A universal algorithm to generate pseudo-random numbers based on uniform mapping as homeomorphism*

Wang Fu-Lai(王福来)†

Department of Mathematics and Statistics, Zhejiang University of Finance and Economics, Hangzhou 310012, *China*

(Received 5 January 2010; revised manuscript received 10 February 2010)

A specific uniform map is constructed as a homeomorphism mapping chaotic time series into [0,1] to obtain sequences of standard uniform distribution. With the uniform map, a chaotic orbit and a sequence orbit obtained are topologically equivalent to each other so the map can preserve the most dynamic properties of chaotic systems such as permutation entropy. Based on the uniform map, a universal algorithm to generate pseudo random numbers is proposed and the pseudo random series is tested to follow the standard 0-1 random distribution both theoretically and experimentally. The algorithm is not complex, which does not impose high requirement on computer hard ware and thus computation speed is fast. The method not only extends the parameter spaces but also avoids the drawback of small function space caused by constraints on chaotic maps used to generate pseudo random numbers. The algorithm can be applied to any chaotic system and can produce pseudo random sequence of high quality, thus can be a good universal pseudo random number generator.

Keywords: pseudo random numbers, uniform map, chaos

PACC: 0545, 0250

1. Introduction

A pseudo random number generator(PRNG) is always used as a tool to obtain symbolic sequences $\{s_n\}$ on a finite integer set by digitizing chaotic time series $\{x_n\}$.[1−9] It is a key problem to produce pseudo random numbers of high quality in many fields of science and technology. A lot of PRNGs have been devised.[1−7] However there are some major defects. Summarizing the drawbacks, the difficulties lies in the following aspects.

(i) Complex theories are involved when PRNGs are applied, which is nonbeneficial to computer hardware implementation[1,8,9] and the uniformity of the obtained random sequences cannot be guaranteed. For example, TD-ERCS chaotic system was constructed to generate the PRNG in Ref. [8], but calculation speed was decreased and extreme difficulty came into being when FPGA was used because arcsine and arccosine functions were applied to guarantee a uniform distribution.[1,8] A universal algorithm was devised to obtain uniform random binary sequences and overcome the difficulty in implementing computer hardwares.[1] Unfortunately, only relatively uniform time series $\{x_n\}$ was obtained because bit-operations were applied too many times and chaotic properties were damaged. It is hard to obtain uniform pseudo random sequences of high quality with the time series $\{x_n\}$.

(ii) PRNGs are undesirable because two most important properties of chaotic systems were not utilized sufficiently.[1−3,8,9] The two most important properties are sensitively dependent on initial conditions and the ergodicity (in other words, invariant density) of chaos orbits.

(iii) Function spaces are very limited.[10−14] Functions have to be skillfully constructed for many algorithms and some algorithms can be applied to only one function, which causes insecurity of stream cipher. For example, Wang and Gan[11] further developed Kawamoto and Horiuchi's theory[12] and constructed a z-Logistic map as a conjugate map to obtain a number series which preserves chaotic properties. By mapping the number orbit onto [0,1], Wang and Gan obtained a pseudo random series almost totally equivalent to the real orbit by which periods could be predicted. It is a very good algorithm to avoid weak keys caused by rounding error of computers for a finite accuracy. However there exist some weak points as follows: (A) prime number theory and inverse trigonometric function are involved, which results in low calculation speed and is nonbeneficial to computer hardware implementation, (B) parameter space is too limited because very large prime number is needed.

*Projected supported by the National Natural Science Foundation of China (Grant No. 10871168).

†Corresponding author. E-mail: flyerwon@sina.com

© 2010 Chinese Physical Society and IOP Publishing Ltd

By prime number theory, numbers of prime numbers with L bit-precision is $(L-1)2^L \ln 2/L(L+1)$. Large prime numbers are hard to test because the density of large numbers becomes sparse and thus it is very time consuming even though computers are used and (C) maps for the algorithm are restricted with strict conditions and the algorithm cannot be extended to high dimensional systems, which chaotic systems of high dimensions have been extensively applied to stream cipher.

