistical independence obtained. On the other hand, if the test is not passed, we clearly do not have statistical independence for the distance d at which the autocorrelation function was calculated. For the tent map we can apply the mean test for any control parameter, because the probability law is uniform for all values of the control parameter. For the logistic map, however, the study is more complex: we know the law only for the control parameter $R = 4$; for this value we can calculate the mean and dispersion of the random variable x_i and we can apply the mean test in the case of the known variance. For values $R \neq 4$, the theoretical mean and the dispersion must be calculated by means of specific procedures that are the object of future study.

Chapter 3

Generating a continuous chaotic space for a chaos-based pRNG

Chapter 3 resumes a pseudo-random number generator based on Hénon's generalized three-dimensional system, a discrete system in time. The results are disseminated in a ISI conference [11] and a Q2 journal [12]. A pseudo-random number generator is a technique for generating sequences whose elements are independent. The random number generator was proposed in the paper [13] and implemented in a Field Programmable Gate Array (FPGA) in the article [14] by the same authors. An EXCLUSIVE OR between the least significant 8 bits of the 64-bit binary representations of the three states of the system leads to a series of apparently random bytes which, for the parameters a and b properly chosen, pass the NIST tests (National Institute of Standards and Technology). These bytes can be used, among other areas of applicability of pRNGs, for text and image encryption, which are also represented on 8 bits, according to the ASCII correspondence.

The mentioned system has a chaotic or hyperhaotic behavior, as the research from [15] shows us, only for certain values of the bifurcation parameters a and b.

The bifurcation diagrams in Fig. 3.1 shows the number of solutions for the system (3.1) with the parameters $a \in (0, 2)$, $b \in [-0.3, 0.3], x, y, z \in (-2, 2)$, when the parameter a is fixed and b is in the range $(-1, 1)$. We observe that in the case of $a = 0.15$, in Fig. 3.1 (left image), for b in $(-0.7, 0.87)$ Hénon's three-dimensional system converges to a single value, being far from the chaotic behavior characterized

by a very large number of solutions, as shown by the b region in $(0.87, 1)$.

$$\begin{aligned} x_{k+1} &= a - y_k^2 - bz_k \\ y_{k+1} &= x_k \\ z_{k+1} &= y_k \end{aligned} \tag{3.1}$$

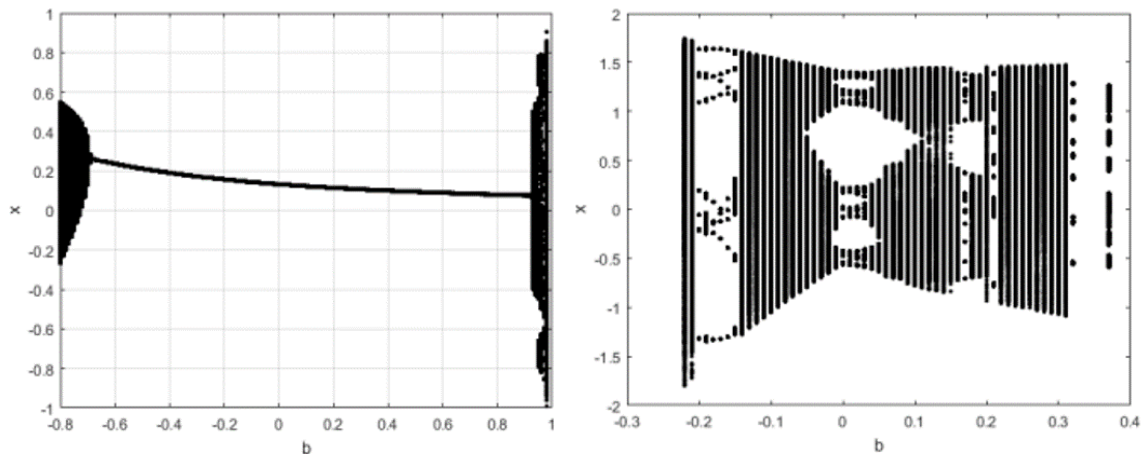


Figure 3.1: Bifurcation diagrams for the parameter $a = 0.15$ (left), $b = 1.4$ (right).

For $a = 1.4$, the situation is even more complex, when b successively changes its value, with a step of 10^{-2}, the system (3.1) alternates between periodic behavior, with 8 solutions, such as for $b = -0.15$, 32 solutions as is the case for $b = 0.02$ or covering the entire amplitude range of x as for $b = -0.19$. This discontinuous pattern for pseudo-randomism must be eliminated to obtain a pRNG with statistically satisfactory properties. A detailed study in this regard is conducted in [16].

The Chapter 3 contribution is, in this context, an algorithm for dynamically changing the values of the bifurcation parameters a and b to generate pseudo-random sequences that do not depend on their initial value at the execution of the pRNG.

In a three-dimensional space, the phase space or attractor, we see in Fig. 3.2, for different pairs of parameters (a, b), that the Hénon system operates in chaotic mode (left), periodic (center), or even divergent to infinity (right).

This chapter comes with an improvement on the pseudo-random number generator from the literature [13], which used a pair of constant parameters in the evolution of

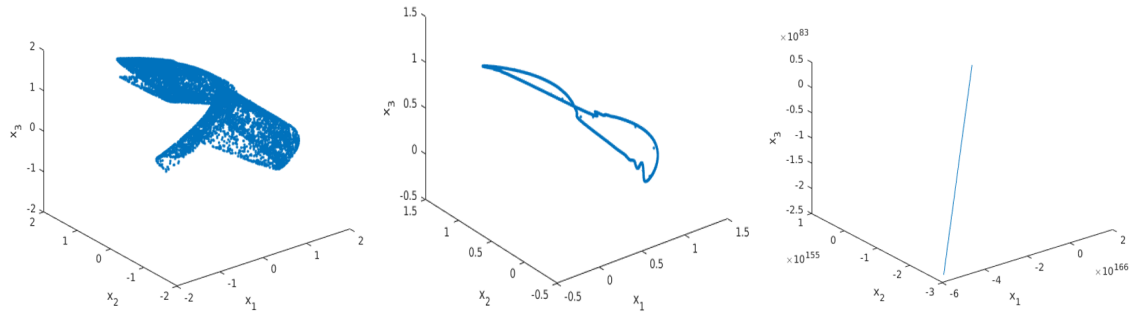


Figure 3.2: Different behavior of the 3-dimensional Hénon system for different (a, b) pairs.

