CYBERNETICS AND PHYSICS, VOL. 14, NO. 4, 2025, 363–370

EXPLOITING GEOMETRIC SYMMETRY IN CHAOTIC MAPS FOR NEXT-GENERATION PSEUDO-RANDOM NUMBER GENERATOR DESIGN

Mokhtar Ouamri
Department of Physics
Faculty of Matter Sciences
University of Tiaret
Algeria
mokhtar.ouamri@univ-tiaret.dz

Article history:
Received 07.11.2025, Accepted 20.12.2025

Abstract

In this work, we propose a new two-dimensional chaotic map defined by sinusoidal and cubic nonlinear interactions. The system is characterized by the presence of a geometric attractor with particularly complex and symmetrical properties. This structure is validated by detailed phase portraits supported by a maximal positive Lyapunov exponent. A bifurcation diagram also reveals rich dynamical transitions, including the appearance of chaos as system parameters vary, showing that chaos is a transient phase of the system. Taking advantage of these strong chaotic properties, we design a pseudo-random number generator (PRNG) based on map iterates. The proposed PRNG generates binary sequences by extracting bits from the decimal parts of chaotic trajectories and storing them in binary format for evaluation. The quality of the generated sequences is rigorously tested using the NIST SP 800-22 statistical suite and ENT test respectively, and the results confirm high-quality randomness. Furthermore, entropy is close to optimal levels, and autocorrelation analysis demonstrates statistical independence between bits. Given its simplicity, high sensitivity to initial conditions and excellent random characteristics, this PRNG is a promising candidate for applications in cryptography, secure communications and stochastic simulation systems.

Key words

Chaotic Map, PRNG, Symmetric Attractor, Information security.

Introduction

In recent decades, chaotic behaviour [Tsonis, 2012] has attracted a lot of attention in mathematics and physics, as well as in engineering and the sciences of life. Chaos, which is defined as deterministic yet unpredictable dynamics, has become a part of many aspects of daily life, including the weather [Tsonis and Elsner, 1989], ecological systems [Upadhyay et al., 1998], economic markets [Guegan, 2009], cybernetical neuroscience [Babich et al., 2025], and biological rhythms [Olsen and Degn, 1985]. The ubiquity of chaos in natural and artificial systems has stimulated researchers to exploit its intrinsic properties such as sensitivity to initial conditions, ergodicity, and topological mixing for technological applications, especially in secure communications and information processing.

Pseudo-random number generators (PRNGs) [Bhattacharjee and Das, 2022] are of fundamental importance in numerous fields, including cryptography [Al-Mhadawi et al., 2023], simulations [Schoo et al., 2005], machine learning [Amigo et al., 2021] and hardware testing [McCollum et al., 2003]. Classic PRNGs can be categorised into several distinct classifications: Xorshift generators [Marsaglia, 2003] are reliant on bit-by-bit operations, while state transition functions power the well-known Mersenne Twister [Matsumoto and Nishimura, 1998]. Cryptographically secure PRNGs (CSPRNGs), of which there are examples based on the Advanced Encryption Standard (AES) [Daemen and Rijmen, 2005], generate pseudo-random outputs by means of block cipher operations. This provides strong resistance to reverse engineering and statistical attacks. In contrast, hardware-based PRNGs [Gupta and Chauhan, 2022] leverage physical phenomena such as thermal noise, ring oscillators or metastability in digital circuits to generate true randomness. However, these methods often necessitate greater implementation costs and complexity. In

recent years, chaos-based pseudorandom number generators [Naik and Singh, 2024] have gained in popularity due to their intrinsic unpredictability, sensitivity to initial conditions and aperiodic nature. These numerical chaotic generators cross the boundary between mathematical nonlinearity and practical feasibility, making them particularly attractive for software implementations and lightweight embedded cryptographic systems.

A good PRNG must produce sequences that are statistically indistinguishable from truly random numbers. It must have several critical properties: a uniform distribution, low autocorrelation, a long period without repetition, as well as high entropy and sensitivity to initial conditions for cryptographic applications. In addition, it must resist prediction and pass recognized statistical tests such as NIST SP 800-22 [Bassham et al., 2010], ENT [Walker, 2008] test or Diehard [Marsaglia, 1996]. Efficiency, portability and ease of implementation are also important factors, especially in resource-constrained systems.