In this paper a uniform map is created as a homeomorphism and thus an algorithm is developed to generate pseudo random sequences. It is proved both theoretically and experimentally that the algorithm can be applied to any chaotic maps. Some typical examples are given to analyse properties of the PRNGs such as proportionality, run length and correlation, and the results are very satisfactory.

2. Algorithm and theorems

2.1. Propose a uniform map as a homeomorphism

Suppose that x_n is a time series obtained from a chaotic system of any dimensions and given as

$$x_n = \{x_0, x_1, x_2, \ldots\}, \tag{1}$$

where x_0 is the initial value or called a seed in cryptography. According to chaotic dynamics, $\{x_n\}$ follows a probability distribution and x_n is sensitively dependent on x_0.

Definition 1 (the uniform map) Uniform map $g(\cdot)$ in this paper is defined as

$$y_n = g(x_n) = \frac{K(x_n)}{N}, \quad n = 1, 2, \ldots, N, \tag{2}$$

where N is the length of the sample series of $\{x_n\}$, sample series of $\{x_1, x_2, \ldots, x_n, \ldots, x_N\}$ is denoted by $\{x_n\}_1^N$, and $K(x_n)$ is the serial number of x_n in $\{x_n\}_1^N$ when arranged in ascending order.

Obviously the series $\{y_n\}_1^N$ follows uniform distribution and $g(\cdot)$ is a homeomorphism. Sequences of $\{x_n\}_1^N$ and $\{y_n\}_1^N$ are topologically equivalent to each other and thus $\{x_n\}_1^N$ and $\{y_n\}_1^N$ preserve many topological invariances as well as some dynamic properties such as permutation entropy.[15,16]

In order to transfer chaotic time series $\{x_n\}_1^N$ into pseudo random sequences $\{s_n\}_1^N$, we define transfer function $T(x_n)$ as follows.

Definition 2 (transfer function) The transfer function is

$$s(n) = T(x_n) = \begin{cases} 0, & x_n \in (\bigcup_{d=1}^{m}) B_{2d-1}^{2m}, \\ 1, & x_n \in (\bigcup_{d=1}^{m}) B_{2d}^{2m}, \end{cases} \tag{3}$$

where $2m$ is a positive integer, $B_1^{2m}, B_2^{2m}, \ldots, B_{2m}^{2m}$ are the $2m$ consecutive equal subintervals of $[0, 1]$.

Transfer function $T(x_n)$ aims to transfer $B_1^{2m}, B_2^{2m}, \ldots$, and B_{2m}^{2m} into 0 and 1 alternatively. $2m$ can be taken to be larger values with N becoming larger, but different values of $2m$ do not have much influence on the results. It will be proved in Subsection 2.3 that the set of $\{s_n\}_{n=1}^N$ has good properties such as balanced proportion of 0's to 1's and δ-like autocorrelation.

2.2. Relevant concepts[1−5]

(i) Uniform 0-1 random sequences

A 0-1 random sequence $\{s_n\}_1^N$ is called a uniform one if $P\{s_n = 0\} = P\{s_n = 1\} = 0.5$.

(ii) Runs and distribution of runs

The length of the same code elements appeared consecutively on $\{s_n\}_1^N$ is called run length. A run with length i is denoted by i-run. The i-runs of '0' and '1' code elements are denoted by $R_i^0, R_i^1, i = 1, 2, \ldots$, relatively.

Runs from 1-run to 4 run together are called short runs.

Theoretically, for a uniform 0-1 random sequence $\{s_n\}_1^N$, s_n takes 0 or 1 with probability 0.5, short runs take up a large proportion, probability of i-run is 2^{-i} and probabilities of R_i^0 and R_i^1 are the same. Probabilities or proportions of R_i^0 and R_i^1 on a sequence are denoted by $P(R_i^0)$, $P(R_i^1)$, $i = 1, 2, \ldots$ relatively.

(iii) Test of runs distribution

The test of run distribution is to test how well the proportions of runs on a 0-1 pseudo random sequence are approximate to the probabilities of runs on a 0-1 random sequence.