the Hénon's generalized system and produced, at each iteration, the values (x, y, z). Their least significant bytes (LSB) were summed modulo 2 without transport (XOR-ized), so that a pseudo-random byte was generated by the system at each iteration. Its block diagram is shown in Fig. 3.3. The improved variant, shown in Fig. 3.4,

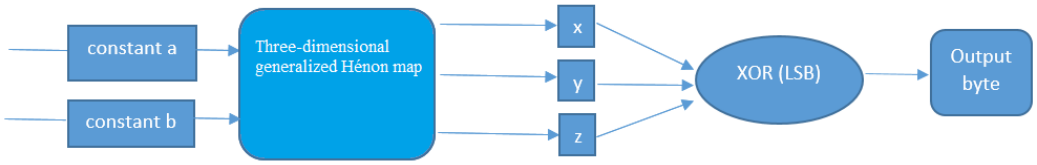


Figure 3.3: The diagram of the previous pRNG.

makes the result of the encryption no longer dependent on the choice of the (a, b) parameters. The update function of these parameters, denoted by e_1, will sum the values x, y and z, at each iteration, and will update the values a and b, scaling them in the corresponding intervals, $(-2, 2)$ for a, respectively $(-0.3, 0.3)$ for b, (3.2). Thus,

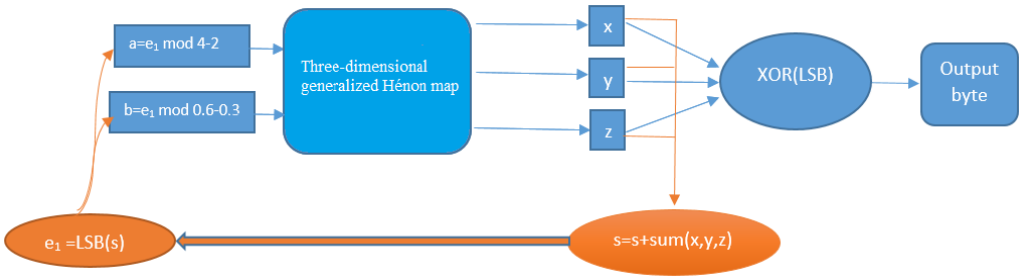


Figure 3.4: The diagram of the proposed pRNG.

in each iteration, we are dealing with a chaotic, periodic or divergent behavior of the Hénon's three-dimensional system, without this alternation to affect the quality of the image encryption. Starting from (a, b) previously given, we go through different regimes in subsequent iterations, as in Fig. 3.2.

$$\begin{aligned} a &= e_1 mod 4 - 2 \\ b &= e_1 mod 0.6 - 0.3 \end{aligned} \tag{3.2}$$

For the top left image in Fig. 3.5, we start with the pair (a, b) using the new pRNG. By changing the parameters of the Hénon's generalized three-dimensional system with the formulas presented, (3.2), the choice of (a, b) parameters corresponding to a chaotic behavior becomes an optional requirement for the proper functioning of the pRNG based on this system. The encrypted image does not reveal the original one clearly, neither visually, nor statistically, as can be seen in the upper right encrypted image in Fig. 3.5 and the corresponding histogram.

In the work [12] is disseminated one step further in research, by adding a second function, in addition to e_1, which can update the parameters (a, b) as the plain image is encrypted, bit by bit. This is e_2 from 3.3. In the extended study an additional aspect is considered: the updating of the parameters (a, b) is no longer done at each iteration, but after an update interval.

$$\begin{aligned} e_1 &= e_1 + x_k + y_k + z_k \\ e_2 &= e_2 + sin((z_k - y_k)/2) \end{aligned} \tag{3.3}$$

This extension tests several systems whose general form is presented in 3.4. We want a configuration in which the place of the Hénon's three-dimensional system can be replaced by any other system capable of manifesting chaotic behavior.

$$\begin{aligned} x^+ &= y \\ y^+ &= z \\ z^+ &= f(x, y, z) \end{aligned} \tag{3.4}$$

