Robust Chaos Based Random Number Generation for Monte Carlo Simulations

Günyaz Ablay

***Abstract*—**Random number generators have an important role in many engineering applications. In this work, a robust chaotic map based random number generation algorithm is studied for Monte Carlo simulations. For use in simulation based solutions, outputs of the chaotic systems must fit the standard uniform distribution on U(0,1) for accurate solutions. Some robust chaotic maps appear to be good candidates for simulation based solutions due to their robust and uniform outputs. The performance of the proposed chaotic random number generator is evaluated through different statistical methods, and its randomness level and suitability are analyzed with the statistical and visual tests. The effectiveness of the approach is validated with a Monte Carlo solution of a stochastic process.

***Keywords*—** chaos, random number, Monte Carlo, simulation.

I. Introduction

SIMULATIONS usually need a random number source because many statistical, mathematical and physical methods rely on the random samples or stochastic processes. Simulation in engineering and natural sciences is extensively used for working on biological processes, reliability analyses, physical processes and particle transports. Randomness is described by a nondeterministic process and statistically independent of the others. The usages of random numbers are not limited to simulations such that many science and engineering fields need random numbers for a variety of objectives including data encryption, gambling, data sampling and modeling. On the other hand, the usage of random numbers in simulation and cryptographic applications requires somewhat different features. It is well-known that cryptosystems and coding need unpredictable random bits, while the applications of simulations require uniform distribution of the random numbers. To meet application based requirements, today true and pseudo random number generators (RNGs) are used. True RNGs, also known as hardware-based generator, operate by measuring unpredictable natural processes such as thermal noise [1], radioactive decay [2] and atmospheric events [3]. The RNGs are often appropriate to cryptosystems. Pseudo RNGs, also known as software-based generators, use deterministic processes to generate a series of outputs from an initial seed state. Today, the random numbers are directly generated by using the computers. In reality, the computer algorithms use mathematical formulas or pre-calculated tables to generate sequences of pseudo-random numbers that seem random. The most successful pseudo random number algorithms are based on recursive or linear congruential generators and feedback shift registers [4], [5]. Pseudo RNGs are much more cost effective and much faster than the true RNGs, but the randomness level of the pseudo-random numbers depends on the level of randomness of the seed. In addition, all the computer based random numbers are typically periodic numbers with long periods such that the periodicity can be ignored in many applications. Alternatively, chaos based random number algorithms have been studied in recent years by utilizing the aperiodic feature of the chaotic systems.

The utilization of the chaos based cryptosystems has been getting a great deal of attention in the last decade. In such applications the studies have been focused on chaotic random bit generations. On the other hand, the usages of the chaos for simulation based applications have been researched by few studies. The usage of random numbers in simulations is different than the cryptographic applications because the simulations (e.g., Monte Carlo) need fast random number generators and a uniform distribution of random numbers. It is shown in [6] that Chebyshev chaotic maps based Monte Carlo simulations can yield a superefficient solution for some specific integral problems so that its approximation error decreases as fast as $1/N^2$ instead of the conventional 1/N. In [7], it is shown that uniform random distribution can be obtained from logistic maps with appropriate transformations and then can be used in Monte Carlo solutions. These studies show evidence that chaos can be used in simulations, but they are utilized from smooth chaotic maps which exhibit periodic windows. In this work, a robust chaotic map whose output sequences fit uniformly distributed numbers over the range $U(0,1)$ is introduced for directly use in random number based simulations. The robust chaotic map does not have any periodic windows for a wide range of parameter variations. The goal in the usage of robust chaotic maps is to get simple, fast, robust and efficient chaos based solutions for practical Monte Carlo simulations.

