ESAIM: PROCEEDINGS AND SURVEYS, November 2014, Vol. 46, p. 177-195
ECIT 2012 – Witold Jarczyk, Daniele Fournier-Prunaret, João Manuel Gonçalves Cabral

FROM CHAOS TO RANDOMNESS VIA GEOMETRIC UNDERSAMPLING

RENE LOZI[1] AND INA TARALOVA[2]

Abstract. We propose a new mechanism for undersampling chaotic numbers obtained by the ring coupling of one-dimensional maps. In the case of 2 coupled maps this mechanism allows the building of a PRNG which passes all NIST Test. This new geometric undersampling is very effective for generating 2 parallel streams of pseudo-random numbers, as we show, computing carefully their properties, up to sequences of 10^{12} consecutives iterates of the ring coupled mapping which provides more than 3.35×10^{10} random numbers in very short time.
Keywords: Cryptography, Chaos, Randomness, Under-sampling, Pseudo-random number generator.

Résumé. Nous proposons un nouveau mécanisme de sous-échantillonnage de nombres chaotiques obtenus par couplage en anneau de fonctions unidimensionnelles. Dans le cas de 2 fonctions couplées, ce mécanisme permet de construire un Générateur de Nombres Pseudo- Aléatoires (GNPA) qui satisfait tous les tests NIST. Ce nouveau sous-échantillonnage géométrique est très efficace pour générer deux séries parallèles de nombres pseudo-aléatoires, comme nous le montrons en étudiant très soigneusement leurs propriétés pour des suites de nombres allant jusqu'à 10^{12} nombres consécutifs de l'application couplée en anneau, ce qui fournit plus de 3.35×10^{10} nombres aléatoires en un temps très court.
AMSC: 9A20, 11K45, 37H10, 65P20
Mots-clefs: Cryptographie, Chaos, Aléatoire, Sous-échantillonnage, Générateurs de nombres pseudo-aléatoires.

Contents

[1] Université de Nice Sophia-Antipolis, Laboratoire J. A. Dieudonné, UMR CNRS 7351, Parc Valrose, 06108 NICE, Cedex 02, France, email : rlozi@unice.fr
[2] L'UNAM, IRCCyN, UMR CNRS 6597, Ecole Centrale de Nantes, 1, rue de la Noë, BP 92101, 44321, NANTES Cedex 3, France, email : Ina.Taralova@irccyn.ec-nantes.fr

Article published online by EDP Sciences and available at http://www.esaim-proc.org or http://dx.doi.org/10.1051/proc/201446015

Introduction

During the last decade, it has been emphasized that the undersampling of sequence of chaotic numbers is an efficient tool in order to build pseudo-random number generators (PRNG) [1]. Randomness appears to be an emergent property of complex systems of coupled chaotic maps [2]. Several kinds of coupling can be considered as ultra-weak coupling or ring coupling,... . An ultra-weak coupling recovers chaotic properties of 1-dimensional maps [3] when computed with floating numbers or double precision numbers. Chaotic undersampling with thresholds based on one component of the coupled system adds random properties to the chaotic sequences. Double threshold sampled sequence (i.e., using both thresholds of different nature) improves such random properties [4]. Ring coupling deals when p 1-dimensional maps are constrained on a torus [5], this coupling can directly provide random numbers, without sampling or mixing, provided the number p of maps is large enough, although it is possible to combine these processes with it. However in lower dimension 2 and 3, the chaotic numbers are not equidistributed on the torus. Therefore we introduce a particular "geometric" undersampling based on the property of piecewise linearity of the invariant measure of the system. This new geometric undersampling is very effective for generating parallel streams of pseudo-random numbers with a very compact mapping.

In Section 1 we briefly recall properties of chaotic mappings, when used alone or ultraweakly coupled. Section 2 describes the route from chaos to randomness via chaotic undersampling, discovered during the last decade. In Section 3, we introduce geometric undersampling in the scope of ring coupled mapping.

1. COMPUTATION OF CHAOTIC NUMBERS

1.1. **Disappointing chaotic numbers**

Chaos theory studies the behavior of dynamical systems that are highly sensitive to initial conditions, an effect which is popularly referred to as the butterfly effect. Small differences in initial conditions (such as those due to rounding errors in numerical computation) yield widely diverging outcomes for chaotic systems, rendering long-term prediction impossible in general. This happens even though these systems are deterministic, meaning that their future behavior is fully determined by their initial conditions, with no random elements involved. In other words, the deterministic nature of these systems does not make them predictable. The first example of such chaotic continuous system in the dissipative case was pointed out by the meteorologist E. Lorenz in 1963 [6].

In order to study numerically the properties of the Lorenz attractor, M. Hénon an astronomer of the observatory of Nice, France, introduced in 1976 a simplified model of the Poincaré map of this attractor [7]. The Lorenz attractor being imbedded in dimension 3, the corresponding Poincaré map is a mapping from the plane $\mathbb{R}^2$ into $\mathbb{R}^2$. Hence the Hénon mapping is also defined in dimension 2 and is associated to the dynamical system

$$\begin{cases} x_{n+1} = y_n + 1 - ax_n^2, \\ y_{n+1} = bx_n. \end{cases} \tag{1}$$

which has been extensively studied since thirty six years. More simple dynamical systems in dimension one, on the interval $J = [-1, 1] \subset \mathbb{R}$ into itself

$$x_{n+1} = f_a(x_n) \tag{2}$$

corresponding to the logistic map

$$f_a \equiv L_a(x) = 1 - ax^2 \tag{3}$$

or the symmetric tent map

$$f_a \equiv T_a(x) = 1 - a|x| \tag{4}$$

have also been fully explored in the hope of generating random numbers easily [8]. However when a dynamical system is realized on a computer using floating point or double precision numbers, the computation is of a discretization, where finite machine arithmetic replaces continuum state space. For chaotic dynamical systems in small dimension, the discretization often has collapsing effects to a fixed point or to short cycles [9, 10].

It seems that the computation of numerical approximations of the periodic orbits leads to unpredictable and somewhat enigmatic results. As says O. E. Lanford III [9] "The reason is that because of the expansivity of the mapping the growth of roundoff error normally means that the computed orbit will remain near the true orbit with the chosen initial condition only for a relatively small number of steps typically of the order of the number of bits of precision with which the calculation is done. It is true that the above mapping like many 'chaotic' mappings satisfies a shadowing theorem (see [11, 12]) which ensures that the computed orbit stays near to some true orbit over arbitrarily large numbers of steps. The flaw in this idea as an explanation of the behavior of computed orbits is that the shadowing theorem says that the computed orbit approximates some true orbit but not necessarily that it approximates a typical one."

The collapsing of iterates of dynamical systems or at least the existence of very short periodic orbits, their non constant invariant measure, and the easily recognized shape of the function in the phase space avoid the use of one-dimensional map (logistic, baker, or tent, ...) as a Pseudo Random Number Generator (see [13] for a survey).

Remark 1.1 However, the very simple implementation in computer program of chaotic dynamical systems led some authors to use it as a base of cryptosystem [14, 15]. In addition it seems that for some applications, chaotic numbers are more efficient than random numbers, that is the case for evolutionary algorithms [16, 17] or chaotic optimization [18].

In this paper we show how to overcome the poor quality of chaotic generators using geometric undersampling. This special undersampling we introduce in this article is one among others undersampling processes we have studied before. In order to explain the difference between these processes we give in Sec. 2 a brief survey of them. Before doing this survey, we have to show how to stabilize the chaotic properties of chaotic number when realized on a computer.

1.2. **Long periodic orbits for ultra-weakly coupled tent map**

The first step in order to preserve the genuine chaotic properties of the continuous models in numerical experiments is reached considering ultra-weak multidimensional coupling of p one-dimensional dynamical systems instead of solely a one-dimensional map.

1.2.1. *System of 2-coupled symmetric tent map*

In order to simplify the presentation below, we use as an example the symmetric tent map (4) with the parameter value $a = 2$, later denoted simply as f, even though others chaotic map of the interval (as the logistic map, the baker transform, ...) can be used for the same purpose (as a matter of course, the invariant measure of the chaotic map considered is preserved).
When $p = 2$, the system is simply described by Eq. (5)

$$\begin{cases} x_{n+1} = (1 - \varepsilon_1) f(x_n) + \varepsilon_1 f(y_n), \\ y_{n+1} = \varepsilon_2 f(x_n) + (1 - \varepsilon_2) f(y_n). \end{cases} \tag{5}$$

We use generally $\varepsilon_1 = 10^{-7}$, $\varepsilon_2 = 2\varepsilon_1$ when computations are done using floating points or $\varepsilon_1 = 10^{-14}$ for double precision numbers. In both cases, with these numerical values, the collapsing effect disappears and the invariant measure of any component is the Lebesgue measure [3] as we show below. In the case of computation using floating points, starting form most initial condition, it is possible to find a Mega-Periodic orbit (i.e. with period equal to 1,320,752). When computations are done with double precision number it is not possible to find any periodic orbit, up to $n = 5 \times 10^{11}$ iterations. In [3] the computations have been performed on a Dell computer with a Pentium IV microprocessor using a Borland C compiler computing with ordinary (IEEE-754) double precision numbers.