(iv) Autocorrelation and cross-correlation

Suppose that $\{x_n\}_1^N$ and $\{u_n\}_1^N$ are two 0-1 pseudo random sequences. Autocorrelation function $R_{xx}(T)$ and $R_{xy}(T)$ are defined as follows:

$$R_{xx}(T) = \frac{1}{N} \sum_{i=0}^{N-1} x_i x_{i+T}, \tag{4}$$

$$R_{xu}(T) = \frac{1}{N} \sum_{i=0}^{N-1} x_i u_{i+T}, \tag{5}$$

where T is the correlation lag.

In this paper, in order to compare with the results of current literature. N is taken to be 2000 and 20000 and the value of T is taken in a range from -500 to 500 when autocorrelation and cross-correlation are computed, respectively.

2.3. Theorems

Theorem 1 Suppose that X is a uniform continuous random variable in $[0,1]$, $\{x_n\}_1^N$ is a sample series and $\{s_n\}_1^N$ is the random sequences transformed by Eq. (3), then $\{\lim_{N\to\infty} s_n\}_1^N$ will follow a uniform 0-1 random distribution.

Proof By classical probability theory, it is obvious that $P\{s_n=0\}=P\{s_n=1\}=0.5$ and $P\{s_n=\xi, s_{n+1}=\eta\}=P\{s_n=\xi\}P\{s_{n+1}=\eta\}$, whatever value $2m$ takes, where $n=1,2,\ldots$, $\xi=0,1, \eta=0,1$. Thus Theorem 1 holds.

Theorem 2 Suppose that $\{x_n\}$ is a chaotic time series, $x_n\in[a,b]$, $y=p(x)$ is an invariant density function, $\{x_{n+k}\}_1^N$ is a sample sequence obtained from $\{x_n\}$, and $\{s_{n+k}\}_1^N$ are obtained by the uniform map (Eq. (2)) and the transfer function (Eq. (3)), then $\{\lim_{k\to\infty} s_{n+k}\}_1^N$ will follow a uniform 0-1 random distribution.

Proof Sensitive dependence on initial values and the ergodicity (in other words, invariant density) of chaos orbits are the two most important characteristics of chaotic systems. We will prove Theorem 2 in 4 steps.

(A) $\{y_n\}$ follows a uniform distribution in $[0,1]$ though $\{x_n\}$ is not periodic, where $\{y_n\}$ is obtained from $\{x_n\}$ by Eq. (2). For any N, the numbers of 0 and 1 in $\{s_n\}_1^N$ equal or differ by 1.

(B) $\{x_n\}$ follows a deterministic distribution. For a certain $2m$ of the number of equal subintervals of $[0,1]$ and some N, suppose $x_i\in B_{k_1}^{2m}=[c_1,c_2)$, then by step (A) $x_i\in B_{k_1}^{2m}=[c_1,c_2)$ will still hold with N changing, that is s_i will not change with sequence length N.

(C) By chaotic dynamics and symbolic dynamics,[17] x_i and x_{i+k} will be asymptotically independent of each other with k increasing and $\lim_{k\to\infty} x_{n+k}$ follows the distribution of $\{x_n\}_1^\infty$, where the density function is $y=p(x)$. Consider step (B), similarly, y_i and y_{i+k}, s_i and s_{i+k} will also be asymptotically independent of each other with k increasing and $\lim_{k\to\infty} P\{s_{i+k}=1\}=\lim_{k\to\infty} P\{s_{i+k}=0\}=1/2$.

Suppose that $y_i, y_j\in B_{k_1}^{2m}, s_{k_1}=1$ (or 0), thus $s_i=s_j=1$. We have $\lim_{k\to\infty} P\{s_{i+k}=\xi, s_{j+k}=\eta\}=\lim_{k\to\infty}(P\{s_{i+k}=\xi\}P\{s_{j+k}=\eta\})=1/4$, $\xi=0,1$, and $\eta=0,1$.

(D) Consider $\{s_{n+k}\}$. By steps (A), (B) and (C), then $\{\lim_{k\to\infty} s_{n+k}\}_1^N$ will follow a uniform 0-1 random distribution, thereby completing the proof.

Theorems 1 and 2 hold for any chaotic map or system, which provides a universal algorithm for the PRNG.