Monte Carlo simulations used in many different sciences and engineering disciplines compute the results based on the repeated random sampling and statistical analysis. For a sequence of N independent random observations, a volume integral of a function h is given by

$$\int_V hdV \cong \mu V \pm \sigma \tag{1}$$

Günyaz Ablay, Abdullah Gül University, Department of Electrical-Electronics Engineering, Kayseri, Turkey.

where μ is the sample average defined by

$$\mu = \frac{1}{N}\sum_{k=1}^{N} h(z_k) \tag{2}$$

and σ^2 is the variance given by

$$\sigma^2 = \frac{1}{N}\sum_{k=1}^{N}\left(h(z_k)-\mu\right)^2 \tag{3}$$

For independent uniform random samples z_k, the variance of the approximation decreases at a rate 1/N. The uniform random numbers can be obtained from a discrete-time chaotic system given by

$$z_{k+1} = g(r, z_k) \tag{4}$$

where r is a real-valued system parameter and $g(.)$ is a piecewise linear function, $g : \mathrm{R} \to \mathrm{R}$. The existence of chaos in the system (4) can be shown with the positive Lyapunov exponents and bifurcation diagrams. If the Lyapunov exponent in the chaotic region is always positive, then the chaotic behavior without any periodic windows in that region is called robust chaos. In this work, a robust chaotic map based random number generation algorithm will be introduced for use in Monte Carlo simulations.

II. Chaotic Random Number Generation

The uniform distribution has a critical role in the random number generation. The uniform distribution has random variable Z restricted to a finite interval $[a,b]$, represented by $U(a,b)$, and a probability density function $q(z)$ given by

$$q(z) = \begin{cases} 1/(b-a) & , a \le z \le b \\ 0 & , \text{ otherwise} \end{cases} \tag{5}$$

If we have a distribution denoted by $U(0,1)$, then it is called a standard uniform distribution. If z is a value sampled from the uniform distribution, then the value $(z-a)/(b-a)$ follows the standard uniform distribution, $U(0,1)$.

For use as an RNG, only a small number of chaotic systems can be a good candidate since the uniform distribution is the main concern. To use chaotic sources as RNGs for Monte Carlo simulations, two main features should be satisfied: (i) the (scaled) output of the chaotic map must fit well to the standard uniform distribution, and (ii) the chaotic RNG must be fast enough not to consume too much simulation time. By considering these two conditions, the robust chaotic maps seem to be a good candidate for RNGs. A robust chaotic system is described with no periodic windows in the chaotic attractor for its parameter space [8]. The robust chaos can only occur in piecewise smooth or discontinuous maps whose Lyapunov exponents remain positive throughout the chaotic parameter range. Such chaotic maps including the piece-wise map [9], tent map [10], skew-tent map [11] and binary shift map [12] can be good candidates for uniform random number generation in the form of $U(0,1)$. It is critical to have simplicity in the RNG algorithms because the least solution time and memory space are needed in the realizations. Hence, (2) the following chaotic map is proposed as a chaotic RNG due to its simplicity and robustness:

$$z_{k+1} = \begin{cases} -rz_k + 1, & z_k > 0 \\ 0, & z_k = 0 \\ -rz_k - 1, & z_k < 0 \end{cases} \quad (3) \tag{6}$$

where the parameter r>0. The discrete system (6) has three fixed points, $z_e = (1/(r+1),\ 0,\ -1/(r+1))$, such that nonzero fixed points are stable since $\partial g(z_e)/\partial z = -r$ (4). This map has only two fixed points in the uniform distribution range which can be a significant advantage when the initial condition is selected randomly. The existence of chaos in the system for $r > 1$ can be demonstrated with positive Lyapunov exponents and bifurcation diagrams. For various parameter values of the new chaotic map (6), bifurcation diagrams exhibiting a route to chaos are shown in Fig. 1 for r versus z_k. It is seen from the bifurcation diagram that the chaotic map do not have any periodic orbit for a wide range of system parameter, $1.4 < r \le 2$.