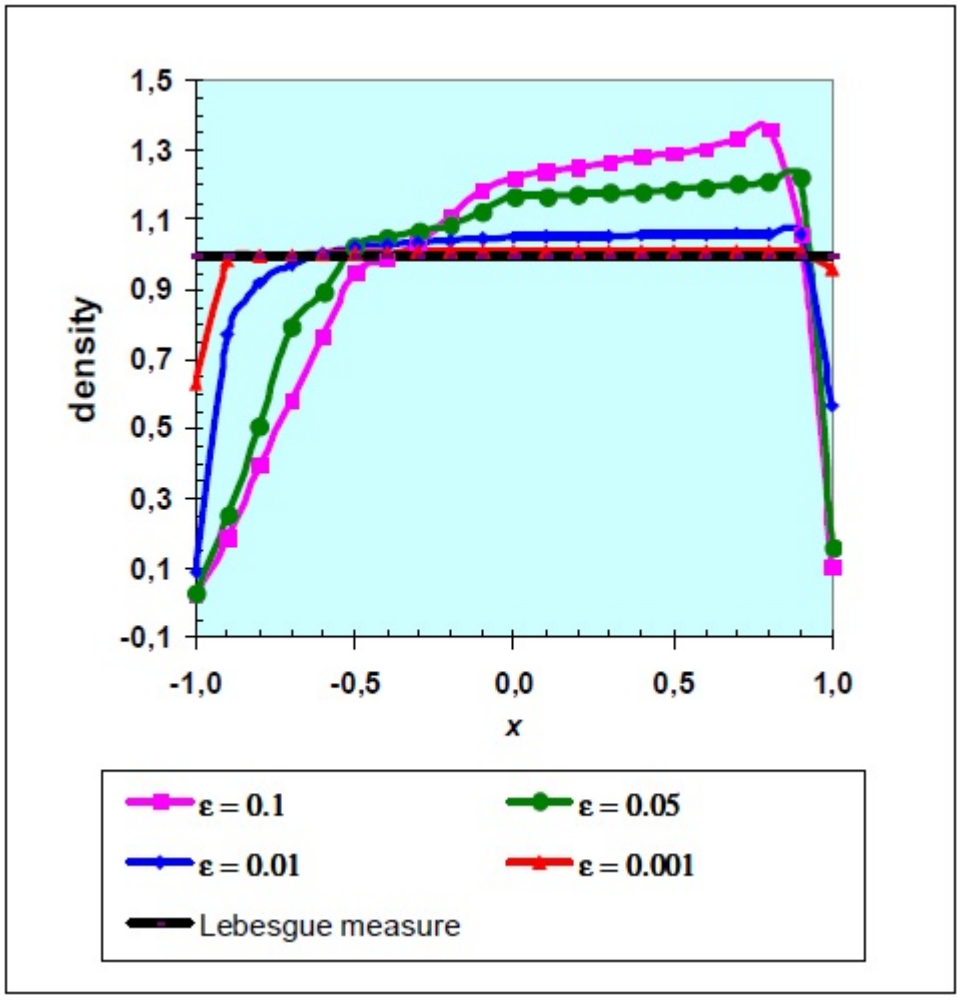


FIGURE 1. Density of iterates of 2-coupled symmetric tent maps, double precision, $N_{disc} = 10^5, \varepsilon_2 = 2\varepsilon_1, \varepsilon_1 = 10^{-1}$ to $10^{-3}, N_{iter} = 10^8$, initial values $x_0 = 0.330, y_0 = 0.3387564$

When ε_1 converges towards 0, the iterates of each component x_n and y_n of equation (5) converge to the Lebesgue measure (Fig. 1).

1.2.2. *System of p-coupled symmetric tent map*

More generally, the coupling of p maps takes the form

$$X_{n+1} = F(X_n) = A \cdot (\underline{f}(X_n)), \tag{6}$$

where

$$\underline{f}(X_n) = \begin{pmatrix} f(x_n^1) \\ \vdots \\ \vdots \\ f(x_n^p) \end{pmatrix}, X_n = \begin{pmatrix} x_n^1 \\ \vdots \\ \vdots \\ x_n^p \end{pmatrix}, \tag{7}$$

and

$$A = \begin{pmatrix} \varepsilon_{1,1} = 1 - \sum_{j=2}^{j=p} \varepsilon_{1,j} & \varepsilon_{1,2} & \cdots & \varepsilon_{1,p-1} & \varepsilon_{1,p} \\ \varepsilon_{2,1} & \varepsilon_{2,2} = 1 - \sum_{j=2, j\neq 2}^{j=p} \varepsilon_{2,j} & \cdots & \varepsilon_{2,p-1} & \varepsilon_{2,p} \\ \vdots & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \vdots & \vdots \\ \varepsilon_{p,1} & \vdots & \vdots & \varepsilon_{p,p-1} & \varepsilon_{p,p} = 1 - \sum_{j=1}^{j=p-1} \varepsilon_{p,j} \end{pmatrix}, \tag{8}$$

with $\varepsilon_{i,i} = 1 - \sum_{j=1, j\neq 1}^{j=p} \varepsilon_{i,j}$ on the diagonal (the matrix A is always a stochastic matrix if the coupling constants verify $\varepsilon_{i,j} > 0$ for every i and j).

It is noteworthy that these families of very weakly coupled maps are more powerful than the usual formulas used to generate random sequences, mainly because only additions and multiplications are used in the computation process, no division being required. Moreover the computations are done using floating point or double precision numbers, allowing the use of the powerful Floating Point Unit (FPU) of the modern microprocessors. In addition, a large part of the computations can be parallelized taking advantage of the multicore microprocessors which appear on the market of laptop computers.

Moreover, a determining property of such coupled map is the high number of parameters used ($p \times (p-1)$) for p coupled equations) which allows to choose it as cipher-keys, when used in chaos based cryptographic algorithms, due to the high sensitivity to the parameters values [2]. It at be shown also, that using control theory techniques, synchronization of two systems (6), with $p = 2$ or 3, can be reached via exact (dead-beat) or asymptotic observers [19].

1.2.3. *Approximated distribution function*

In order to assess numerical computations more accurately, we define an approximation $P_{M,N}(x)$ of the invariant measure also called the probability distribution function linked to the 1-dimensional map f, when computed with floating numbers (or numbers in double precision). In this scope we consider a regular partition of M small intervals (boxes) r_i of J defined by

$$s_i = -1 + \frac{2i}{M}, i = 0, M \tag{9}$$

$$r_i = [s_i, s_{i+1}[, \quad i = 0,\ M-2 \quad \textit{and} \quad r_{M-1} = [s_{M-1}, 1] \tag{10}$$

the length of each box is equal to $\frac{2}{M}$ and the r_i intervals form a partition of the interval J

$$J = \bigcup_{0}^{M-1} r_i \tag{11}$$

All iterates $f^{(n)}(x)$ belonging to these boxes are collected, after a transient regime of Q iterations decided *a priori*, (i.e. the first Q iterates are neglected). Once the computation of $N + Q$ iterates is completed, the relative number of iterates with respect to N/M in each box r_i represents the value $P_N(s_i)$. The approximated $P_N(x)$ defined in this article is then a step function, with M steps. As M may vary, we define

$$P_{M,N}(s_i) = \frac{M}{N}(\sharp r_i) \tag{12}$$

where $\sharp r_i$ is the number of iterates belonging to the interval r_i. $P_{M,N}(x)$ is normalized to 2 on the interval J.

$$P_{M,N}(x) = P_{M,N}(s_i), \forall x \in r_i \tag{13}$$

In the case of p-coupled maps, we are more interested by the distribution of each component $x^1, x^2, x_1^2, ..., x^p$ of X rather than the distribution of the variable X itself in J^p. We then consider the approximated probability distribution function , $P_{M,N}(x^j)$ associated to one among several components of $F(X)$ defined by (6), which are one-dimensional maps. In this paper we use equally N_{disc} for M and N_{iter} for N, when they are more explicit.