3. Constructing a universal algorithm of PRNG and testing empirically

3.1. Constructing a universal algorithm of PRNG

By Section 2 we construct a universal algorithm of PRNG in four steps as follows:

Step 1 For any chaotic map or system, we obtain chaotic time series $\{x_n\}_0^{N'}$ and $\{x_{n+k}\}_1^N$, where $N'\geq kN$ and x_0 is the seed;

Step 2 Obtain $\{Kx_n\}_0^{N'}$ by Eq. (2). Let $B(n)=K(n+1)-K(n)$, and then $\{B_n\}_1^{N'}$ will be obtained. If the number of 0 in $\{B_n\}_1^{N'}$ is no less than 1, x_0 means weak key in the computer accuracy. A weak key will be given up. It takes about 5 s of computer time to test weak keys for a series with length 80000;

Step 3 By Eq. (2) we have $\{K(x_{n+k})\}_1^N$ and $\{y_{n+k}\}_1^N$;

Step 4 By Eq. (3) uniform 0-1 pseudo random sequences $\{s_{n+k}\}_1^N$ will be obtained.

In fact, the results will be very satisfactory when $k=1$ because of coarse graining of $2m$ equal subintervals. k can be one of parameters in signal codes.

3.2. Typical chaos maps

To show the universality of the algorithm, we select some typical chaotic systems including weak and strong chaos ones, discrete and continuous chaos maps and chaotic systems. The maps or the systems range from one dimension to three dimensions. They are as follows:

(i) Map1: Logistic map

$$x_{n+1}=\mu x_n(1-x_n), \mu\in(0,4]; \tag{6}$$

(ii) Map2: z-logistic map[11,12]

$$l_n=zl_{n-1}, \tag{7}$$

where z is a generator of Z_m^* (Z_m^* is the set with numbers that are relatively coprime to m, with m being a large prime number), l_0 is any integer satisfying $1 \leq l_0 \leq m-1$.

(iii) Map3: Henon map

$$\begin{aligned} x_{n+1} &= 1 - ax_n^2 + y_n, \\ y_{n+1} &= bx_n; \end{aligned} \tag{8}$$

(iv) Equations: Lorenz differential equations

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= -\beta x + yz, \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= -\sigma y + \sigma z, \\ \frac{\mathrm{d}z}{\mathrm{d}t} &= -xy + \rho y - z. \end{aligned} \tag{9}$$

where $\sigma = 10, \rho = 28, \beta = 8/3$, initial values are $x_0 = y_0 = 0, z(0) = \varepsilon$, ε is a small constant and assumed to be $\varepsilon = 10^{-10}$.

3.3. Proportions of runs

In this subsection we take $2m = 200$, $N = 20000$ and $k = 1$.

Denote $(P(R_i^0), P(R_i^1), P(R_i^0) + P(R_i^1)/2)$ by $P_i(\cdot,\cdot,\cdot)$, $\sum_{i=1}^{4}(P(R_i^0) + P(R_i^1))/2$ by $P_{1-4}(\cdot)$ and $(P(R_i^0) + P(R_i^1))/2$ by $P_i(\cdot)$, where $i = 1, 2, 3, 4$. The results are shown in Table 1.

Table 1. Proportions of short runs in pseudo random sequences by different chaotic maps.

maps	$P_1(\cdot,\cdot,\cdot)$	$P_2(\cdot,\cdot,\cdot)$	$P_3(\cdot,\cdot,\cdot)$	$P_4(\cdot,\cdot,\cdot)$	$P_{1-4}(\cdot)$
Map 1(a)	0.5,0.5,0.5	0.2499,0.2499,0.2499	0.1239,0.1245,0.1242	0.0625,0.0633,0.0629	0.9369
Map 1(b)	0.5,0.5,0.5	0.2502,0.2502,0.2502	0.1244,0.1250,0.1247	0.0627,0.0625,0.0626	0.9374
Map 1(c)	0.5,0.5,0.5	0.2509,0.2509,0.2509	0.1255,0.1234,0.1244	0.0631,0.0603,0.0617	0.9370
Map 2(a)	0.5,0.5,0.5	0.2513,0.2513,0.2513	0.1243,0.1254,0.1249	0.0609,0.0629,0.0619	0.9381
Map 2(b)	0.5,0.5,0.5	0.2482,0.2482,0.2482	0.1212,0.1230,0.1221	0.0569,0.0614,0.0592	0.9294
Map 3(a)	0.5,0.5,0.5	0.2495,0.2495,0.2495	0.1263,0.1251,0.1257	0.0628,0.0639,0.0634	0.9386
Map 3(b)	0.5,0.5,0.5	0.2486,0.2486,0.2486	0.1231,0.1266,0.1249	0.0616,0.0626,0.0621	0.9356
Equations	0.5,0.5,0.5	0.2514,0.2514,0.2514	0.1283,0.1269,0.1276	0.0669,0.0637,0.0653	0.9443
0-1 i.i.d	0.5	0.25	0.1250	0.0625	0.9375