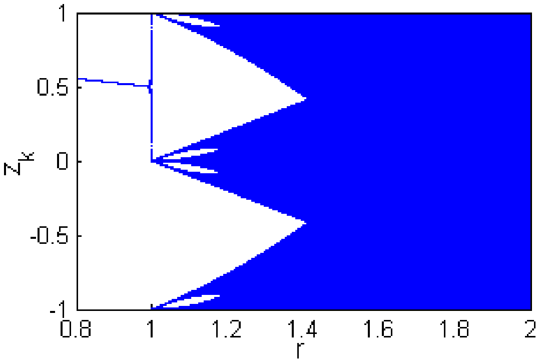


Fig. 1: Bifurcation diagram for the chaotic map (6).

The Lyapunov exponent of the chaotic map is calculated as

$$\Lambda = \ln|r| \tag{7}$$

Therefore, when $|r| > 1$, the Lyapunov exponent is always positive, which means that the chaotic map is robust (i.e., robust chaos). The positive Lyapunov exponent is critical for existence of the chaos and a very useful measure of the randomness, given by the Kolmogorov–Sinai (KS) entropy [13]. The KS entropy, in general, is calculated by the sum of the positive Lyapunov exponents without prior knowledge of the source statistics. Thus, the maximum achievable entropy from the chaotic map (6) is obtained as $\ln|r| = 0.693$.

For random number generation from chaotic map (6), the system parameters are selected as $r = 1.9999$, and the chaotic outputs are scaled to the $U(0,1)$ with $u_k = |x_k|$. Since the quality of the uniform distribution is critical for Monte Carlo simulations, the randomness assessments of the generated random numbers can be conducted with the visual and test statistics based approaches. It is well-known that the quality of the random number generators cannot only be determined by the statistical tests, but they are required to get enough idea about the randomness level of the observations. Firstly, the random number generator is evaluated with visual methods which provide a nice and quick way to get a rough consequence about the generator's performance. Four visual techniques are used in this work: the run sequence plot, histogram, lag plot and autocorrelation plot. Figure 2a shows the run sequence obtained from the chaotic map (6). It shows a random pattern without any periodic, upward or downward trends. The normalized histogram plot of the chaotic RNG output is provided in Fig. 2b for 100 categories. The histogram verifies that the data follows the feature of standard uniform distribution, where there is almost the same number of observations in each category. Figure 3a illustrates the lag plot of the data, which is an effective method for detecting outliers. Existence of some significant outliers is an indication of problems in the random number generator. It is clear that there are no outliers in the figure. Since the chaotic map is one-dimensional, the data points are spread evenly across the symmetric lines (i.e., a good indication of uniformity). The last visual method shown in Fig. 3b is the autocorrelation plot of the chaotic data. It displays that the data is random without any repeating patterns and have property of independence because all the values are in control and all the correlations are small (i.e., inside the standard bounds ±0.0063).