The discrepancies E_1 (in norm L_1), E_2(in norm L_2) and E_∞ (in norm L_∞) between $P_{N_{disc},N_{iter}}(x^j)$ and the Lebesgue measure, which is the invariant measure associated to the symmetric tent map, are defined by

$$E_{1,N_{disc},N_{iter}}(x^j) = \|P_{N_{disc},N_{iter}}(x^j) - 1\|_{L_1} \tag{14}$$

$$E_{2,N_{disc},N_{iter}}(x^j) = \|P_{N_{disc},N_{iter}}(x^j) - 1\|_{L_2} \tag{15}$$

$$E_{\infty,N_{disc},N_{iter}}(x^j) = \|P_{N_{disc},N_{iter}}(x^j) - 1\|_{L_\infty} \tag{16}$$

As previously said, Fig. 1 shows the convergence of the density of iterates of the components of 2-coupled symmetric tent maps to the Lebesgue measure when ε_1 converges towards 0. Moreover, for a fixed value of N_{disc} when the number N_{iter} increases, the discrepancy between $P_{N_{disc},N_{iter}}(x^j)$ and the Lebesgue measure is expected to converge towards 0, except if there exist periodic orbits of finite length lower than N_{iter} which capture the iterates. In this case whatsoever the value of N_{iter} is, the approximated distribution function converges to the distribution function of the periodic orbit, if it is unique, or to the average of the distribution functions of the periodic orbits observed, if not.

Figure 2 shows the errors $E_{1,N_{disc},N_{iter}}(x^j)$ versus the number of iterates of the approximated distribution functions, with respect to the first variable x^1, for 2 and 3-coupled symmetric tent map. Same results are obtained for the other variables x^2 or x^3 .

The 3-coupled symmetric tent maps model considered here with very very small value of ε_1 , seems a sterling model of generator of chaotic numbers with a uniform distribution of these numbers over the interval J. It produces very long periodic orbits: Gigaperiodic orbits (i.e. with length of period between 10^9 and 10^{12}) when

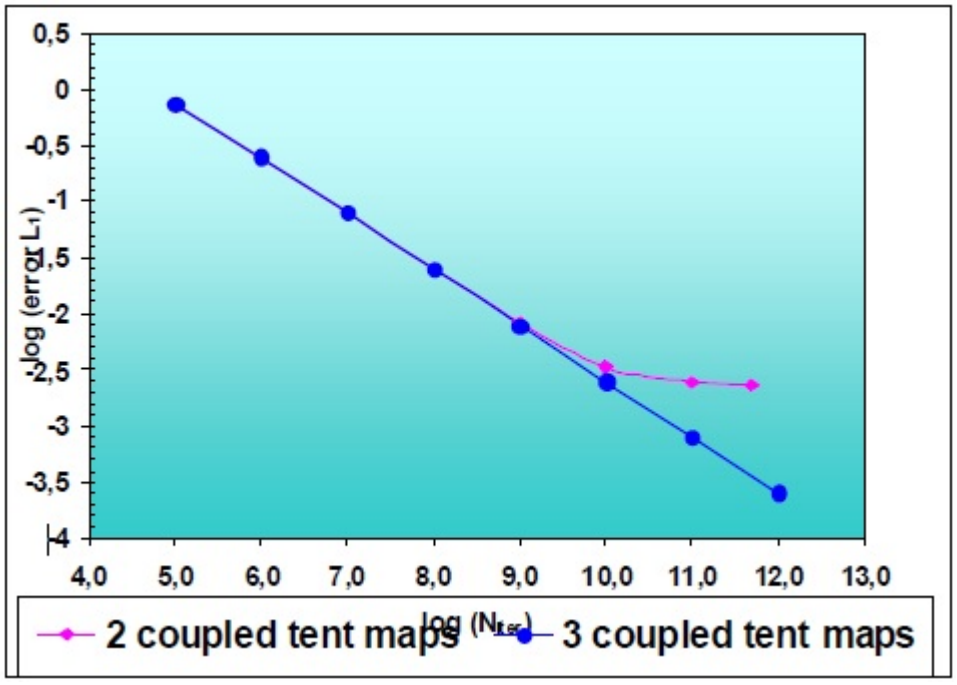


FIGURE 2. Error $E_{1,N_{disc},N_{iter}}(x^j)$ for 2 and 3-coupled symmetric tent map, double precision, $N_{disc} = 10^5, \varepsilon_1 = 10^{-14}$, $\varepsilon_2 = 2\ \varepsilon_1$, $N_{iter} = 10^5$ *to* 10^{12}, initial values $x_0^1 = 0.330, \quad x_0^2 = 0.338764, \quad x_0^3 = 0.331353429$

computed with simple precision numbers, and orbits of unknown length when computed with double precision numbers. However these chaotic sequences are not at all random sequences.

2. THE ROUTE FROM CHAOS TO RANDOMNESS VIA CHAOTIC UNDERSAMPLING

Chaotic numbers are not pseudo-random numbers because the plot of the couples of any component (x_n^l, x_{n+1}^l) of iterated points (X_n, X_{n+1}) in the corresponding phase plane reveals the map f used as one-dimensional dynamical systems to generate them via Eq. (6). Nevertheless, we have recently introduced a family of enhanced Chaotic Pseudo Random Number Generators (CPRNG) in order to compute faster long series of pseudorandom numbers with desktop computer [1, 4]. This family is based on the previous ultra weak coupling which is improved in order to conceal the chaotic genuine function.

In this Section, we describe briefly how works this first process of undersampling, the chaotic one.

2.1. **Chaotic undersampling**

In order to hide f in the phase space (x_n^l, x_n^l) two mechanisms are used. The pivotal idea of the first one mechanism is to sample chaotically the sequence $(x_0^l, x_1^l, x_2^l, ..., x_n^l, x_{n+1}^l, ...)$ generated by the l-th component x^l , selecting x_n^l every time the value x_n^m of the m-th component x^m, is strictly greater (or smaller) than a threshold $T \in J$, with $l \neq m$, for $1 \leq l, m \leq p$.
That is to say to extract the subsequence $(x_{n_{(0)}}^l, x_{n_{(1)}}^l, x_{n_{(2)}}^l, ..., x_{n_{(q)}}^l, x_{n_{(q+1)}}^l, ...)$ denoted here $(\overline{x_0}, \overline{x_1}, \overline{x_2}, ..., \overline{x_q}, \overline{x_{q+1}}, ...)$ of the original one, in the following way given $1 \leq l, m \leq p, l \neq m$

$$\begin{cases} n_{(-1)} = -1 \\ \overline{x_q} = x^l_{n_{(q)}}, \quad with \quad n_{(q)} = Min\{r > n_{(q-1)} | x^m_r > T,\ r \in \mathbb{N}\} \end{cases} \tag{17}$$

The sequence $(\overline{x_0}, \overline{x_1}, \overline{x_2}, ..., \overline{x_q}, \overline{x_{q+1}}, ...)$ is then the sequence of chaotic pseudo-random numbers.

The above mathematical formula can be best understood in algorithmic way. The pseudocode, for computing iterates of (17) corresponding to N iterates of (6) is:

$X_0 = (x^1_0, x^2_0, ..., x^{p-1}_0, x^p_0) = seed;$
$n = 0; \quad q = 0$
$do \quad \{while \quad n < N$
$\qquad do \quad \{while \quad (x^m_n \leq T) \quad compute \quad (x^1_n, x^2_n, ..., x^{p-1}_n, x^p_n); \quad n++\}$
$\quad compute \quad (x^1_n, x^2_n, ..., x^{p-1}_n, x^p_n); \quad then \quad n(q) = n; \quad \overline{x_q} = x^j_{n_{(q)}}; \quad n++;\ q++\}$

This chaotic sampling is possible due to the independence of each component of the iterated points X_n versus the others [20].

Remark 2.1 Albeit the number $NSampl_{iter}$ of pseudo-random numbers $\overline{x_q}$ corresponding to the computation of N iterates is not known *a priori*, considering that the selecting process is again linked to the uniform distribution of the iterates of the tent map on J, this number is equivalent to $\frac{2N}{1-T}$.

2.2. **Chaotic mixing**

A second mechanism can improve the unpredictability of the pseudo-random sequence generated as above, using synergistically all the components of the vector X_n, instead of two. Given $p-1$ thresholds

$$T_1 < T_2 < ... < T_{p-1} \in J \tag{18}$$

and the corresponding partition $J_0 = [-1, T_1], \quad J_1 =]T_1, T_2[, \quad J_k = [T_k, T_{k+1}[\quad for \quad 1 < k < p-1 \quad and$ $J_{p-1} = [T_{p-1}, 1[$, with

$$J = \bigcup_{k=0}^{p-1} J_k \tag{19}$$

(note that this partition of J is different from the regular previous one (11) used for the approximated distribution function). The second simple mechanism is based on the chaotic undersampling combined with a chaotic mixing of the $p-1$ sequences
$(x^1_0, x^1_1, x^1_2, ..., x^1_n, x^1_{n+1}, ...)$, $(x^2_0, x^2_1, x^2_2, ..., x^2_n, x^2_{n+1}, ...)$, $(x^{p-1}_0, x^{p-1}_1, x^{p-1}_2, ..., x^{p-1}_n, x^{p-1}_{n+1}, ...)$
...
using the last one $(x^p_0, x^p_1, x^p_2, ..., x^p_n, x^p_{n+1}, ...)$ in order to distribute the iterated points with respect to this given partition, defining the subsequence $(\overline{x_0}, \overline{x_1}, \overline{x_2}, ..., \overline{x_q}, \overline{x_{q+1}}, ...)$ (in pseudocode) by

$X_0 = (x^1_0, x^2_0, ..., x^{p-1}_0, x^p_0) = seed;$
$n = 0; \quad q = 0$
$do \quad \{while \quad n < N$
$\qquad do \quad \{while \quad (x^p_n \in J_0) \quad compute \quad (x^1_n, x^2_n, ..., x^{p-1}_n, x^p_n); \quad n++\}$
$\qquad compute \quad (x^1_n, x^2_n, ..., x^{p-1}_n, x^p_n);$
$\quad let \quad k \quad be\ such\ that\ x^p_n \in J_k;\ then\ n(q) = n; \quad \overline{x_q} = x^k_{n_{(q)}}; \quad n++;\ q++\}$

Remark 2.2 In this case also, $NSampl_{iter}$ is not known *a priori*, however, considering that the selecting process is linked to the uniform distribution of the iterates of the tent map on J, one has $NSampl_{iter} \approx \dfrac{2N}{1-T_1}$

Remark 2.3 This second mechanism is more or less linked to the whitening process [21, 22].