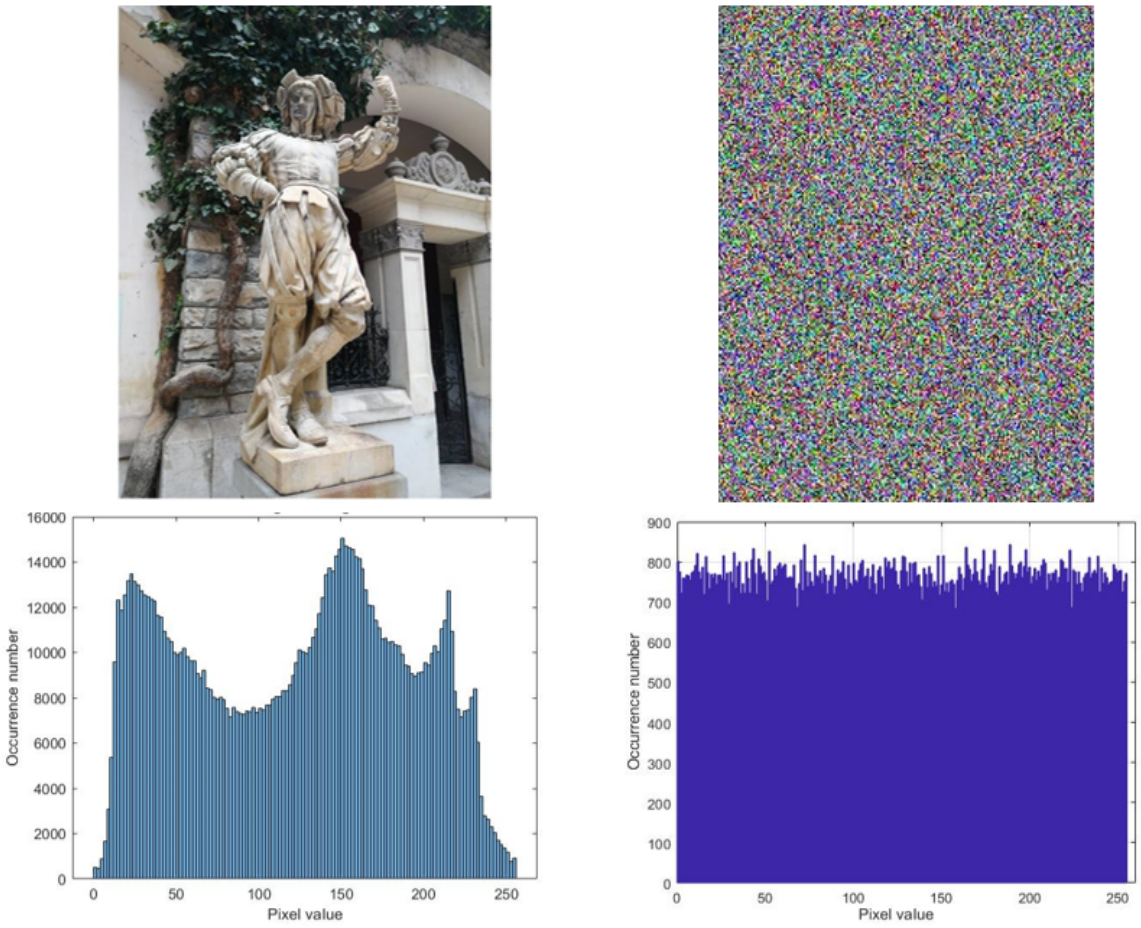


Figure 3.5: The result of encryption with the new pRNG: the original image (top left) and the corresponding histogram (bottom left), the encrypted image (top right) and its histogram (bottom right).

The resulting PRNG is tested with the test batteries of the American National Institute of Standards and Technologies (NIST), but also with the U01 Test [17]. With this test platform, several variants of the pRNG proposed by the present paper were tested, as well as other pseudo-random number generators in the literature, [18, 19], to serve for the comparison of its performances.

Chapter 4

Cryptanalysis of the key-matrix in a symmetric encryption

The idea of Chapter 4 started from some works in the literature [20, 21, 22]. While attending a workshop *Chaos Analysis and Applications* in Cergy-Pontoise, France, we discussed some ideas about the importance of statistical analysis in cryptanalysis. These discussions led to the dissemination of the results in an ISI journal article [24] and an ISI conference ATOMn 2020 [23].

In this chapter we perform a statistical analysis of an encryption scheme of multimedia data in general, in particular, here, of an image. The original X image is multiplied by an array ϕ containing pseudo-random numbers, the secret key. The result is stored in the Y matrix. This is the investigated private communication scheme, (4.1). The histograms of an image encrypted with this method are shown in Fig. 4.1.

$$Y \;=\; \phi \cdot X \tag{4.1}$$

We aim to investigate the scenario in which the cryptanalyst finds the ϕ matrix, wanting to extract the secret key. This is easy if the attacker has the encryption machine at his disposal. He encrypts a known image, X'. The output of the algorithm gives him Y'. Knowing X', he multiply 4.1, to the right, by the inverse of the matrix X' and he finally obtains the key matrix ϕ.

It is known that the ϕ matrix has as elements the values of the logistic map generated as in Fig. 4.2, in the scenario investigated here. Therefore, the attacker tries, with the secret matrix at his disposal, to find out additional information about

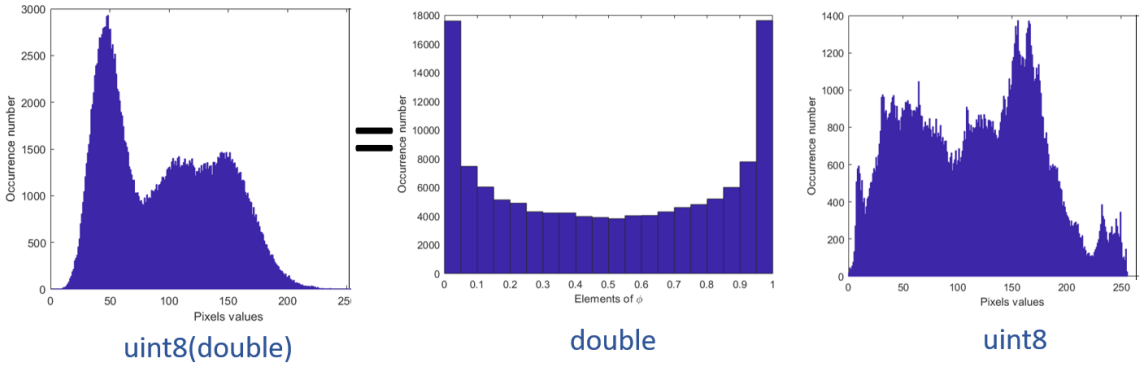
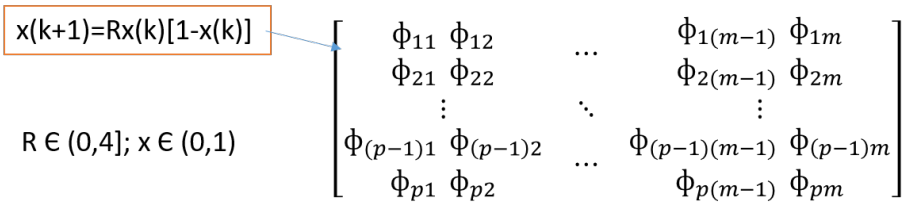