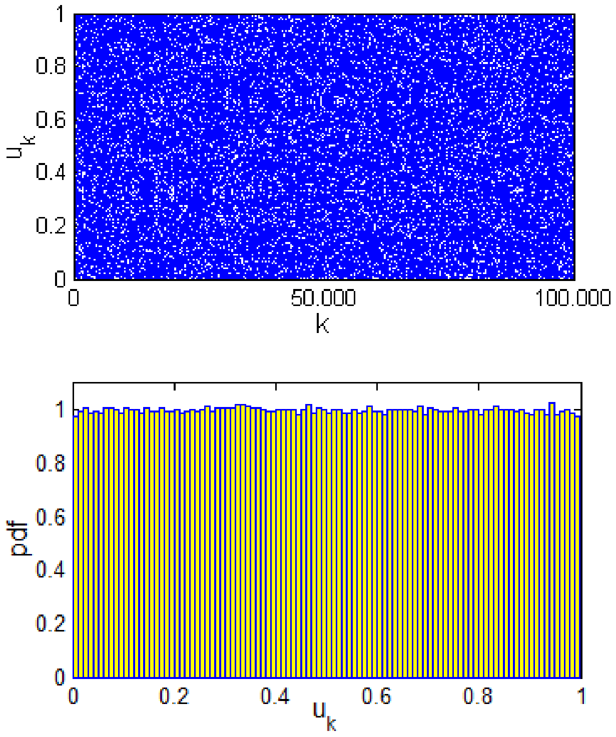


Fig. 2: Run sequence and histogram plots

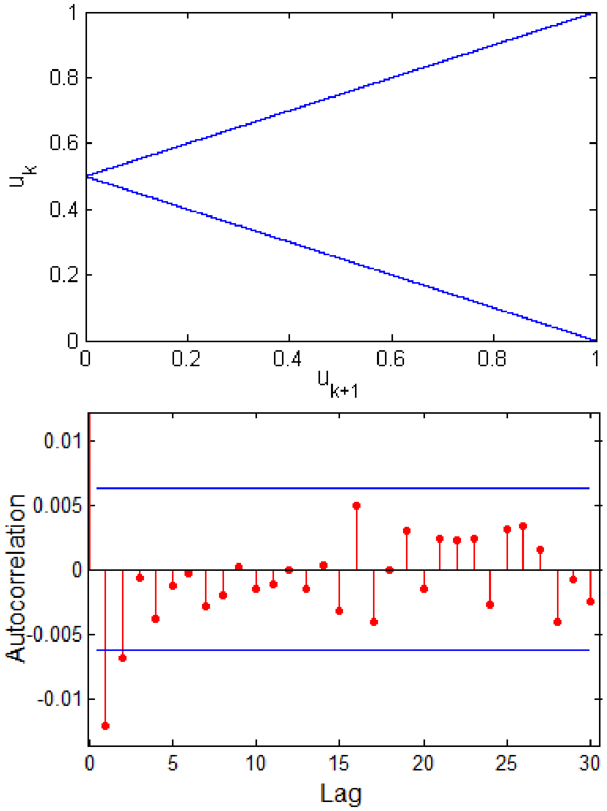


Fig. 3: Lag plot (u_{k+1} vs u_k) and autocorrelation plot

The chaotic random numbers are also evaluated for randomness with some qualitative statistical tests. The statistical tests must be simple and easy enough to implement, and be suitable for analyzing different trends and random numbers. For these reasons, four tests which are commonly used in literature are selected: the chi-square goodness-of-fit test, runs test above-below the median, reverse arrangements test and overlapping sums test. The *chi-square goodness-of-fit test* is a test of distributional accuracy and widely used in the analysis of random numbers. This test is used to measure how closely the generated random numbers follow the uniform distribution. The *runs test above-below the median* is a powerful method in detecting fluctuating trends in the observations and does not need any assumptions about the observations (i.e., a distribution-free test). If a fluctuating trend exists, then it would suggest non-randomness. The *reverse arrangements test* is also a powerful test in detecting bias or monotonic trends in the observations [14]. The test provides highly accurate test results about randomness. The *overlapping sums test* is based on the empirical chi-square test, but it is difficult to use this test on a daily basis.

TABLE I: STATISTICAL TEST RESULTS FOR THE PROPOSED CHAOTIC RNG.

Tests	Critical values	Statistics	Results
Runs test above / below the median	$\lvert z \rvert<1.93$	0.1970	success
Chi-square goodness-of-fit test	$\chi^2_{0.05,\,7} < 14.067$	0.80883	success
Overlapping sums test	$\chi^2_{0.05,\,9} < 16.92$	5.642	Success
Reverse arrangements test	2145<h<2804	2498	success

The visual and test statistics based analysis results show that the chaotic RNG yields a good sequence of random numbers for use in simulation studies. The following application demonstrates the effectiveness of the method for Monte Carlo simulations.