In response to this challenge, the research community has proposed several promising approaches, including those presented in [Agarwal, 2021; Kiran et al., 2023; Camara et al., 2020; Araki et al., 2024; Ouamri, 2025]. Kiran et al.'s approach [Kiran et al., 2023] presents a chaos-based PRNG designed on the basis of two discrete chaotic models: Hem's cubic map and Ricker's population model. Binary sequences are generated by iterating these systems with specific initial parameters, producing highly sensitive results suitable for RGB image encryption. Camara et al.'s approach [Camara et al., 2020] explores a biometrics-inspired approach, where human gait data captured by six triaxial sensors (accelerometers and gyroscopes) serve as a source of dynamic entropy. The processed signals produce binary sequences that pass ENT, Diehard and NIST tests, validating their suitability for use in safety-critical IoT systems. Araki et al. proposes in [Araki et al., 2024] a new chaos-based PRNG architecture using an extremum coding mechanism. By exploiting the intrinsic butterfly effect in chaotic systems, the authors introduce a method to generate a random extremum coded sequence (RECS). This RECS is used to dynamically update the modulation parameters of a chaotic system, thus improving the statistical quality of the generated pseudo-random sequences. To strengthen cryptographic strength and approach true chance, the design integrates SHA3-256 hash in the output pipeline. Combined, these techniques produce a hybrid design that merges deterministic chaos with cryptographic post-processing. The proposed system has superior statistical performance compared to previous designs and passes all standard random test suites. Additionally, in [Ouamri, 2025], we introduced a novel PRNG based on a two-dimensional chaotic system, the Cubic Trigonometric Coupled Map (CTCM). By exploiting the strong nonlinear interactions and complex dynamics of the CTCM, our generator attains near-optimal entropy levels and passes standard randomness tests, demonstrating its suitability for high-security applications.

Chaotic maps using trigonometric functions are of particular interest in the field of nonlinear dynamics. These functions, such as sine and cosine, introduce not only essential non-linearities but also periodic behaviors that trap trajectories in complex patterns. The Sine Map [Pareek et al., 2005], for example, is an emblematic model that nevertheless illustrates in a simple way how dynamic systems can produce a wide range of chaotic behaviors. The Clifford Attractor [Peitgen et al., 2004], on the other hand, demonstrates how interactions between several trigonometric variables can reveal fascinating geometric structures that are not only random but also symmetrical, demonstrating the richness and diversity of the strange attractors predicted by these systems. The ability of these attractors to exhibit fractal boundaries and dense orbits means they have great potential as generators of pseudorandom numbers.

In this article, we present a new two-dimensional chaotic map, called Symmetric Sin-Cos Cubic Map (SSCCM). This map is inspired by chaotic systems based on the sine function and has characteristics of symmetry and non-linearity similar to those of the Clifford attractor. The SSCCM shows a rich and complex chaotic behavior, including a high sensitivity to initial conditions, fractal geometry, and a large parameter space leading to sustained chaos. Taking advantage of these dynamic properties, we propose a design methodology for a new chaos-based pseudorandom number generator. The resulting PRNG aims to achieve high entropy, low correlation and appropriate statistical randomness for cryptographic and secure communication applications.

The remainder of this paper is organized as follows. In Section 2, we introduce the proposed chaotic system, the Symmetric Sin-Cos Cubic Map (SSCCM), and analyze its dynamical properties through the computation of the Lyapunov exponent, the bifurcation diagram, and phase portraits to demonstrate its chaotic behavior. In Section 3, we present the design of a chaotic pseudo-random number generator (PRNG) based on the SSCCM. The statistical quality of the generated sequences is evaluated using standard tests, including the NIST SP 800-22 test suite, Shannon entropy analysis, and sensitivity to initial conditions. Finally, Section 4 concludes the paper and discusses possible future extensions of the proposed system.