Figure 4.1: Histograms corresponding to the original image (right), the ϕ matrix (center) and the encryption result (left).

the control parameter R and the initial condition that generated it. Is this possible? We are considering both the case where the ϕ matrix is generated with the logistic



$$\begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1(m-1)} & \phi_{1m} \\ \phi_{21} & \phi_{22} & & \phi_{2(m-1)} & \phi_{2m} \\ \vdots & & \ddots & \vdots & \\ \phi_{(p-1)1} & \phi_{(p-1)2} & \cdots & \phi_{(p-1)(m-1)} & \phi_{(p-1)m} \\ \phi_{p1} & \phi_{p2} & & \phi_{p(m-1)} & \phi_{pm} \end{bmatrix}$$

Figure 4.2: ϕ matrix generation.

map considering all the successive iterations, and the case when we take the iterations with a certain sampling step. This strategy is inspired by the team's articles on the subject of the statistical independence distance illustrated in *Chapter 2*, [6, 7].

The 2 scenarios are:

- The ϕ matrix, generated with the pseudo-random number generator - represented by the logistic map - is constructed from the successive iterations of the logistic map;
- The sampling distance between the considered iterations of the ϕ matrix is the statistical independence distance.

For both cases we consider the transient time of 250 iterations, according to the results in the literature. So let's assume we're the cryptanalyst and we found the ϕ

matrix. We intend to determine the control parameter R. For this purpose we use several statistical tools: the Smirnov test (to test whether two populations come from the same distribution), histograms (visual interpretation of data), the autocorrelation function (to see the influence of the control parameter on it), and the investigation of the successive iterations.

The random process associated to the logistic map is an ergodic process determined by the R parameter, as we discussed in *Chap. 2*. So the initial condition, x_0, is irrelevant. The R parameter used to generate the ϕ matrix can be retrieved by applying the Smirnov test. The test is passed if we apply it to samples extracted from the matrix ϕ and samples extracted from the trajectories of the logistic map having the same parameter R, even if we start from different initial conditions.

Histograms clearly show that for a different control parameter, the frequency distribution is different (for successive iterations or considering the independence distance), Fig. 4.3.

The autocorrelation function investigated in *Chap. 2* is also studied in this

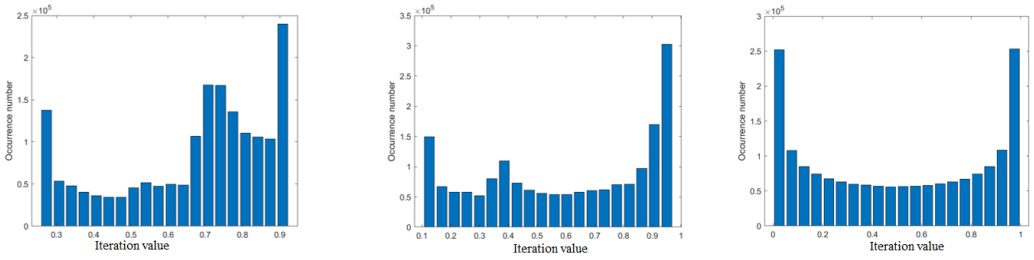


Figure 4.3: Histograms of the logistic map for $x(0) = 0.4557$, $d = 1$ and $R = 3.7$ (left), $R = 3.89$ (center), $R = 4$ (right).

context. We conclude that its shape depends on the R parameter. Analyzing the autocorrelation function for the same R and starting from different initial conditions, the autocorrelation function has the same form. If R is different, even for the same initial conditions the form of the autocorrelation function differs. We have the same results when we consider successive iterations and when we consider iterations with a given distance. The random process is determined only by the R parameter, as we intuited from the beginning and as one can see in Fig. 4.4.

Investigating the iterations of the ϕ matrix, to recover the R parameter, consid-

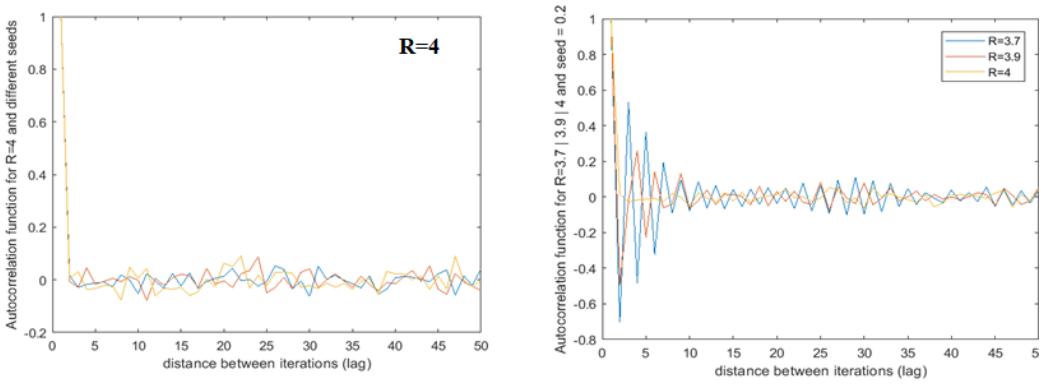


Figure 4.4: The autocorrelation function for $R = 4$ (left) and variable R (right).

ering successive iterations, but also considering the values sampled according to the independence distance, the results show that : if we know successive iterations we can find the parameter R with a relative error of order 10^{-9} as in Table 4.1, while if we have access to the iterations of the logistic map with a specific distance between samples, the recovered R is very different from the original one as can be seen in Table 4.2. These results show that successive iterations extracted from the logistic map are needed in order to apply this method.

R	3.999029407218239	4
$\hat{\mathbf{R}}$	3.999029407100511	4.000000000137501
[($\hat{\mathbf{R}}$ - R)/R] [%]	**2.94391433550104e-9**	**3.437525e-9**

Table 4.1: Erori relative între R estimat și R original, $d = 1$.