Remark 2.4 Actually, one can choose any of the components in order to sample and mix the sequence, not only the last one.

2.3. **Enhanced chaotic undersampling**

On can eventually improve the CPRNG previously introduced with respect to the infinity norm instead of the L_1 or L_2 norms because the L_∞ norm is more sensitive than the others ones to reveal the concealed f [4]. For this purpose we introduce a second kind of threshold $T^{'} \in N$, together with $T_1, ..., T_{p-1} \in J$ such that the subsequence is $(\overline{x_0}, \overline{x_1}, \overline{x_2}, ..., \overline{x_q}, \overline{x_{q+1}}, ...)$ defined (in pseudo-code) by

$X_0 = (x_0^1, x_0^2, ..., x_0^{p-1}, x_0^p) = seed;$
$n = 0; \quad q = 0$
$do \quad \{while \quad n < N$

$do \quad \{while \quad (n \leq n_{(q-1)} + T^{'} \; and \; x_n^p \in J_0)$
$compute \quad (x_n^1, x_n^2, ..., x_n^{p-1}, x_n^p); \quad n++\}$
$compute \quad (x_n^1, x_n^2, ..., x_n^{p-1}, x_n^p);$
$let \quad k \quad be \; such \; that \; x_n^p \in J_k;$
$then \; n(q) = n; \quad \overline{x_q} = x^k_{n_{(q)}}; \quad n++; \; q++\}$

Remark 2.5 In this case also, $NSampl_{iter}$ is not known *a priori*, it is more complicated to give an equivalent to it. However, considering that the selecting process is linked to the uniform distribution of the iterates of the tent map on J, and to the second threshold $T^{'}$, it comes that
$NSampl_{iter} \leq Min\{\frac{2N}{1-T_1}, \frac{N}{T'}\}$.

Remark 2.6 The second kind of threshold $T^{'}$ can also be used with only the chaotic sampling, without the chaotic mixing.

2.4. **A window of emergence of randomness**

In [1,2] we show that if one consider the errors $E_{1,N_{disc},N_{iter}}(x) = \|P_{N_{disc},N_{iter}}(x) - 1\|_{L_1}$, $E_{2,N_{disc},N_{iter}}(x^j) = \|P_{N_{disc},N_{iter}}(x^j) - 1\|_{L_2}$, $E_{\infty,N_{disc},N_{iter}}(x^j) = \|P_{N_{disc},N_{iter}}(x^j) - 1\|_{L_\infty}$ together with the correlated distribution functions which assess the independence of each component of the iterated points X_n versus the others, a window of emergence comes clearly into sight for the values $\varepsilon_i \in [10^{-15}, 10^{-7}]$, in the case $p = 4$ and $\varepsilon_{i,j} = \varepsilon_i = i\varepsilon_i$. We have also performed NIST test developed by the National Institute of Standards and Technology [23], in order to check carefully the random nature of such numbers [24].

Then there is a route from chaos to randomness using the process of chaotic undersampling.

3. GEOMETRIC UNDERSAMPLING

The previous route from chaos to randomness uses chaotic undersampling. It is possible to couple in another way p tent maps on the torus $J^p = [-1, 1]^p \subset \mathbb{R}^p$, which can directly provide random numbers, without sampling or mixing, provided p is large enough, although it is possible to combine these processes with it. After reviewing this ring coupling in high dimension, we introduce the new geometric undersampling in order to obtain randomness with small values of p (for example $p = 2$).

3.1. **Ring coupled mapping**

Consider the mapping defined on the p-dimensional torus $M_p : J^p \to J^p$

$$M_p \begin{pmatrix} x_n^1 \\ x_n^2 \\ \vdots \\ x_n^p \end{pmatrix} = \begin{pmatrix} x_{n+1}^1 \\ x_{n+1}^2 \\ \vdots \\ x_{n+1}^p \end{pmatrix} = \begin{pmatrix} 1 - 2|x_n^1| + k_1 \times x_n^2 \\ 1 - 2|x_n^2| + k_2 \times x_n^3 \\ \vdots \\ 1 - 2|x_n^p| + k_p \times x_n^1 \end{pmatrix} \tag{20}$$

with the parameters $k_i \in \{-1, 1\}$. In order to confine every variable x_n^j on J^p we do, for every iteration, the transform

$$\begin{cases} if\ (x_{n+1}^j < -1)\ add\ 2 \\ if\ (x_{n+1}^j > -1)\ substract\ 2 \end{cases} \tag{21}$$

The particularity of this coupling is that each variable x^j is coupled only with itself and x^{j+1}, as displayed on Fig. 3a. At first glance, in order to enrich the random properties of the map, it could seem interesting to add supplementary cross couplings between these variables, as shown on Fig. 3b. However in this case a cross-coupling is inappropriate because it would increase the determinism and therefore deteriorate the statistical properties which we are looking for.

To evaluate the random properties of these generators, the set of NIST tests have been used again.

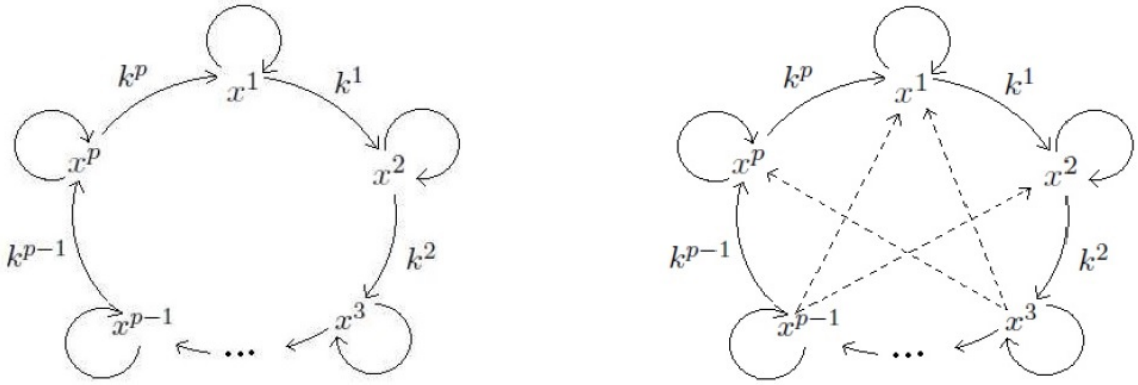


FIGURE 3. a) Ring coupling between the variables x^j. b) Cross coupling between the variables x^j.

The random properties validations of both a 4-dimensional system and a 10-dimensional one have been carried out [5]. For this purpose, the chaotic carrier output needs to be quantised and binarized (0 and 1) in order to be validated as being random using NIST tests. Therefore, different methods of binarization (converting real signals to binary ones) have been implemented and compared.

A first 1-bit binarization has been applied to the system (21) output, defined as $y_n = x_n^j$ with $j \in [\![1, p]\!]$

$$\begin{cases} if\ (y_n \geq 0)\ b = 1 \\ else\ b = 0 \end{cases} \tag{22}$$

The results showed to be highly sensitive to the type of binarization. Eventually, after testing several different methods, a 32-bit binarization has been chosen as being the most suitable solution. Because the system is confined to the p-dimensional torus J^p, 31 bits are assigned to represent the decimal part, and 1 bit to the sign. To illustrate the results, the NIST tests for the 4-dimensional system with parameters $k_i \in (-1)^{i+1}$ are shown in Fig. 4. The chosen conditions are: length of the original sequence $= 10^8$ bits, length of bit string $= 10^6$ bits, quantity of bit strings $= 100$. The output of the system has been arbitrary chosen as being: $y = x_n^4$.

Furthermore, as the results show their independence from the initial conditions, every bit string in this test is the resulting sequence of a different randomly chosen initial condition. The criterion for a successful test is that the p-value has to be superior to the significance level (0.01 for this case). For the present model, all tests were successful thus the sequences can be accepted as being random.

RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES

generator is <data/lozi_10_positif.txt>

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE	PROPORTION	STATISTICAL TEST
8	5	13	9	9	12	6	19	8	11	0.102526	96/100	Frequency
11	16	9	10	10	10	14	6	8	6	0.437274	99/100	BlockFrequency
11	5	8	11	10	5	11	11	13	15	0.419021	97/100	CumulativeSums
8	6	17	10	10	6	7	11	15	10	0.213309	97/100	CumulativeSums
5	8	17	15	6	8	6	14	10	11	0.075719	98/100	Runs
11	11	10	13	9	5	8	8	15	10	0.637119	99/100	LongestRun
6	8	17	14	10	8	9	15	7	6	0.122325	99/100	Rank
9	10	9	13	10	10	9	8	12	10	0.991468	99/100	FFT
14	15	8	10	14	10	11	9	4	5	0.191687	98/100	NonOverlappingTemplate
10	8	11	9	9	13	7	12	10	11	0.964295	99/100	OverlappingTemplate
13	16	6	8	7	10	13	10	8	9	0.455937	100/100	Universal
9	10	12	8	10	11	5	14	11	10	0.816537	97/100	ApproximateEntropy
6	5	6	5	9	11	5	6	8	5	0.637119	65/66	RandomExcursions
3	5	6	7	10	10	9	6	4	6	0.407091	65/66	RandomExcursionsVariant
3	8	8	12	12	9	13	8	13	14	0.319084	100/100	Serial
4	3	12	18	12	8	8	14	9	12	0.028817	100/100	LinearComplexity

FIGURE 4. Example of NIST Test for $k^i = (-1)^{i+1},\ i = 1, 4$, each sequence of components satisfies the NIST test for randomness.

3.2. **Low dimensional model**

Although the system (20) is a good CPRNG when $p \geq 4$, in lower dimension 2 and 3, the chaotic numbers are not equidistributed on the torus (see Fig. 5).

In order to improve the ring coupling mechanism in low dimension, we introduce now a very new type of undersampling based on geometric nature of the invariant measure. We present this very new mechanism which allows the emergence of randomness from chaos, in the simplest case, the 2-dimensional ring mapping M_2 on the square J^2 , with $k^1 = k^2 = 1$. Let M_2 be defined by

$$\begin{cases} x_{n+1}^1 = 1 - 2|x_n^1| + x_n^2 \\ x_{n+1}^2 = 1 - 2|x_n^2| + x_n^1 \end{cases} \tag{23}$$

$$with \quad \begin{cases} if\ (x_{n+1}^j < -1)\ add\ 2 \\ if\ (x_{n+1}^j > -1)\ substract\ 2 \end{cases} \tag{24}$$

3.2.1. *Critical lines.*

Figure 5 shows the distribution of the iterates of system (23) (the transient of the first $= 10^6$ iterations has been cut off). It can be observed that the attractor contains regions where the point density is lower, and two lozenge-like holes. It is possible to define critical lines which form a partition of the square J^2. The critical lines CL [25] are singularities of dimension 1 and represent an important tool for the analysis of noninvertible maps. The holes on Fig. 5 are completely delimited by segments of the critical lines $CL_1^{A1}, CL_1^{B4}, CL_1^{C2}, CL_1^{D4}$ and $CL_1^{A2}, CL_1^{B3}, CL_1^{C1}, CL_1^{D3}$, defined below.

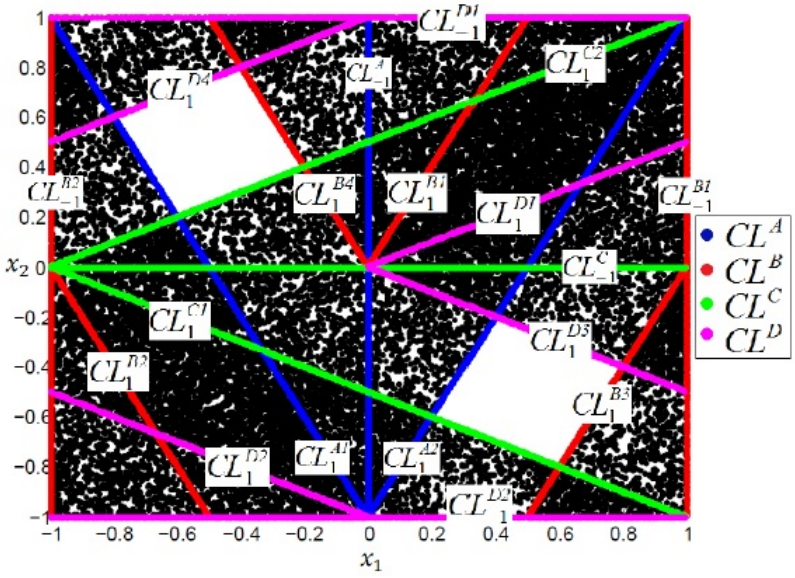


FIGURE 5. Critical lines of the map M_2 on the torus J^2 (a square) [26].

The critical lines separate regions of the phase space with different number of preimages (backward iterates). In the case of piecewise linear maps, they are the first iterates of the lines of discontinuity CL_{-1} of the system.

For the two dimensional system (23), there are four groups of critical lines CL with preimages CL_{-1} given by

Critical lines A: $CL_{-1}^{A} : x^1 = 0$

$$and \quad \begin{cases} CL_1^{A1} : x^2 = -2x^1 - 1 \quad if\ x^2 > 0 \\ CL_1^{A2} : x^2 = 2x^1 - 1 \quad if\ x^2 < 0 \end{cases} \tag{25}$$

Critical lines B: $CL_{-1}^{B} : x^1 = -1$

$$and \quad \begin{cases} CL_1^{B1} : x^2 = 2x^1 \quad if\ x^2 < 0,\ x^1 \in [0, 0.5] \\ CL_1^{B2} : x^2 = -2x^1 - 2 \quad if\ x^2 > 0,\ x^1 \in [-1, -0.5] \\ CL_1^{B3} : x^2 = 2x^1 - 2 \quad if\ x^2 < 0,\ x^1 \in [0.5, 1] \\ CL_1^{B4} : x^2 = -2x^1 \quad if\ x^2 > 0,\ x^1 \in [-0.5, 0] \end{cases} \tag{26}$$

Critical lines C: $CL_{-1}^{C} : x^2 = 0$

$$and \quad \begin{cases} CL_1^{C1} : x^2 = -\frac{1}{2}(x^1 + 1) \quad if\ x^1 > 0 \\ CL_1^{C2} : x^2 = \frac{1}{2}(x^1 + 1) \quad if\ x^2 < 0 \end{cases} \tag{27}$$

Critical lines D: $CL_{-1}^{D} : x^2 = -1$

$$and \quad \begin{cases} CL_1^{D1} : x^2 = \frac{x^1}{2} \quad if\ x^1 < 0,\ x^2 \in [0, 0.5] \\ CL_1^{D2} : x^2 = -\frac{x^1}{2} - 1 \quad if\ x^1 > 0,\ x^2 \in [-1, -0.5] \\ CL_1^{D3} : x^2 = -\frac{x^1}{2} \quad if\ x^1 > 0,\ x^2 \in [-0.5, 0] \\ CL_1^{D4} : x^2 = \frac{x^1}{2} + 1 \quad if\ x^1 < 0,\ x^2 \in [0.5, 1] \end{cases} \tag{28}$$

3.2.2. *Markov partition.*

Our aim is first to use the partition defined by these critical lines in order to do a cell-to-cell analysis and, by the means of a Markov process, to compute explicitly the invariant measure of iterates associated to system (23). Figure 6 displays the 32 sub-regions of the square J^2, labelled from a to p and a' to p'. For clarity of the presentation, we have labelled from (I) to (IV), the four quadrants of J^2.