III. Application to a Markov Jump Process

Consider a stochastic reaction process in a volume V given by [15]

$$X \underset{p_2}{\overset{p_1}{\rightleftarrows}} Y \;,\quad 2X \underset{p_4}{\overset{p_3}{\rightleftarrows}} Z \;,\quad W + X \underset{p_6}{\overset{p_5}{\rightleftarrows}} 2X \tag{8}$$

The stochastic model of the process is described with a four-dimensional Markov jump process with the following transition rates [4], [16]

$$\begin{aligned}
&q_V\left((w,x,y,z),(w,x-1,y+1,z)\right) = p_1 x\\
&q_V\left((w,x,y,z),(w,x+1,y-1,z)\right) = p_2 y\\
&q_V\left((w,x,y,z),(w,x-2,y,z+1)\right) = p_3 x(x-1)/(2V)\\
&q_V\left((w,x,y,z),(w,x+2,y,z-1)\right) = p_4 z\\
&q_V\left((w,x,y,z),(w-1,x+1,y,z)\right) = p_5 wx/V\\
&q_V\left((w,x,y,z),(w+1,x-1,y,z)\right) = p_6 x(x-1)/(2V)
\end{aligned} \tag{9}$$

where $q_V(.)$ is the probability function, $p_1,\ldots,p_6$ are reaction rates and V is the volume. While the process is stochastic, the deterministic model of this process can be used for fast evaluations as follows

$$\begin{aligned}
&\dot{w} = -p_5 wx + 0.5 p_6 x^2\\
&\dot{x} = -p_1 x + p_2 y - p_3 x^2 + 2 p_4 z + p_5 wx - 0.5 p_6 x^2\\
&\dot{y} = p_1 x - p_2 y\\
&\dot{z} = 0.5 p_3 x^2 - p_4 z
\end{aligned} \tag{10}$$

The solution of the deterministic model can be obtained from numerical methods, e.g. Runge-Kutta 4th order method. The stochastic simulation of the Markov jump process is also straightforward for a finite volume and a finite time-step. Figure 4 shows the convergence of the stochastic process to the deterministic limit with increasing volume (V=1 and V=10). For instance, the solution is indistinguishable from the deterministic results if we choose V=100. It is clear that the stochastic solution approaches to the deterministic results while the volume increase also needs a much longer solution time and a much larger memory space. Figure 5 displays time responses of the process states with deterministic limits for the volume V=1. It is seen that the Markov jump process approaches steady-state response of the process as time increases. It is clear that the stochastic solutions are compatible with the deterministic results, and thus, the results obtained with the use of chaotic RNG are quite accurate and effective.