In Table 1, the expressions in Column 1 correspond to maps or equations as follows:

Map 1(a)— Map 1($\mu = 3.91$); Map 1(b)—Map 1($\mu = 4.00$); Map 1(c)—Map 1($\mu = 3.66$); Map 2(a)—Map 2($L_0 = 31420, z = 2, m = 99999821$); Map 2(b)—Map 2($l_0 = 23156, z = 2, m = 99999787$); Map 3(a)—Map 3($a = 1.4, b = 0.3, x_0 = 0.3,\ y_0 = 0.1, N = 20000$); Map 3(b)—Map 3($a = 1.4, b = 0.3, x_0 = 0.3,\ y_0 = 0.1$); Equations—Lorenz Equations ($\Delta t = 0.0137$) (set by the computer itself); 0-1 i.i.d (independent and identically distributed)—uniform 0-1 random sequences (theoretical values).

For comparison, we cite the results of Ref. [8] as follows:

From Tables 1 and 2, we conclude that the algorithm is suitable for any chaotic map or series and the results are very satisfactory while the algorithms shown in Refs. [1], [8]–[14] have no universality.

Table 2. Proportions of short runs in pseudo random sequences by different chaotic maps obtained by the Monte–Carlo method.

series or maps	$P_1(\cdot)$	$P_2(\cdot)$	$P_3(\cdot)$	$P_4(\cdot)$	$P_{1-4}(\cdot)$
m series	0.4773	0.2345	0.1529	0.0640	0.9287
Logistic	0.5563	0.2441	0.0861	0.0473	0.9338
Chebyshev	0.5787	0.1843	0.0737	0.0598	0.8965
SCQC	0.4854	0.2613	0.1275	0.0704	0.9447
TD-ERCS	0.4857	0.2495	0.1243	0.0706	0.9301

3.4. Statistic proportionality of the 0-1 pseudo random sequences

Because $\{y_{n+k}\}_1^N$ follows a standard mean distribution, theoretical and experimental results both show that the means and the standard deviation are 0.5 for all the maps in Table 1.

3.5. Examples of 0-1 pseudo random sequences

(A) $\{s_{n+k}\}_1^{200}$ for Map 1($\mu = 4.0$):
11011101101010110001100101001101001001110001100001110001110100100110011100
01101011110100010010100100011100110111001100100010011000001110000111100100
01101 00100011110110110100001001111101011100101100101$\cdots$
(B) $\{s_{n+k}\}_{201}^{400}$ for Lorenz Equations(where 200 transient points are omitted):
00110110011110000111100010000101011110000100000001101111000000100101111100
10011110111110101111100100101010010011011010011101110100010111011101011010
11010010001011011010001100001100100001010011101111 ...

3.6. Autocorrelation and cross-correlation

By Eqs. (4) and (5), we obtain autocorrelation and cross-correlation in the pseudo random sequences in Subsection 3.3. The results are shown in Figs. 1 and 2.

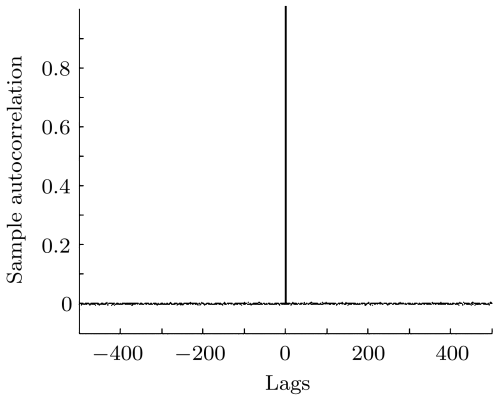


Fig. 1. Autocorrelation in pseudo random sequences generated by Logistic map when $\mu = 4.00$.