R	3.999029407218239	4
$\hat{\mathbf{R}}$	3.052585858468511	0.227703495214937
[($\hat{\mathbf{R}}$ - R)/R] [%]	**23.66683143268213**	**94.30741261962658**

Table 4.2: Erori relative între R estimat și R original, $d > 1$.

Chapter 5

Cryptographic algorithms based on wavelet packets and chaotic systems

This chapter presents improvements of some cryptographic algorithms in the literature by using wavelet packets and chaotic systems. Wavelet packets are used in the pre-processing phase of the algorithm, while chaotic functions are used to spread the pixels' values across the all possible range. The results are disseminated in a journal article [24] and two ISI conference papers, [23] and [25].

In Fig. 5.1 is presented the 2D discrete wavelet transform, implemented using digital filters and descending sampling. On a decomposition level we have an approximation coefficient (A_{j+1}) and 3 detail coefficients (H_{j+1}, V_{j+1}, D_{j+1}). The Haar transform was used in the simulations because it maps to the needs of the algorithm, allowing the reconstruction of the signal, in our case the image, without introducing edge effects.

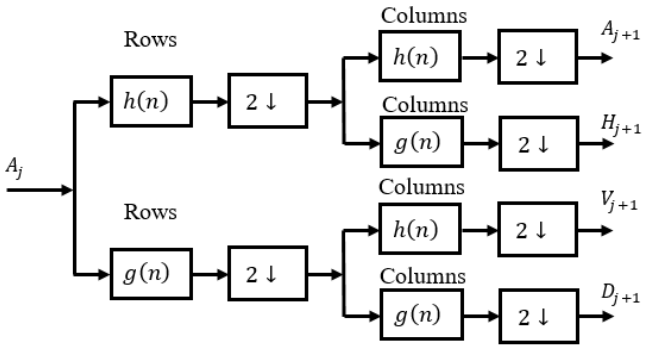


Figure 5.1: 2D discrete wavelet transform.

Paper [24] presents an improvement of the cryptographic algorithm from the literature, [26], using the wavelet packets and the tent map chaotic system. The permutation of the wavelet packets are used to replace the simple permutation of the pixels, while tent function replaces the Hénon system and is used for his uniform distribution which spread the pixels' values. The original scheme and the improved one are presented in Fig. 5.2.

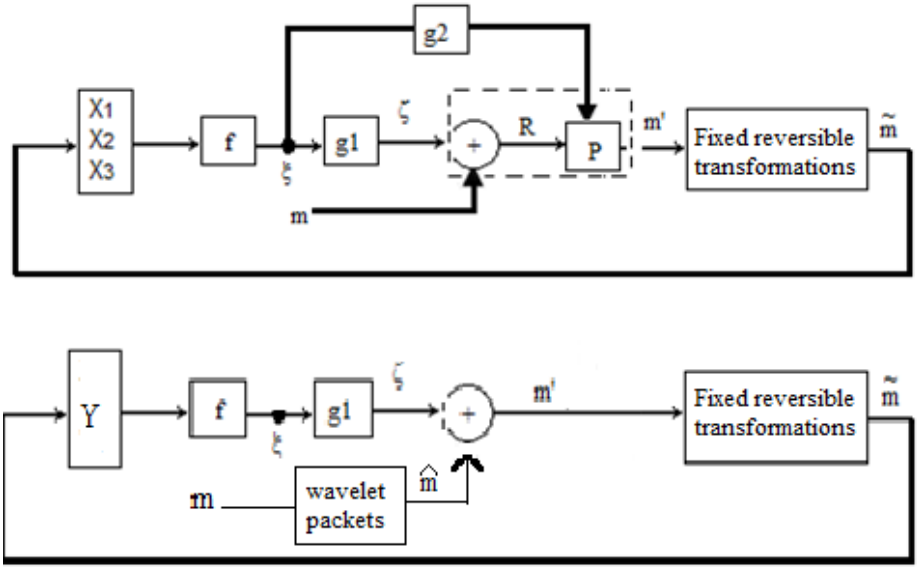


Figure 5.2: The original scheme (up) and the proposed one (down).

The significance of the proposed scheme is as follows:

- $[Y]$ represents the tent function. For p' being the control parameter and the initial condition $y(0)$ randomly chosen, results a particular realization of the random process.
- f corresponds to a random variable transformation that leads to a discrete random variable $\xi = f(sin(y))$, with $\xi \in 1, 2, ..., 10$.
- g_1 allocates to the random variable ξ an integer in the range $[0, 255]$, $g_1(\xi) = \zeta$.
- m is the original image.
- the permutation $P = g_2(\xi)$ from the previous algorithm is replaced by the decomposition of the image with wavelet packets using the Haar transform and the permutation of these packets with the logistic function, ec. 2.2.

- $\widehat{m}$ is the 8-bit ASCII representation of a pixel belonging to the image permuted with wavelet packets.
- m' is the cryptogram obtained after encryption, being a byte resulted from the *bitxor* operation between $\widehat{m}$ and ζ.
- the resulting binary message is converted to decimal and scaled by a ν factor; scaling allows the inclusion of the message in the evolution of the tent function without affecting its dynamics.
- $\widetilde{m}$ is the scaled message that is added to the evolution of the chaotic tent map.

We took a practical example, an image, the non-uniform histogram of which can be seen in Fig. 5.3. After decomposing it into wavelet packets, mixing them and recomposing the signal, results an image whose histogram is not uniform, so we can not yet speak of a satisfactory encryption.

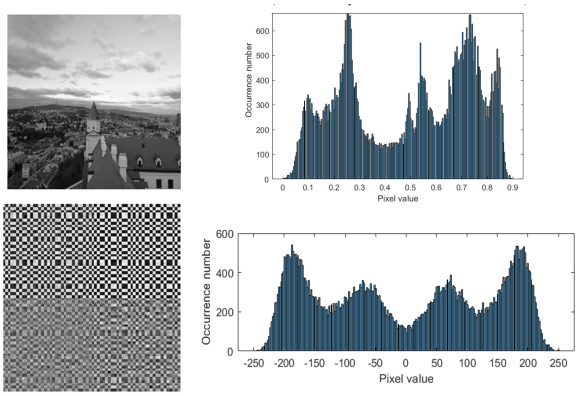


Figure 5.3: Original image and its histogram (up) & Encrypted image using wavelets and its histogram (down).