1 Symmetric Sin-Cos Cubic Map (SSCCM) model

1.1 Model presentation

In this section, we propose a novel two-dimensional discrete-time dynamical system called the *Symmetric Sin-Cos Cubic Map (SSCCM)* that is defined as follows:

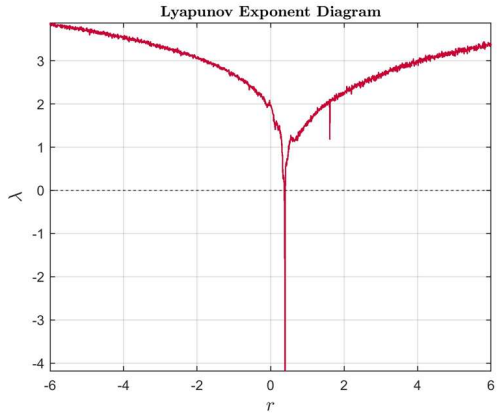

Figure 1: Lyapunov exponent λ as a function of r for the SSCCM with fixed parameters $a = 3.6$ and $b = 3.2$, illustrating the onset and persistence of chaotic behavior.

$$\begin{cases} x_{n+1} = ar\sin^3(ax_n) - br\cos^3(by_n) - a\sin(ax_n) \\ y_{n+1} = ar\cos^3(ax_n) - br\sin^3(by_n) - b\sin(bx_n) \end{cases} \tag{1}$$

where x_n and y_n represent the state variables at discrete time step n, while parameters a, b, and r respectively control system complexity and nonlinearity.

1.2 Sensitivity to Initial Conditions: Lyapunov Exponent Analysis

One of the characteristics of chaotic systems is their sensitivity to initial conditions, meaning that arbitrarily close initial states diverge exponentially with time. This phenomenon is measured quantitatively using the Lyapunov exponent λ [Sandri, 1996], which is defined by:

$$\lambda = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \ln|\mu_i| \tag{2}$$

where μ_i denotes the largest eigenvalue (in modulus) of the Jacobian matrix $J(X_i)$ at iteration i, and $X_i = [x_i, y_i]^T$.

A positive value of λ indicates chaos, meaning that small differences in initial conditions grow exponentially over time.

For this purpose, we calculated λ for different values of the control parameter r, while keeping $(a, b) = (3.6, 3.2)$. The diagram in Figure 1 illustrates the behavior of λ as a function of $r \in [-6, 6]$. As can be seen, the Lyapunov exponent remains strictly positive over a wide range of parameters, reaching values in excess of 2.5, confirming the highly chaotic nature of the system. However, we noticed a notable dip around $r \approx 0.387$, where the exponent briefly becomes negative, indicating a stable periodic or quasi-periodic regime. Beyond this critical region, the system quickly reverts to chaotic dynamics, with λ increasing almost linearly with r.

1.3 Bifurcation analysis of SSCCM

This section investigates the long-term behavior of the SSCCM system as a function of the control parameter r. In the figure 2, we present the bifurcation diagrams of the SSCCM system by varying the control parameter r, while keeping the parameters $a = 3.6$ and $b = 3.2$ fixed. We observe that the x and y components exhibit complex bifurcation structures, reflecting the system's high sensitivity to parameter changes. As r increases, we notice a transition from regular dynamics to irregular scattering, suggesting the emergence of chaos. The calculation of Lyapunov exponents confirms that the system is mainly chaotic in this range. However, we identify a remarkable window around $x \approx 0.385$, where the structure temporarily reduces to a narrow band and the Lyapunov exponent becomes negative, indicating a return to stable, non-chaotic behavior. This suggests that the system intermittently alternates between order and chaos during the variation of r, which is characteristic of nonlinear dynamical systems. This analysis provides a better understanding of how slight changes in parameters can induce qualitative changes in the long-term behavior of the SSCCM.

1.4 Phase portrait analysis

This section is devoted to the analysis of the phase portrait generated by the SSCCM model for the parameter set $(a, b, r) = (3.6, 3.2, 6)$. Figure 3 reveals a remarkable geometric structure consisting of symmetrical petal-shaped loops centered around the origin. This intricate organization reflects the presence of a strange attractor, which emerges from the complex nonlinear interactions between the system's variables.

Furthermore, the chaotic nature of the dynamics is quantitatively supported by the computation of the maximum Lyapunov exponent, evaluated at $\lambda = 3.354$, thereby confirming the system's sensitivity to initial conditions and the presence of deterministic chaos.