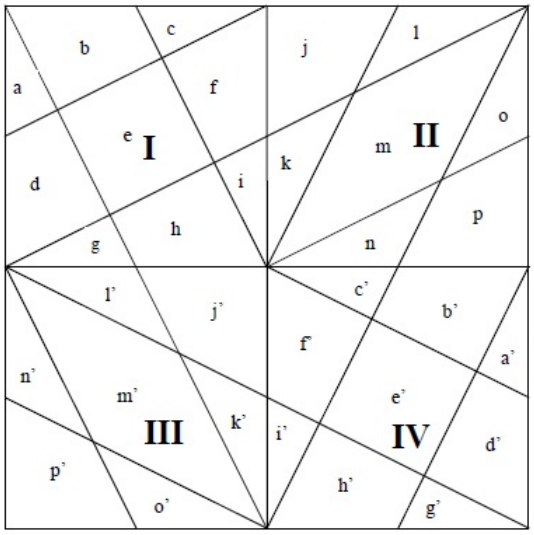
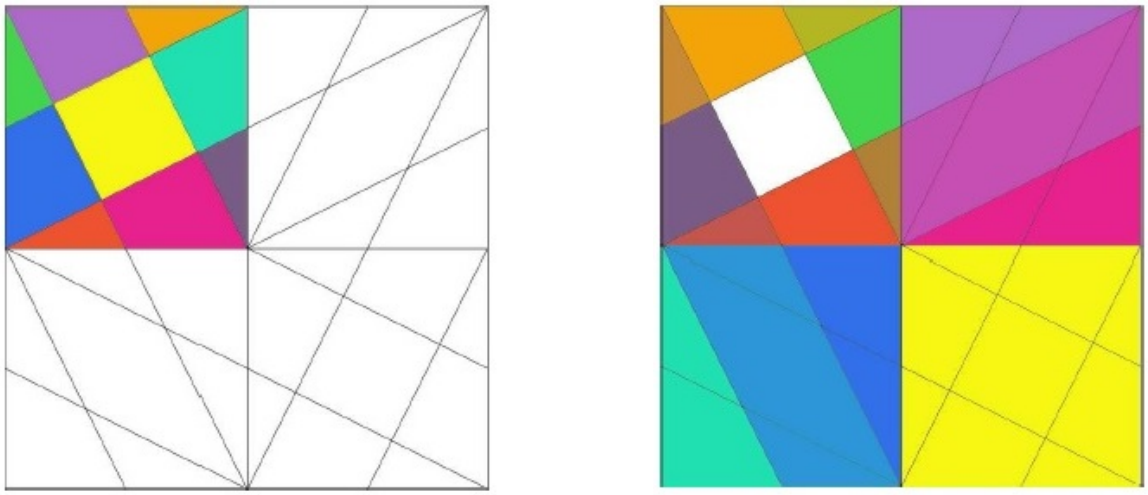


FIGURE 6. The 32 sub-regions for a partition of the square J^2.

Straightforward computation shows that the images of each region, by the mapping M_2, is either one, two or three regions of the same partition of the square J^2. Figures 7a and 7b display the images of the regions embedded in the first quadrant (I). Figures 8a and 8b display the images of the regions embedded in the second quadrant (II). The colour is the same for every region and its corresponding image, except when two regions are mapped on the same region, in this case there is a mix of colours on the common part of the image.

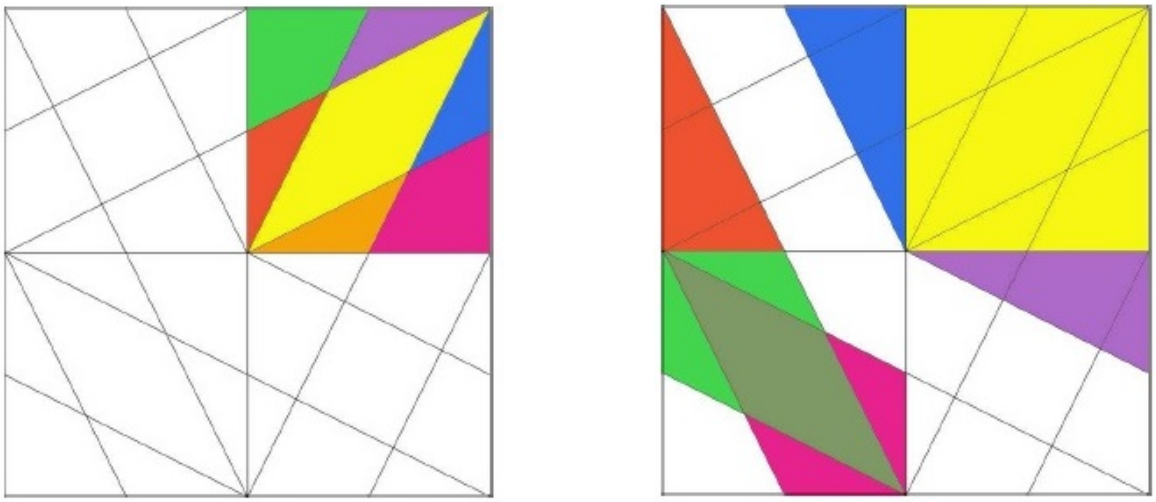
The overall correspondence between regions of the partition and their image is given by the Markov matrix M_a which is displayed on Fig. 9. The computation of the coefficients of this matrix, which are rational numbers, is based on the ratios of surfaces of bounded regions.

In order to display the 32×32 matrix M_a on one page, we have labelled the coefficients using letters **which are not related** to the names of the regions.



a) b)

FIGURE 7. a) The nine regions a to i of quadrant (I) b) The images M_2 (I) of the nine regions of quadrant (I).



a) b)

FIGURE 8. a) The seven regions j to p of quadrant (II) b) The images M_2 (II) of the seven regions of quadrant (II).

$$\begin{cases} o = \frac{1}{12};\ p = \frac{1}{20};\ q = \frac{3}{20};\ q = \frac{4}{20}; \\ s = \frac{1}{9};\ t = \frac{2}{9};\ u = \frac{4}{9};\ v = \frac{1}{6}; \\ w = \frac{1}{3};\ x = \frac{1}{5};\ y = \frac{3}{5};\ z = \frac{2}{3}. \end{cases} \tag{29}$$

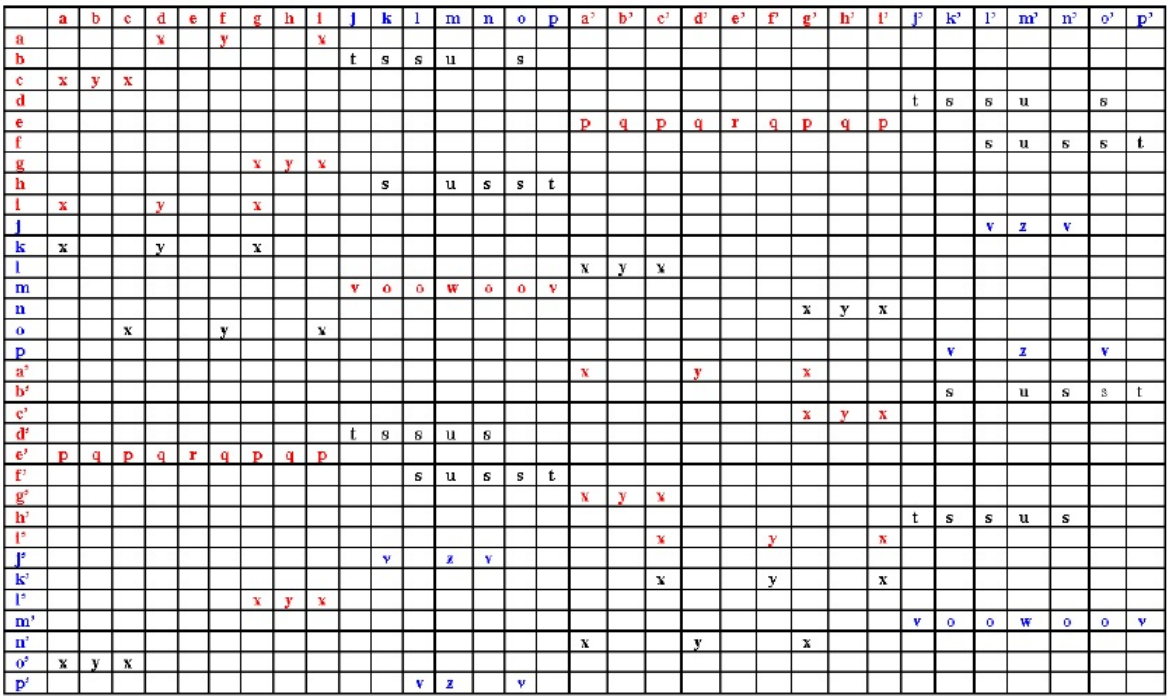


FIGURE 9. Markov Matrix M_a.

3.2.3. *Exact computation of invariant measure associated to M_2*

With the help of Markov matrix M_a, it is straightforward to compute explicitly the invariant measure associated to M_2. For every region on Fig. 6, we define a quantity of initial points called $Q^i, i = 1, 32$ uniformly scattered on it, and we compute its surface S_i. We normalise both quantities to $\sum_i Q^i = |Q| = 4$, and $\sum_i S^i = |S| = 4$. Hence it is possible to define the density of iterates on each region.

$$d^i = \frac{Q^i}{S_i} \tag{30}$$

Let $Q = \begin{pmatrix} Q^1 \\ \vdots \\ Q^{32} \end{pmatrix}$ and $D = \begin{pmatrix} d^1 \\ \vdots \\ d^{32} \end{pmatrix}$ the vectors of quantities and densities obtained applying (30) to every region. Then starting from an arbitrary initial repartition of points on J^2, say $Q = \begin{pmatrix} Q_0^1 \\ \vdots \\ Q_0^{32} \end{pmatrix}$, and applying repeatedly the equation

$$Q_{m+1} = M_a^t Q_m \tag{31}$$

The sequence of vectors $\{Q_m\}_{m\in\mathbb{N}}$ converges to a limit vector $\overline{Q}$ which satisfies

$$\overline{Q} = M_a^t \overline{Q} \tag{32}$$

and gives the invariant measure, the density of which is the vector $\overline{D}$, using (30).