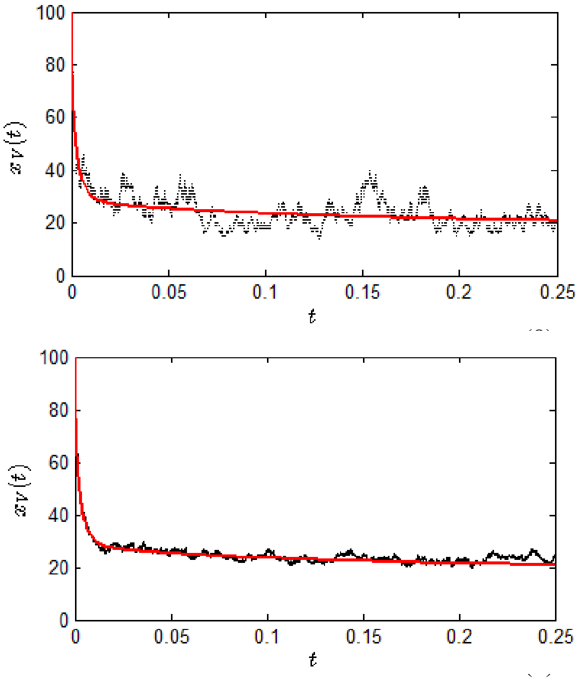
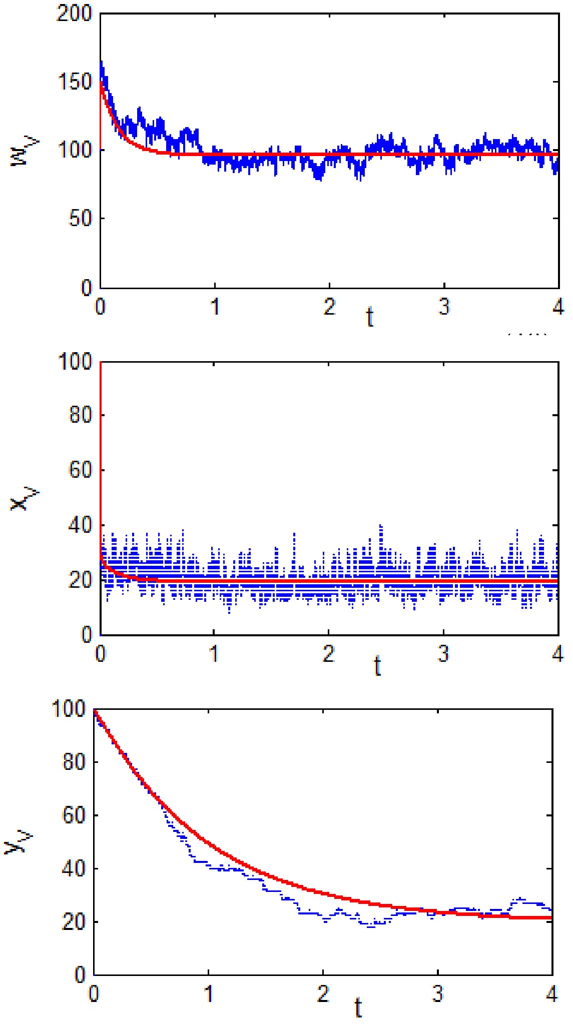


Fig. 4: Deterministic (red line) and stochastic solutions for the state x for a given volume, (a) V=1, (b) V=10



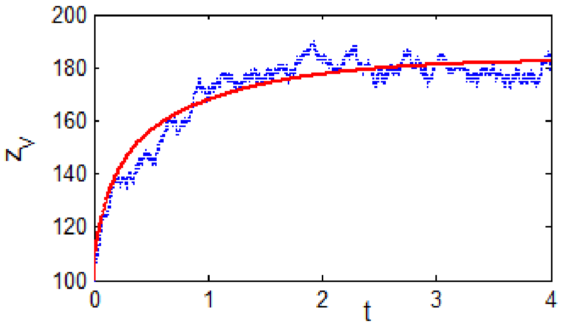


Fig. 5: Deterministic (red line) and stochastic (blue dots) solutions for all states for V=1

IV. Conclusion

The practical random numbers are directly generated by using the computer algorithms including linear congruential generators and feedback shift registers. All the computer based random numbers have periodic behaviors with long periods in which the periodicity is usually ignored in many applications. On the other hand, random numbers can also be generated from robust chaotic maps by utilizing the aperiodic, deterministic, ergodic and mixing feature of the chaotic dynamics.

This work provides a robust chaotic map based random number generation and its application to Monte Carlo simulations. The chaotic map output fits standard uniformly distributed numbers over the range $U(0,1)$ and does not have any periodic windows for a wide range of control parameters. In general, chaotic RNGs are similar to the conventional recursive random number algorithms, but they yield aperiodic results with very simple equations and eliminate the length of the conventional RNG algorithms, e.g., Mersenne Twister [17]. In addition, the chaotic RNGs are able to provide fast, high-quality and practical solutions with easy implementations in basic embedded systems or microprocessors. This work shows that the robust chaotic maps are able to provide simple, fast and efficient chaos based solutions for practical applications.

Acknowledgment

This work was supported by Research Fund of the Abdullah Gül University under project number FAB-2015-4.

References

[1] D. C. Ranasinghe, D. Lim, S. Devadas, D. Abbott, and P. H. Cole, “Random numbers from metastability and thermal noise,” *Electronics Letters*, vol. 41, no. 16, pp. 13–14, Aug. 2005. http://dx.doi.org/10.1049/el:20051559

[2] J. Walker, “HotBits: Genuine Random Numbers,” 2016. [Online]. Available: https://www.fourmilab.ch/hotbits/. [Accessed: 12-Mar-2016].