The observed attractor, though chaotic, remains both bounded and highly structured, reflecting dynamics that

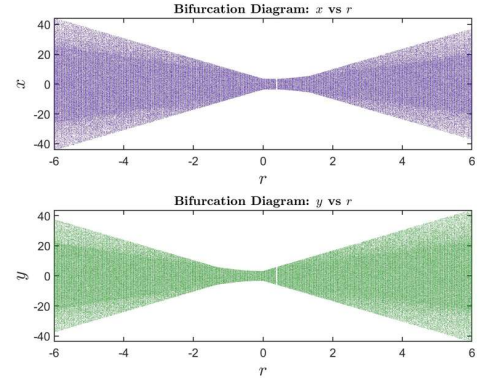

Figure 2: Bifurcation diagrams x vs. r (top) and y vs. r (bottom) for the SSCCM system.

are simultaneously unstable and confined. This dual nature, coupled with the richness of its geometric features, suggests the presence of strong topological mixing and a dense distribution of trajectories in the phase space. Such characteristics are particularly desirable in applications including pseudo-random number generation (PRNG), secure encryption systems, and complex signal modulation, where unpredictability and sensitivity to initial conditions are essential.

2 Chaotic PRNG Based SSCCM model

We propose a novel pseudo-random number generator (PRNG) based on a two-dimensional chaotic map, whose behavior is governed by five secret parameters: the initial conditions x_0, y_0, and the nonlinear coefficients a, b, and r. These parameters collectively constitute the cryptographic key of the system, offering a vast key space and high sensitivity to slight variations. The generator functions by iteratively applying a deterministic nonlinear system, yielding chaotic trajectories in the (x_n, y_n) phase space. A critical requirement for secure operation is that the system possesses a positive Lyapunov exponent, which guarantees exponential divergence of nearby trajectories, thereby enhancing the unpredictability and cryptographic strength of the output. Binary sequences are extracted from the chaotic orbit by computing the normalized value:

$$v_n = |\pi x_n + 3y_n| \bmod 1, \tag{3}$$

which ensures that $v_n \in [0, 1]$. This value is then scaled as

$$z_n = \lfloor v_n \times 2^{32} \rfloor \tag{4}$$

to produce a 32-bit unsigned integer. The resulting integer is subsequently converted into a binary string composed of ASCII characters '0' and '1'. This procedure is repeated at each iteration, enabling the generation of a continuous binary stream suitable for cryptographic or stochastic applications.

2.1 Entropy assessment

In this section, we evaluate the entropy [Mertens and Bauke, 2004], a physical quantity that quantifies the degree of randomness and unpredictability of a pseudo-random number generator (PRNG). Entropy is defined as:

$$\text{Entropy} = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{5}$$

where p_i is the probability of occurrence of symbol i, and n is the total number of possible symbols.

In our experiment, the binary sequence generated by the chaotic PRNG was converted to a byte-level sequence. We then followed the evolution of entropy as a function of sequence length, as shown in Fig. 5. We found that entropy increases rapidly during the first few megabytes, reaching a value close to 7.99998 bits per byte. This means, in effect, that PRNG produces highly random and unpredictable data from the very start of generation. Interestingly, the entropy stabilizes around a remarkably constant value of 7.9999982 bits per byte once the sequence length surpasses 9×10^7 bytes. At higher levels, this point of convergence shows that the suggested PRNG can produce data with almost perfect randomness.

2.2 NIST-Based randomness assessment

We applied the NIST Statistical Test Suite, a commonly used benchmark for evaluating randomness in cryptographic generators, to assess the statistical quality of the binary sequences generated by our chaotic PRNG. Aspects of randomness such as frequency balance, pattern distribution, and structural complexity are all addressed by the fifteen statistical procedures that make up this test suite.

Two separate bitstreams of length 50×10^6 bits were produced using the proposed system for the evaluation. To satisfy the test suite's input requirements, each stream was divided into 50 sub-sequences of 10^6 bits.

The initial conditions were selected to examine the sensitivity of the system, and the generation was carried out with fixed parameters $(a, b) = (3.6, 3.2)$. For the first sequence, the initial conditions $(x_0, y_0) = (0.2, 0.3)$ were used, while a slightly perturbed version $(x_0, y_0) = (0.2 + 10^{-10},\ 0.3 + 10^{-10})$ was used for the second sequence.

Both Sequence 1 and Sequence 2 pass all of the standard NIST randomness tests, as seen in Table 1, according to the statistical analysis based on the p-value, since all of the p-values remain above the usual significance threshold of 0.01. This demonstrates that both sequences exhibit acceptable statistical properties of randomness.