Numerical results: Starting from the initial guess $Q_0 = \begin{pmatrix} Q_0^1 \\ \vdots \\ Q_0^{32} \end{pmatrix} = \begin{pmatrix} 1/8 \\ \vdots \\ 1/8 \end{pmatrix}$, $\overline{Q}$ is obtained rapidly, as

$$Q_{500} = \begin{pmatrix} Q_{500}^1 \\ Q_{500}^2 \\ Q_{500}^3 \\ Q_{500}^4 \\ \vdots \\ Q_{500}^{29} \\ Q_{500}^{30} \\ Q_{500}^{31} \\ Q_{500}^{32} \end{pmatrix} = \begin{pmatrix} 1/14 \\ 3/28 \\ 1/14 \\ 3/28 \\ \vdots \\ 4/7 \\ 3/28 \\ 3/28 \\ 1/7 \end{pmatrix} = \overline{Q}, \ \textit{which gives using } (30), D_{500} = \begin{pmatrix} d_{500}^1 \\ d_{500}^2 \\ d_{500}^3 \\ d_{500}^4 \\ \vdots \\ d_{500}^{29} \\ d_{500}^{30} \\ d_{500}^{31} \\ d_{500}^{32} \end{pmatrix} = \begin{pmatrix} 10/7 \\ 5/7 \\ 10/7 \\ 5/7 \\ \vdots \\ 12/7 \\ 9/7 \\ 9/7 \\ 6/7 \end{pmatrix} = \overline{D}$$

Remark 3.1. Computing directly this density, iterating (23) up to 10^{11} iterates leads to the same result.

3.3. **Geometric undersampling**

The exact computation of the density $\overline{D}$ of the invariant measure shows that this density is constant on each region. The geometric undersampling process consists in magnifying a square G included in one region (as for example the square $G = [0.36, 0.64] \times [0.36, 0.64]$ included in region m on Fig. 10), up to the size of the square J^2.

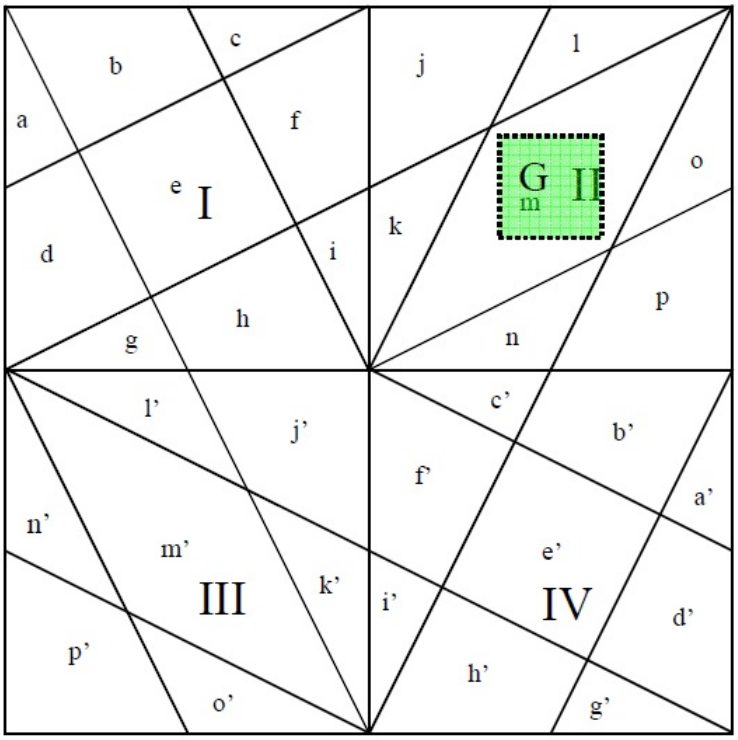


FIGURE 10. The square $G = [0.36, 0.64] \times [0.36, 0.64]$ in which iterates of (23) are geometrically undersampled.

3.3.1. *Algorithm of geometric undersampling*

Let $G = [x_l^1, x_r^1] \times [x_l^2, x_r^2]$ the square in which we will undersample the iterate of (23) and, $x_{mean}^1 = \frac{x_l^1 + x_r^1}{2}$, $x_{mean}^2 = \frac{x_l^2 + x_r^2}{2}$. In algorithmic form, the pseudo-code, to geometric undersample N iterates of (23) is

$X_0 = (x_0^1, x_0^2)$
$n = 0;$
$\quad do\ \{\ while\ n < N\ compute\ (x_n^1, x_n^2);\ if\ (x_n^1, x_n^2) \in G\ then$
$\quad \overline{(x_q^1} = 2[\frac{x_n^1 - x_{mean}^1}{x_r^1 - x_l^1}],\ \overline{(x_q^2} = 2[\frac{x_n^2 - x_{mean}^2}{x_r^2 - x_l^2}];\ q = q + 1;\ n{+}{+}\ \}$

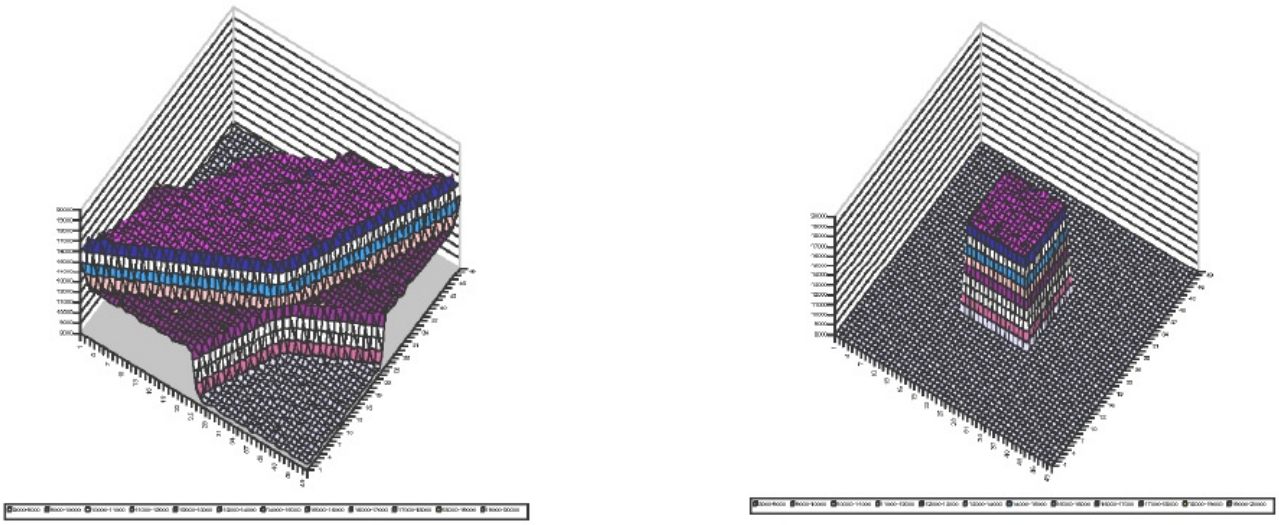
Remark 3.2. In this case, the undersampling process provides two streams of pseudo-random numbers.

Remark 3.3. In this case, $NSampl_{iter}$ the number of geometrically undersampled iterates is not known *a priori*, however, considering that the selecting process is linked to the uniform distribution of the iterates of the tent map on J^2, one has $NSampl_{iter} \approx \frac{(x_r^1 - x_l^1)^2}{4} \times d^m$, where d^m is the density of the measure in region m.

3.3.2. *Numerical tests*

We have applied this process in the case of the square G of Fig. 10, with $N = 10^{12}$ which gives $NSampl_{iter} \approx 3.35 \times 10^{10}$. Figure 11a displays the densities of the seven regions j, k, l, m, n, o, p of quadrant (II) which are equal to gives

$$\begin{cases} \overline{d^j} = \dfrac{6}{7};\ \overline{d^k} = \dfrac{9}{7};\ \overline{d^l} = \dfrac{9}{7};\ \overline{d^m} = \dfrac{12}{7}; \\ \overline{d^n} = \dfrac{9}{7};\ \overline{d^o} = \dfrac{9}{7};\ \overline{d^p} = \dfrac{6}{7}. \end{cases}$$



a) b)

FIGURE 11. a) Densities of the seven regions j, k, l, m, n, o, p of quadrant (II) b) Uniform density of iterates in the square $G = [0.36, 0.64] \times [0.36, 0.64]$ of quadrant (II).

Figure 11b shows the uniform density of iterates in the square $G = [0.36, 0.64] \times [0.36, 0.64]$ of quadrant (II). On Fig. 12 the square is magnified up to the size of the square J^2. The vertical scale is fitted near the invariant Lebesgue measure.