After including the image permuted with wavelet packets, in the evolution of the chaotic tent map, as we described in the presented algorithm, results a perfectly encrypted image, whose uniform histogram can be seen in Fig. 5.4. It is clear the improvement we bring compared to the histogram of the encrypted image with the reference algorithm [26]; in addition, the entropy of the encrypted image is close to the maximum value, 8, and the scatter plot shows that the pixels are dispersed across the all range of possible values.

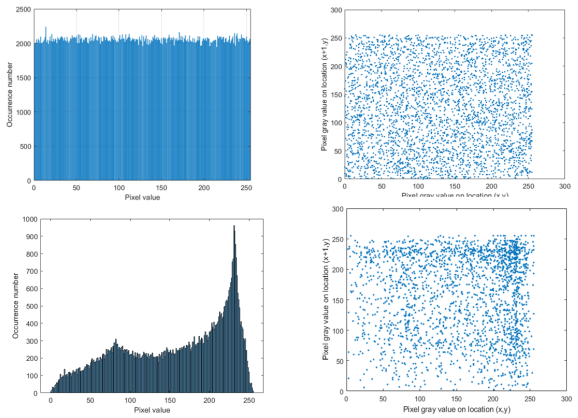


Figure 5.4: Histogram and correlation of the encrypted image with the proposed algorithm (up) vs. histogram and correlation of the encrypted image with the reference algorithm (down), [26].

Considering the vulnerabilities of the encryption algorithm presented in Chapter 4 - an algorithm in which a matrix generated with the chaotic logistic function multiplies with the clear message - an improvement was also proposed in this chapter by adding new steps, namely wavelet packet decomposition, permutation with the logistic function and the use of the 3D Arnold chaotic system. These additional steps slow down the attacker's efforts to break the cipher. The algorithm has been further improved. Thus, the permutation of wavelet packets using the logistic function was replaced with the permutation using the chaotic Baker function. Also, the generation of the ϕ matrix was no longer performed with the logistic function, but with the tent function. Several chaotic systems were thus comparatively tested.

Chapter 6

Conclusions, perspectives and original contributions

6.1 Conclusions and perspectives

Chapter 1 places this work in the larger domain it belongs to, that of statistical methods applied in the field of chaotic signals with the perspective of their use in cryptography and, more broadly, in the field of communications.

Chapter 2 presents the first contribution of the thesis, a mean test on the autocorrelation function, a test that supports the statistical independence distance corresponding to chaotic systems, studied in the previous articles of the team. Since we have not found works on this topic in the literature, it is considered an original study from both theoretical and experimental points of view. The study is customized on two chaotic systems in discrete time, the tent map and the logistic map. The sensitivity of the logistic map to the control parameter and the effect on the mean test is also studied. The variation of the parameter affects the results of the mean test and is in close connection with the statistical independence distance. It is confirmed the assumption that decorrelation appears faster than the statistical independence. I worked with double and extended precision, using a library for arbitrary arithmetic precision, concluding that the accuracy used in the calculations did not cause major changes in the proposed test results because the independence distance and decorrelation appear faster than the distance at which the trajectories begin to be altered, the pseudo-randomly behavior starting to degrade.

Chapter 3 proposes an algorithm to increase the performance of a pseudo-random number generator based on the generalized Hénon three-dimensional chaotic system, in discrete time. The algorithm dynamically changes the value of the bifurcation parameters a and b. The study was extended by a template of random number generators, configurable for various chaotic systems and various parameters. As a perspective, I aim to demonstrate that the generator is cryptographically efficient and can find its applicability in the real world.

Chapter 4 analyzes from the cryptanalyst's point of view an encryption scheme particularized on images. The focus is on determining the R control parameter corresponding to the matrix, the secret key, generated with the logistic map. Several statistical methods are applied for this purpose: Smirnov test, histograms, autocorrelation function, as well as the succesive iterations investigation. One perspective, still uninvestigated, is the dynamic change, at each k iteration, of the R parameter value corresponding to the ϕ matrix.

Chapter 5 investigates the benefits of wavelet packets in the pre-processing phase of the encryption scheme and their use together with chaotic systems. An algorithm in the literature is improved with the help of wavelet packets and the chaotic system represented by the tent function. Also, new steps are added to the encryption scheme from Chapter 4 to increase the robustness of the algorithm.

Chapter 6 presents the conclusions, perspectives and the author's curiosity to study the evolution of dynamic systems in continuous time, [27], [28].

The results of the research in this thesis continued the work of the team, and also represent future research directions.

6.2 Dissemination of research activity

6.2.1 Journal articles

1. Octaviana Datcu, **Corina Macovei**, Radu Hobincu, *"Chaos Based Cryptographic Pseudo-Random Number Generator Template with Dynamic State*

Change," Appl. Sci. 2020, https://doi.org/10.3390/app10020451, Published: 8 January 2020, https://www.mdpi.com/2076-3417/10/2/451, Applied Sciences-Basel, Vol. 10, Issue: 2, Article Number: 451, DOI: 10.3390/app10020451, JAN 2020, MDPI, ST Alban-Anlage 66, CH-4052 Basel, Switzerland, WOS 000522540400027, Impact Factor: 2.474, **Q2**.