2.3 ENT test assessment

In this section, we tested a 100,000,000-byte sequence using the ENT suite [Walker, 2008], which assesses the statistical quality of a pseudo-random generator by measuring the randomness, uniformity, independence and entropy of the data produced. The results of this test are summarized in the table 2. it is clear that the PRNG passes all empriques tests. The measured entropy is 7.999998 bits per byte, which is very close to the maximum value of 8 bits, indicating an almost perfect distribution of the 256 possible symbols. The estimated optimal compression is 0%, thus confirming that the sequence contains no regularities that can be exploited by a compression algorithm. Furthermore, the Chi-square test statistic is 256.11 per 100,000,000 samples, and such a value would be exceeded 46.88% of the time by a perfectly random generator. This result indicates that the distribution of values is very close to that of an 8-bit

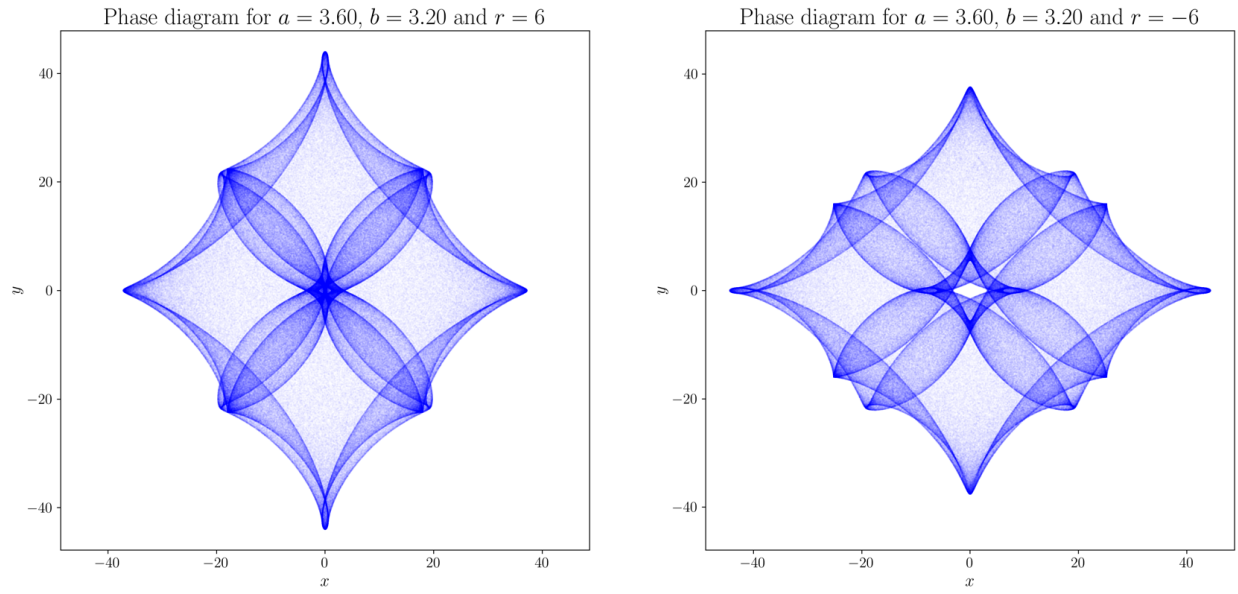

Figure 3: Phase portraits of the SSCCM system for $a = 3.6$ and $b = 3.2$, with two opposite values of the control parameter r. Left: $r = 6$ shows a typical chaotic attractor with dense geometry. Right: $r = -6$ preserves the chaotic nature, but reveals a distinct geometric configuration due to the sign reversal.