[3] M. Haahr, “RANDOM.ORG - Introduction to Randomness and Random Numbers,” 2016. [Online]. Available: https://www.random.org/randomness/. [Accessed: 12-Mar-2016].

[4] D. P. Kroese, T. Taimre, and Z. I. Botev, *Handbook of Monte Carlo Methods*, 1 edition. Hoboken, N.J: Wiley, 2011. http://dx.doi.org/10.1002/9781118014967

[5] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996. http://dx.doi.org/10.1201/9781439821916

[6] C.-A. Yang, K. Yao, K. Umeno, and E. Biglieri, “Using Deterministic Chaos for Superefficient Monte Carlo Simulations,” *IEEE Circuits and Systems Magazine*, vol. 13, no. 4, pp. 26–35, Fourthquarter 2013.

[7] J. A. R. Blais and Z. Zhang, “Exploring pseudo- and chaotic random Monte Carlo simulations,” *Computers & Geosciences*, vol. 37, no. 7, pp. 928–934, Jul. 2011. http://dx.doi.org/10.1016/j.cageo.2011.01.009

[8] S. Banerjee, J. A. Yorke, and C. Grebogi, “Robust Chaos,” *Physical Review Letters*, vol. 80, no. 14, pp. 3049–3052, Apr. 1998. http://dx.doi.org/10.1103/PhysRevLett.80.3049

[9] G. Ablay, “Chaotic map construction from common nonlinearities and microcontroller implementations,” *International Journal of Bifurcation and Chaos*, vol. to be appear, 2016.

[10] I. Campos-Cantón, E. Campos-Cantón, J. S. Murguía, and H. C. Rosu, “A simple electronic circuit realization of the tent map,” *Chaos, Solitons & Fractals*, vol. 42, no. 1, pp. 12–16, Oct. 2009. http://dx.doi.org/10.1016/j.chaos.2008.10.037

[11] T. Stojanovski, J. Pihl, and L. Kocarev, “Chaos-based random number generators. Part II: practical realization,” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 3, pp. 382–385, Mar. 2001. http://dx.doi.org/10.1109/81.915385

[12] A. S. Lima, I. C. Moreira, and A. M. Serra, “Transition between the tent map and the Bernoulli shift,” *Physics Letters A*, vol. 190, no. 5–6, pp. 403–406, Aug. 1994. http://dx.doi.org/10.1016/0375-9601(94)90723-4

[13] R. Frigg, “In what sense is the Kolmogorov-Sinai Entropy a measure for chaotic behaviour? Bridging the gap between dynamical systems theory and communication theory,” *British Journal for the Philosophy of Science*, vol. 55, no. 3, pp. 411–434, Sep. 2004. http://dx.doi.org/10.1093/bjps/55.3.411

[14] J. S. Bendat and A. G. Piersol, *Random Data: Analysis and Measurement Procedures*, 4 edition. Wiley, 2011.

[15] D. T. Gillespie, “A general method for numerically simulating the stochastic time evolution of coupled chemical reactions,” *Journal of Computational Physics*, vol. 22, no. 4, pp. 403–434, Dec. 1976. http://dx.doi.org/10.1016/0021-9991(76)90041-3

[16] P. K. Pollett and A. Vassallo, “Diffusion Approximations for Some Simple Chemical Reaction Schemes,” *Advances in Applied Probability*, vol. 24, no. 4, pp. 875–893, 1992. http://dx.doi.org/10.2307/1427717

[17] M. Matsumoto and T. Nishimura, “Mersenne Twister: A 623-dimensionally Equidistributed Uniform Pseudo-random Number Generator,” *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, pp. 3–30, Jan. 1998. http://dx.doi.org/10.1145/272991.272995