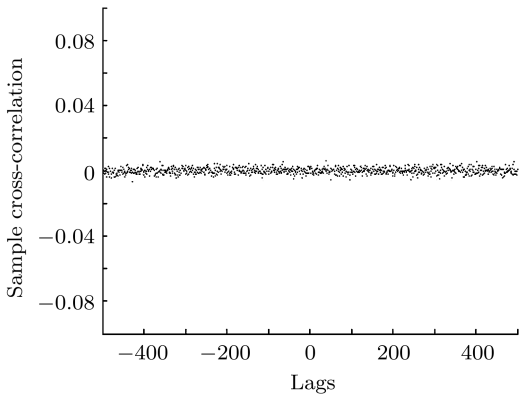


Fig. 2. Cross-correlation between two pseudo random sequences generated by Logistic maps when $\mu = 4.00$ and $\mu = 3.99$.

In addition, the maximum autocorrelation value in Fig. 1 is 0.0059 except when Lags = 0, and the maximum cross-correlation in Fig. 2 is 0.0062. From Figs. 1 and 2 it follows that the correlation function is similar to δ-like function and the cross-function is nearly zero. Comparing Figs. 1 and 2 with the figures in Ref. [11], we conclude that the pseudo random sequences in this paper is of higher quality.

4. Conclusions

The pseudo random sequences generated by the uniform map and that transfer function defined in this paper provide us with a universal algorithm that is suitable for any chaotic map. By the empirical test, proportions of runs, autocorrelation and cross-correlation are all satisfactory. In Section 3, the k in $\{s_{n+k}\}_1^N$ can replaced by k_n, where k_n can vary with n and take any positive integer that is not very large, e.g. from 1 to 10, which means that k, μ, x_0, and maps themselves can act as seeds and consequently the parameter space is enlarged. With these properties, the generator can be applied to many fields such as communication security, electronic games, digital simulation, etc.

References

[1] Sheng L Y, Xiao Y Y and Sheng Z 2008 *Acta Phys. Sin.* **57** 4007 (in Chinese)

[2] Wichmann B A and Hill I D 2006 *Comput. Stat. and Data Anal.* **51** 1614

[3] Sánchez S, Criado R and Vega C 2005 *Math. and Coput. Model.* **42** 809

[4] Gu Q L and Gao T G 2009 *Chin. Phys.* B **18** 84

[5] Long M and Qiu S S 2007 *Chin. Phys.* **16** 2254

[6] Wang L, Wang F P and Wang Z J 2006 *Acta Phys. Sin.* **55** 3964 (in Chinese)
[7] Xu S J and Wang J Z 2008 *Acta Phys. Sin.* **57** 37 (in Chinese)
[8] Sheng L Y, Cao L L, Sun K H and Jiang W 2005 *Acta Phys. Sin.* **54** 4031 (in Chinese)
[9] Xiang F and Qiu S S 2008 *Acta Phys. Sin.* **57** 6132 (in Chinese)
[10] Stojanoski T and Kovarev L 2001 *IEEE Trans. CAS-1* **48** 281
[11] Wang X S and Gan J R 2002 *Chinese J. Comput.* **25** 352 (in Chinese)
[12] Kawamoto S and Horiuchi T 2004 *Int. J. Bifurc. Chaos* **14** 3607
[13] Kohda T 2002 *IEEE Circ. Sys. Mag.* **64** 4
[14] Baranousky A and Daems D 1995 *Int. J. Bifurc. Chaos* **5** 1585
[15] Hou W, Feng G L, Deng W J and Li J P 2008 *Acta Phys. Sin.* **57** 2663 (in Chinese)
[16] Zunino L, Zanin M, Tabak B M, Perez D G and Rosso O A 2009 *Physica* A **388** 14
[17] Wang F L and Yang H H 2009 *Chin. Phys.* B **18** 4042