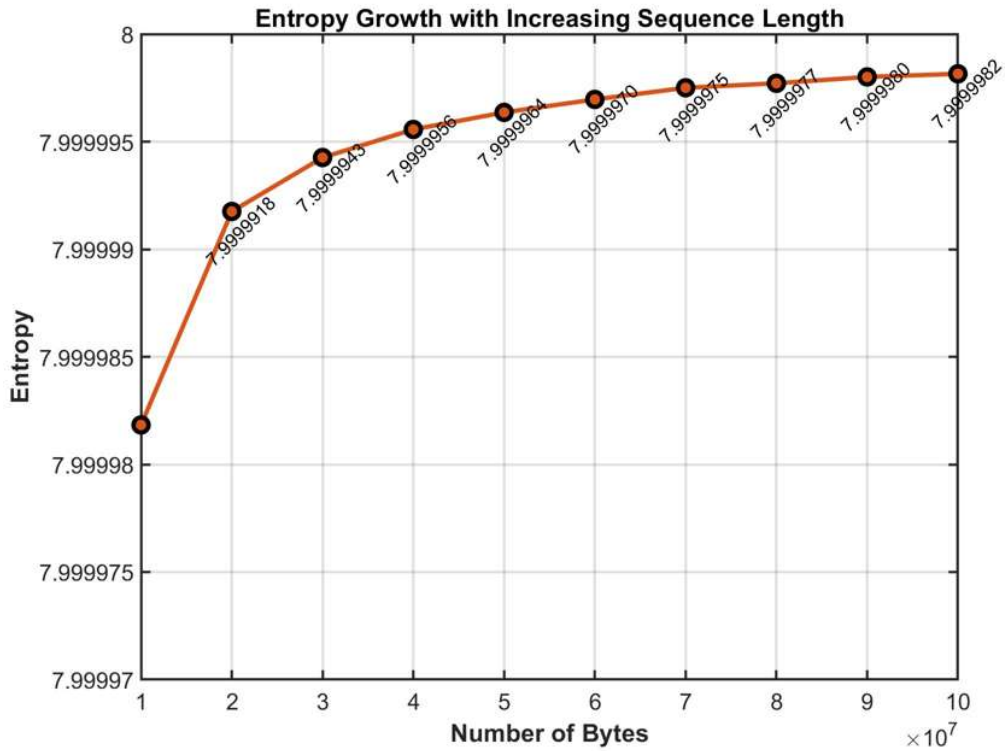

Figure 4: Estimated entropy versus byte length for the SSCCM-based PRNG.

discrete uniform distribution. We have also found the arithmetic mean of the bytes to be 127.4719, which is very close to the expected mean of 127.5 for an 8-bit uniform distribution. As for the estimate of π by the Monte Carlo method, we obtained $\pi \approx 3.142833246$, with a relative error of just 0.04%, further reinforcing the idea of a good dispersion of the simulated points. Furthermore, we noted that the serial correlation coefficient was 0.000071, a near-zero value that attests to statistical independence between successive bytes.

Table 1: Results of NIST Statistical Tests for Two Chaotic Sequences

Statistical Test	**Sequence 1**	**Sequence 2**
Frequency	0.319084	0.383827
Block Frequency	0.262249	0.051942
Cumulative Sums (Forward)	0.096578	0.739918
Cumulative Sums (Reverse)	0.319084	0.699313
Runs	0.419021	0.699313
Longest Run	0.574903	0.739918
Rank	0.911413	0.883171
FFT	0.883171	0.616305
Overlapping Template	0.319084	0.534146
Universal	0.816537	0.289667
Approximate Entropy	0.213309	0.085587
Random Excursions	0.931952	0.534146
Random Excursions Variant	0.772760	0.004301
Serial	0.534146	0.299251
Linear Complexity	0.779188	0.426145

Table 2: Entropy and Randomness Evaluation Metrics

Measure	**Value**	**Optimal Value**
Entropy	7.999998 bits per byte	8 bits per byte
Optimal compression rate	0% (no detectable redundancy)	0%
Chi-square (100,000,000 samples)	256.11 (exceeded 46.88% of the time)	Random $\approx$ 50%
Arithmetic mean of byte values	127.4719	127.5
Monte Carlo estimation of π	3.142833246 (error: 0.04%)	3.1416
Serial correlation coefficient	0.000071	0.0

Computational Performance Evaluation

In this section, we evaluated the efficiency of our PRNG by generating 50 million bits on a computer with a Ryzen 7 5700X processor and 16 GB RAM, performing multiple attempts to ensure consistency. The average generation time was around 0.168 seconds per megabit, demonstrating good randomness and efficiency, suitable for real-time or large-scale applications.

Key Space Analysis

This section evaluates the security robustness of our chaotic pseudorandom generator (PRNG) by estimating the size of its key space. In our case, the secret key consists of five real parameters: the initial conditions x_0 and y_0, as well as the control parameters r, a and b, all belonging to the set of real numbers. The effective entropy of the key depends on the system architecture and the standard for representing floating-point numbers. In accordance with the IEEE 754 standard, real numbers are generally represented in 32-bit (single-precision) or 64-bit (double-precision) formats. In single precision, each parameter provides around 32 bits of entropy, while in double precision, it provides around 64 bits. As a result, the total key space can reach up to 2^{160} for 32-bit systems and 2^{320} for 64-bit systems. This considerable key space, combined with high sensitivity to initial conditions, guarantees high resistance to brute-force attacks and underlines the suitability of our PRNG for cryptographic and security-critical applications.