2. **Corina Macovei**, Adina-Elena Lupu (Blaj), Mircea Răducanu, *"Enhanced cryptographic algorithm based on chaotic map and wavelet packets",* U.P.B. Sci. Bull., Series C Vol. 82, Iss. 4, 2020, ISSN 2286-3540

6.2.2 Conference articles

3. **Corina Macovei**, Alezandru Văduva, Adriana Vlad și Marta Zamfir, *"A mean test on the autocorrelation function of a chaotic signal aiming to support the statistical independence sampling distance,"* 2019 International Symposium on Signals, Circuits and Systems (ISSCS), Iasi, Romania, 2019, pp. 1-4, doi: 10.1109/ISSCS.2019.8801778, WOS:000503459500050

4. Radu Hobincu, Octaviana Datcu și **Corina Macovei**, *"Entropy global control for a chaos based pRNG,"* 2019 42nd International Conference on Telecommunications and Signal Processing (TSP), Budapest, Hungary, 2019, pp. 432-435. doi: 10.1109/TSP.2019.8768818. WOS:000493442800094

5. Octaviana Datcu, Radu Hobincu, **Corina Macovei**, *"Singular Value Decomposition to Determine the Dynamics of a Chaotic Regime Oscillator,"* 2019 International Semiconductor Conference, CAS Proceedings, Sinaia, Romania, October 2019, pp. 119-122, WOS:000514295300024, **Julkaisu conference 1**, https://www.imt.ro/cas/, https://www.tsv.fi/julkaisufoorumi/haku.php?nimeke=&konferenssilyh=CAS&issn=&tyyppi=kaikki&kieli=&maa=&wos=&scopus=&nappi=Search.

6. Octaviana Datcu, Radu Hobincu, **Corina Macovei**, *"Genetic Algorithms for high-Order sliding-Mode Observers,"* 2019 International Semiconduc-

tor Conference, Sinaia, CAS Proceedings, Romania, October 2019, pp. 305-308, WOS:000514295300064, **Julkaisu conference 1**, https://www.imt.ro/cas/, https://www.tsv.fi/julkaisufoorumi/haku.php?nimeke=&konferenssilyh=CAS&issn=&tyyppi=kaikki&kieli=&maa=&wos=&scopus=&nappi=Search.

7. **Corina Macovei**, Alexandru Văduva, Adriana Vlad și Bogdan Badea, *"The autocorrelation function of the logistic map chaotic signal in relation with the statistical independence issue"*, 13th International Conference on Communications (COMM), Bucuresti, Romania, June 2020, pp. 25-30, IEEE, doi: 10.1109/COMM48946.2020.9142000

8. **Corina Macovei**, Adina-Elena Lupu, Mircea Răducanu și Octaviana Datcu , *"Key extraction in a chaos-based image cipher and wavelet packets"*, ATOM-N 2020, The 10th edition of the International Conference "Advanced Topics in Optoelectronics, Microelectronics and Nanotechnologies," 20 - 23 August 2020, Constanta, Romania, **EXCELLENT PAPER AWARD**

9. **Corina Macovei**, Mircea Răducanu și Octaviana Datcu, *"Image encryption algorithm using wavelet packets and multiple chaotic maps"*, International Symposium on Electronics and Telecommunications (ISETC), IEEE, November 2020, Timisoara, Romania

6.2.3 Research reports

10. **Corina Macovei**, *"Analysis of statistical independence between sets of experimental data extracted from the tent function and the Hénon chaotic dynamic system."* (June 2015)

11. **Corina Macovei**, *"Security application model that tracks the user profile considering personal information protection."* (December 2015)

12. **Corina Macovei**, *"Statistical study on the use of binary sequences generated by tent function in cryptographic applications. Illustration on images."* (June 2016)

13. **Corina Macovei**, *"Study on the statistical autocorrelation function associated with the chaotic tent map signal in relation with the statistical independence distance."* (December 2016)

14. **Corina Macovei**, *"Simulations with various control parameters pairs of the dynamic Hénon system to test a pseudorandom number generator."* (June 2017)

6.2.4 Summer school, symposium, internship, workshop

15. *"Security and Privacy in Digital Life 2015"*, Summer School Privacy, Security and Trust, June 29th - July 10th 2015, Trento, Italy

16. **Corina Macovei**, Adriana Vlad și Marta Zamfir, *"On the autocorrelation function of the skew tent map chaotic signal in relation with the statistical independence sampling distance,"* Annual Symposium of Doctoral School ETTI-B, SADETTI, 2018, Bucharest, Romania

17. Stagiu la Applications École Nationale Supérieure de l'Électronique et de ses Applications (ENSEA), Cergy, France, 24 May -14 June 2019

18. **Corina Macovei**, Adina-Elena Blaj, Octaviana Datcu, Radu Hobincu, *"Cryptanalysis of a compressive sensing communication scheme,"* Workshop "Analyse du chaos et applications," Applications École Nationale Supérieure de l'Électronique et de ses Applications (ENSEA), Cergy, France, December 2019

Bibliography

[1] The butterfly effect and the maths of chaos, Maths careers and The Institute of mathematics and its applications, online on 30th of June 2020 at https://www.mathscareers.org.uk/article/the-butterfly-effect-and-the-maths-of-chaos/

[2] C. Macovei, A. Văduva, A. Vlad and M. Zamfir, "A mean test on the autocorrelation function of a chaotic signal aiming to support the statistical independence sampling distance," in International Symposium on Signals, Circuits and Systems (ISSCS2019), Iaşi, 2019, pp. 1-4, IEEE.

[3] C. Macovei, A. Văduva, A. Vlad and B. Badea, "The autocorrelation function of the logistic map chaotic signal in relation with the statistical independence issue," In 2020 13th International Conference on Communications (COMM) (pp. 25-30). IEEE.

[4] A. Leontitsis (2020). Spearman Rank Correlation, `https://www.mathworks.com/matlabcentral/fileexchange/4374-spearman-rank-correlation`, MATLAB Central File Exchange. Retrieved July 8, 2020.

[5] Correlation coefficients, Mathworks, online at `https://www.mathworks.com/help/matlab/ref/corrcoef.html` on 8 July 2020.