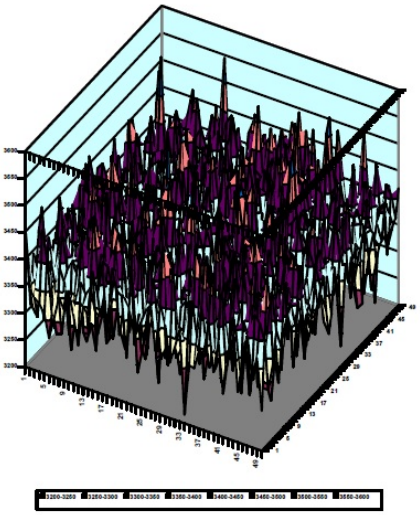


FIGURE 12. Uniform density of iterates of the square $G = [0.36, 0.64] \times [0.36, 0.64]$ magnified to the square J^2.

We have also used NIST test to confirm the random property of the geometrical undersampling process. They are all successful (Fig. 13).

RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE	PROPORTION	STATISTICAL TEST
12	9	7	9	12	11	8	11	13	8	0.924076	99/100	Frequency
1	4	3	4	7	5	9	16	16	35	0.000000 *	100/100	BlockFrequency
9	9	10	12	11	8	9	10	10	12	0.996335	99/100	CumulativeSums
10	9	12	12	9	7	10	10	9	12	0.983453	99/100	CumulativeSums
11	12	11	8	12	7	12	6	10	11	0.883171	99/100	Runs
9	9	13	8	9	8	17	8	10	9	0.595549	100/100	LongestRun
6	11	11	11	9	8	8	14	9	13	0.798139	100/100	Rank
15	10	7	8	8	8	15	16	7	6	0.153763	97/100	FFT
12	9	10	13	9	11	7	15	4	10	0.474986	98/100	NonOverlappingTemplate
12	6	10	6	13	6	8	8	17	14	0.145326	99/100	OverlappingTemplate
18	12	13	11	9	10	5	8	9	5	0.145326	99/100	Universal
11	8	12	11	11	14	8	10	7	8	0.883171	99/100	ApproximateEntropy
3	5	6	9	4	3	7	5	6	11	0.145326	59/59	RandomExcursions
7	6	6	2	6	7	6	7	4	8	0.637119	59/59	RandomExcursions
2	6	4	5	5	6	10	6	7	8	0.334538	59/59	RandomExcursionsVariant
8	15	13	12	9	12	13	5	9	4	0.224821	98/100	Serial
9	9	6	13	13	7	12	9	10	12	0.798139	99/100	LinearComplexity

The minimum pass rate for each statistical test with the exception of the random excursion (variant) test is approximately = 96 for a sample size = 100 binary sequences.

FIGURE 13. Geometrical undersampling: each sequence of components satisfies the NIST test for randomness.

4. CONCLUSION

We have proposed a new mechanism of undersampling of chaotic number obtained by the ring coupling mechanism of one-dimensional maps. In the case of 2 coupled maps this mechanism allows the building of a CPRNG which passes all NIST Test.

This new geometric undersampling is very effective for generating 2 parallel streams of pseudo-random numbers, as we have shown, computing carefully their properties, up to sequences of 10^{12} consecutives iterates

of (23) which provides more than 3.35×10^{10} random numbers in very short time. In a forthcoming paper we will test the 3-dimensional case.

References

[1] R. Lozi, "Complexity leads to randomness in chaotic systems," *Mathematics in Science and Technology: Mathematical Methods, Models and Algorithms in Science and Technology. (A.H. Siddiqi, R.C. Singh, P. Manchanda Editors), World Scientific Publisher, Singapore*, pp. 93–125, 2011.

[2] R. Lozi, "Emergence of randomness from chaos," *International Journal of Bifurcation and Chaos*, vol. 22, no. 02, pp. 1250021–1/15, 2012.

[3] R. Lozi, "Giga-periodic orbits for weakly coupled tent and logistic discretized maps," *Modern Mathematical Models, Methods and Algorithms for Real World Systems,* A. H. Siddiqi, I. S. Duff and O. Christensen (Editors)*, Anamaya Publishers, New Delhi, India*, pp. 80–124, 2006.

[4] R. Lozi, "Chaotic pseudo random number generators via ultra weak coupling of chaotic maps and double threshold sampling sequences," *In proceedings of: ICCSA 2009 The 3rd International Conference on Complex Systems and Applications, University of Le Havre, France, June 29-July 02 (2009)*, pp. 20–24, 2009.

[5] A. E. Rojas, I. Taralova, and R. Lozi, "New alternate ring-coupled map for multi-random number generation," *Journal of Nonlinear Systems and Applications*, vol. 4, no. 3-4, pp. 65–69, 2013.

[6] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of the atmospheric sciences*, vol. 20, no. 2, pp. 130–141, 1963.

[7] M. Hénon, "A two-dimensional mapping with a strange attractor," *Communications in Mathematical Physics*, vol. 50, no. 1, pp. 69–77, 1976.

[8] J. C. Sprott *Chaos and time-series analysis, Oxford University Press, Oxford, UK*, 2003.

[9] O. E. Lanford III, "Some informal remarks on the orbit structure of discrete approximations to chaotic maps," *Experimental Mathematics*, vol. 7, no. 4, pp. 317–324, 1998.

[10] P. Góra, A. Boyarsky, M. S. Islam, and W. Bahsoun, "Absolutely continuous invariant measures that cannot be observed experimentally," *SIAM Journal on Applied Dynamical Systems*, vol. 5, no. 1, pp. 84–90, 2006.

[11] K. J. Palmer, "Shadowing in dynamical systems: theory and applications," *Kluwer Academic Publications*, vol. 501, pp. 65–69, 2000.

[12] S. Y. Pilyugin, "Shadowing in dynamical systems," *Lecture Notes in Math.*, vol. 1706, 1999.

[13] R. Lozi, "Can we trust in numerical computations of chaotic solutions of dynamical systems?," *In Topology and Dynamics of Chaos,* Ch. Letellier, R. Gilmore (Eds.)*, World Scientific Series in Nonlinear Science Series A , Chapt. 3*, 2013.

[14] M. S. Baptista, "Cryptography with chaos," *Physics Letters A*, vol. 240, no. 1, pp. 50–54, 1998.

[15] M. R. K. Ariffin and M. S. M. Noorani, "Modified Baptista type chaotic cryptosystem via matrix secret key," *Physics Letters A*, vol. 372, no. 33, pp. 5427–5430, 2008.

[16] M. Pluhacek, V. Budikova, R. Senkerik, Z. Oplatkova, and I. Zelinka, "Extended initial study on the performance of enhanced PSO algorithm with Lozi chaotic map," *Advances in Intelligent Systems and Computing, Nostradamus: Modern Methods of Prediction, Modeling and Analysis of Nonlinear Systems*, vol. 192, pp. 167–177, 2013.

[17] T. W. Tang, A. G. Allison, and D. Abbott, "Parrondo's games with chaotic switching," *Proc. SPIE Noise in Complex Systems and Stochastic Dynamics II*, vol. 5471, pp. 520–530, 2004.

[18] E. Araujo and L. d. S. Coelho, "Particle swarm approaches using Lozi map chaotic sequences to fuzzy modelling of an experimental thermal-vacuum system," *Applied Soft Computing*, vol. 8, no. 4, pp. 1354–1364, 2008.

[19] S. Hénaff, I. Taralova, and R. Lozi, "Exact and asymptotic synchronization of a new weakly coupled maps system," *Journal of Nonlinear Systems and Applications*, vol. 1, no. 3-4, pp. 87–95, 2010.

[20] R. Lozi, "New enhanced chaotic number generators," *Indian Journal of Industrial and Applied Mathematics*, vol. 1, no. 1, pp. 1–23, 2008.

[21] J. Viega, "Practical random number generation in software," *in Proceedings of 19th Annual Annual Computer Security Applications Conference*, pp. 129–140, 2003.

[22] J. Viega and M. Messier *Secure Programming Cookbook for* C *and* C++*: Recipes for Cryptography, Authentication, Input Validation & More (O'Reilly, Sebastopol, CA)*, 2003.

[23] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," *NIST*, 2001.

[24] R. Lozi, I. Taralova, and S. Hénaff, "Dynamical analysis of a new statistically highly performant deterministic function for chaotic signals generation," *In Proceeding of Physcon 2009, Catania, Italy, 1-4 september, IPACS open Access Electronic Library*, 2009.

[25] C. Mira, L. Gardini, A. Barugola, and J.-C. Cathala, "Chaotic dynamics in two-dimensional noninvertible maps," *World Scientific Series on Nonlinear Science, Series A*, vol. 20, 1996.

[26] I. Taralova, A. Espinel, and R. Lozi, "Dynamical and statistical analysis of a new lozi function for random numbers generation," *in Proceeding of Physcon 2011, León, Spain, 5-8 september, IPACS open Access Electronic Library*, 2011.