2.4 Comparative Analysis of PRNGs Based on ENT Metrics

In this section, we used the `ENT` statistical test suite to evaluate and compare the randomization quality of different pseudorandom number generators. Consequently, the results obtained allow us to make a fair and standardized comparison between various PRNG architectures.

Among the methods tested in the table 3, our proposed PRNG stands out by showing exceptional performance on all indicators. It reaches an entropy of 7.999998 bits per byte, which makes it practically indistinguishable from a true random source and places it on the same level as the best references, such as those of Camara *et al.* and S. Araki *et al.* Moreover, the result of the chi-square (256.11 with a p-value of 46.88%) indi-

cates a very uniform byte distribution, superior to several other methods that showed insufficient or excessive dispersion. In addition, the arithmetic mean of 127.4719 is extremely close to the ideal value of 127.5, thus confirming a balanced use of the entire byte range. Furthermore, the Monte Carlo estimate of π reveals a very low error (0.04%), and the serial correlation coefficient of 0.000071 confirms the strong independence between successive values. Consequently, these results highlight the robustness, the unpredictability, and the statistical reliability of our approach, while highlighting its suitability for secure communications and implementations on light hardware. We saw in this paper that Symmetric Sin-Cos Cubic Map model provides a strong mathematical basis for generating high quality pseudorandom sequences. This two-dimensional chaotic system, combining sinusoidal, cosinoidal and cubic non-linearities, generates a rich, complex and difficult to predict dynamics. Its chaotic character is confirmed by a strictly positive Lyapunov exponent and by a fractal structure of the attractor, attesting to a high sensitivity to initial conditions and non-periodic behavior.

3 Conclusion

The Sin-Cos Cubic Map (SSCCM)-based pseudorandom number generator (PRNG) is built around simple arithmetic operations, making it easy to implement even in resource-constrained environments. However, due to the intrinsic complexity of the SSCCM model, the generator manages to produce a high quality of randomness, stemming from the non-linear geometric symmetries embedded in its chaotic dynamics. Experimental evaluations confirm its effectiveness: the generator displays entropy values close to the ideal 8 bits per byte, minimal serial correlation, and a statistically uniform distribution. These features enable it to pass benchmark chance tests such as ENT, chi-square, and Monte Carlo estimation of π. In short, the PRNG derived from the Sin-Cos Cubic Map represents an attractive balance between simplicity, performance and security. Its properties make it a strong candidate for lightweight cryptographic applications, embedded system security, and secure communication protocols.

References

Agarwal, S. (2021). Designing a pseudo-random bit generator using generalized cascade fractal function. *Chaos Theory and Applications*, **3** (1), pp. 11–19.

Al-Mhadawi, M. M., Albahrani, E. A., and Lafta, S. H. (2023). Efficient and secure chaotic prng for color image encryption. *Microprocessors and Microsystems*, **101**, pp. 104911.

Amigo, G., Dong, L., and Ii, R. J. M. (2021). Forecasting pseudo random numbers using deep learning. In *2021 15th International Conference on Signal Processing and Communication Systems (ICSPCS)*, IEEE, pp. 1–7.

Araki, S., Wu, J.-H., and Yan, J.-J. (2024). A novel design of random number generators using chaos-based extremum coding. *IEEE Access*, **12**, pp. 24039–24047.

Babich, N., Chen, O., Chulkin, V., Marzel, E., Rybalko, A., and Fradkov, A. (2025). Outline of cybernetical neuroscience. *Cybernetics and Physics*, **14** (1), pp. 13–18.

Bassham, L. E., Rukhin, A. L., Soto, J., Nechvatal, J. R., Smid, M. E., Leigh, S. D., Levenson, M., Vangel, M., Heckert, N. A., and Banks, D. L. (2010). A statistical test suite for random and pseudorandom number generators for cryptographic applications.

Bhattacharjee, K. and Das, S. (2022). A search for good pseudo-random number generators: Survey and empirical studies. *Computer Science Review*, **45**, pp. 100471.