[6] B. Badea and A. Vlad, "Revealing statistical independence of two experimental data sets: an improvement on Spearman's algorithm," in International Conference on Computational Science and Its Applications, pp. 1166-1176, Springer, Berlin, Heidelberg.

[7] A. Văduva, A. Vlad and B. Badea, "Evaluating the performance of a test-method for statistical independence decision in the context of chaotic signals," In 2016 International Conference on Communications (COMM), pp. 417-422, IEEE, Bucharest, Romania

[8] A. Luca, A. Vlad, B. Badea and M. Frunzete, "A study on statistical independence in the tent map," In 2009 International Symposium on Signals, Circuits and Systems (pp. 1-4). IEEE.

[9] A.Vlad, A. Luca and M. Frunzete, "Computational measurements of the transient time and of the sampling distance that enables statistical independence in the logistic map," in O. Gervasi et al. (eds) Computational Science and Its Applications – ICCSA 2009, Lecture Notes in Computer Science, vol 5593, pp. 703-718, Springer, 2009.

[10] O. Hodea and A. Vlad, "Logistic map sensitivity to control parameter and its implications in the statistical behaviour," In International Symposium on Signals, Circuits and Systems ISSCS2013, pp. 1-4, IEEE, Iasi, Romania

[11] R. Hobincu, O. Datcu and C. Macovei, (2019, July). "Entropy global control for a chaos based pRNG," In 2019 42nd International Conference on Telecommunications and Signal Processing (TSP), pp. 432-435, IEEE, Budapest, Hungary

[12] O. Datcu, C. Macovei and R. Hobincu, (2020). "Chaos Based Cryptographic Pseudo-Random Number Generator Template with Dynamic State Change," Applied Sciences, 10(2), 451, doi: 10.3390/app10020451

[13] R. Hobincu, O. Datcu, "A novel Chaos Based PRNG Targeting Secret Communication," 12th International Conference on Communications (COMM),pp. 459-462, Bucharest, Romania, 2018, WOS: 000449526000086.

[14] R. Hobincu, O. Datcu, "FPGA Implementation of a Chaos Based PRNG Targeting Secret Communication," 13th Symposium on Electronics and

Telecommunications (ISETC 2018), pp. 1-4, Timişoara, Romania, WOS: 000463031500052.

[15] D.A. Miller, G. Grassi, "A discrete generalized hyperchaotic Hénon map circuit," Proceedings of the 44th IEEE 2001 Midwest Symposium on Circuits and Systems, Dayton, OH, 2001, 328-331, Vol. 1.

[16] R. Hobincu, O. Datcu, "NIST tests versus bifurcation diagrams and Lyapunov exponents when evaluating chaos-based pRNGs," (ITISE 2018) International Conference on Time Series and Forecasting Proceedings of Papers, Granada, Spain, ISBN: 978-84-17293-57-4.

[17] A. Suciu, R.A. Toma, K. Marton, "Parallel implementation of the TestU01 statistical test suite," In Proceedings of the 2012 IEEE 8th International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Romania, 30 August–1 September 2012; pp. 317–322.

[18] M. Matsumoto, T. Nishimura, "Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator." ACM Trans. Model. Comput. Simul. (TOMACS) 1998, 8, 3–30.

[19] D. J. Bernstein, "ChaCha, A Variant of Salsa20." Workshop Record of SASC 2008: The State of the Art of Stream Ciphers. Available online: https://cr.yp.to/chacha/chacha-20080120.pdf (accessed on 25 November 2020).

[20] L. Yu, J.-P. Barbot, G. Zheng and H. Sun, (2010). "Compressive sensing with chaotic sequence". IEEE Signal Processing Letters, 17(8), 731-734.

[21] L. Yu, J.-P. Barbot, G. Zheng and H. Sun, (2010, July). "Toeplitz-structured chaotic sensing matrix for compressive sensing". In 2010 7th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP 2010) (pp. 229-233). IEEE.

[22] M. Frunzete, L. Yu, J.-P. Barbot and A. Vlad, (2011, September). "Compressive sensing matrix designed by tent map, for secure data transmission". In Signal

Processing Algorithms, Architectures, Arrangements, and Applications SPA 2011 (pp. 1-6). IEEE.

[23] C. Macovei, A. Lupu, M. Răducanu and O. Datcu, "Key extraction in a chaos-based image cipher with wavelet packets," ATOM-N 2020, The 10th edition of the International Conference "Advanced Topics in Optoelectronics, Microelectronics and Nanotechnologies," 20 - 23 August 2020, Constanta, Romania.

[24] C. Macovei, A. Lupu and M. Răducanu, "Enhanced cryptographic algorithm based on chaotic map and wavelet packets," U.P.B. Sci. Bull., Series C Vol. 82, Iss. 4, 2020, ISSN 2286-3540

[25] C. Macovei, M. Răducanu and O. Datcu, "Image encryption algorithm using wavelet packets and multiple chaotic maps," International Symposium on Electronics and Telecommunications (ISETC), IEEE, Timisoara, Romania, November 2020.

[26] O. Datcu, J.-P. Barbot, A. Vlad, "New enciphering algorithm based on Chaotic Generalized Henon Map", in Modeling, Simulation and Applications Selected Papers from the 3rd Chaotic Modeling and Simulation International Conference CHAOS2010) edited by Christos H. Skiadas, Ioannis Dimotikalis & Charilaos Skiadas, World Scientific, Singapore, 2011, ISBN: 978-981-4350-33-4

[27] O. Datcu, R. Hobincu, C. Macovei, "Singular Value Decomposition to Determine the Dynamics of a Chaotic Regime Oscillator," 2019 International Semiconductor Conference, Sinaia, Romania, October 2019, pp. 119-122, WOS:000514295300024, JULKAISU conference 1.

[28] O. Datcu, R. Hobincu, C. Macovei, "Genetic Algorithms for high-Order sliding-Mode Observers," In 2019 International Semiconductor Conference, Sinaia, Romania, October 2019, pp. 305-308, WOS:000514295300064, JULKAISU conference 1.