Camara, C., Martín, H., Peris-Lopez, P., and Entrena, L. (2020). A true random number generator based on gait data for the internet of you. *IEEE Access*, **8**, pp. 71642–71651.

Daemen, J. and Rijmen, V. (2005). Rijndael/aes. In *Encyclopedia of Cryptography and Security*, pp. 520–524. Springer.

Guegan, D. (2009). Chaos in economics and finance. *Annual Reviews in Control*, **33** (1), pp. 89–93.

Gupta, M. D. and Chauhan, R. (2022). Recent development of hardware-based random number generators on fpga for cryptography. In *VLSI, Microwave and Wireless Technologies: Select Proceedings of ICVMWT 2021*, pp. 489–500. Springer.

Kiran, H. E., Akgul, A., Yildiz, O., and Deniz, E. (2023). Lightweight encryption mechanism with discrete-time chaotic maps for internet of robotic things. *Integration*, **93**, pp. 102047.

Marsaglia, G. (1996). The diehard battery of tests of randomness. `http://stat.fsu.edu/pub/diehard/`. Accessed: July 2025.

Marsaglia, G. (2003). Xorshift rngs. *Journal of Statistical software*, **8**, pp. 1–6.

Matsumoto, M. and Nishimura, T. (1998). Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, **8** (1), pp. 3–30.

McCollum, J. M., Lancaster, J. M., Bouldin, D. W., and Peterson, G. D. (2003). Hardware acceleration of pseudo-random number generation for simulation applications. In *Proceedings of the 35th Southeastern Symposium on System Theory, 2003.*, IEEE, pp. 299–303.

Mertens, S. and Bauke, H. (2004). Entropy of pseudo-random-number generators. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, **69** (5), pp. 055702.

Naik, R. B. and Singh, U. (2024). A review on applications of chaotic maps in pseudo-random number gen-

Table 3: Comparison of Statistical Randomness Metrics for Different PRNG Models

PRNG	**Entropy**	**O P**	**Chi-square**	**Arithmetic mean**	**Monte Carlo** π	**Serial correlation**
Camara et al.	7.999985	0%	244.55	127.5467	3.141160791	0.000331
Cubic Maps	7.9997	–	287.474	127.4868	3.1459	-0.0011
Ricker's P. Model	7.9998	–	263.369	127.4565	3.1495	-0.0005
S. Araki et al.	7.999994	0%	269.38	127.5005	3.142428800	-0.000216
Proposed	7.999998	0%	256.11 (46.88%)	127.4719	3.142833246	0.000071

erators and encryption. *Annals of Data Science*, **11** (1), pp. 25–50.

Olsen, L. F. and Degn, H. (1985). Chaos in biological systems. *Quarterly reviews of biophysics*, **18** (2), pp. 165–225.

Ouamri, M. (2025). Cubic trigonometric chaotic systems for high-quality pseudo-random number generation. *Cybernetics and Physics*, **14** (2), pp. 154–161.

Pareek, N., Patidar, V., and Sud, K. (2005). Cryptography using multiple one-dimensional chaotic maps. *Communications in Nonlinear Science and Numerical Simulation*, **10** (7), pp. 715–723.

Peitgen, H.-O., Jürgens, H., Saupe, D., and Feigenbaum, M. J. (2004). *Chaos and fractals: new frontiers of science*, vol. 106. Springer.

Sandri, M. (1996). Numerical calculation of lyapunov exponents. *Mathematica Journal*, **6** (3), pp. 78–84.

Schoo, M., Pawlikowski, K., and McNickle, D. C. (2005). A survey and empirical comparison of modern pseudo-random number generators for distributed stochastic simulations.

Tsonis, A. and Elsner, J. (1989). Chaos, strange attractors, and weather. *Bulletin of the American Meteorological Society*, **70** (1), pp. 14–23.

Tsonis, A. A. (2012). *Chaos: from theory to applications*. Springer Science & Business Media.

Upadhyay, R. K., Iyengar, S., and Rai, V. (1998). Chaos: an ecological reality? *International Journal of Bifurcation and Chaos*, **8** (06), pp. 1325–1333.

Walker, J. (2008). Ent: A pseudorandom number sequence test program. `https://www.fourmilab.ch/random/`. Accessed: July 2025.