Indonesian Journal of Electrical Engineering and Computer Science
Vol. 44, No. 1, October 2026, pp. 82~96
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v44.i1.pp82-96

Chaos-driven nonce traversal using a modified Tent map for proof-of-work

Assia Merzoug[1], Sabah Delenda[1], Hana Ali Pacha[2]
[1]Department of Computer Science, Institute of Science, Laboratory of Science for mathematics, Computer Science and Engineering Applications, University Center of Barika, Barika, Algeria
[2]Quartz Laboratory ECAM-EPMI, Graduate School of Electrical Engineering, 13 Bd de l'Hautil, France

Article Info

Article history:

Received Jul 27, 2026
Revised Aug 9, 2026
Accepted Sep 10, 2026

Keywords:

Blockchain mining
Chaotic dynamical systems
Finite-precision chaos
Modified Tent map
Nonce traversal
Proof-of-work
Pseudo-random sequences

ABSTRACT

Proof-of-work (PoW) mining evaluates cryptographic hash functions over different block-header candidates until a hash satisfies a predefined difficulty target. This paper proposes a non-sequential nonce traversal mechanism based on a modified one-dimensional Tent map. Rather than accelerating mining or altering the security assumptions of PoW, the proposed approach investigates whether chaotic dynamics can provide a deterministic, reproducible, and statistically dispersed ordering of 32-bit nonce candidates while remaining fully compatible with the conventional mining process. The dynamical properties of the proposed modified Tent map are analyzed through bifurcation diagrams, Lyapunov exponent estimation, and sensitivity to initial condition analysis. The generated nonce sequences are evaluated using Shannon entropy, correlation, autocorrelation, duplicate-rate, and nonce-space coverage metrics, and are compared with sequential enumeration, the classical Tent map, MT19937, and Xorshift32. Experimental results demonstrate that the proposed method generates high-entropy, weakly correlated, and well-dispersed nonce sequences while showing improved overall finite-precision behavior under the tested conditions compared with the classical Tent map. PoW simulations further confirm that the proposed traversal strategy preserves compatibility with conventional mining workflows without changing the expected probability of solving the PoW puzzle for a fixed number of distinct hash evaluations. The proposed method should therefore be regarded as an alternative deterministic nonce-ordering strategy rather than a mining-acceleration technique.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Assia Merzoug
Department of Computer Science, Institute of Science
Laboratory of Science for Mathematics, Computer Science and Engineering Applications
University Center of Barika, Barika, Algeria
Email: assia.merzoug@cu-barika.dz

1. INTRODUCTION

Blockchain technology has emerged as a major paradigm for decentralized and tamper-resistant distributed computing since the introduction of Bitcoin by Nakamoto in 2008. By combining cryptographic hash functions, digital signatures, and distributed consensus, blockchain enables untrusted participants to maintain a shared ledger without centralized authority. Beyond cryptocurrencies, it has been applied to finance, healthcare, industrial internet of things (IIoT), supply-chain management, cloud computing, digital identity, and smart contracts [1]–[5]. Among existing consensus mechanisms, PoW remains one of the most widely studied consensus mechanisms and is used in several established blockchain systems. In PoW systems

such as Bitcoin, miners repeatedly evaluate a cryptographic hash function on candidate block headers until a hash satisfying a predefined difficulty target is obtained [1], [6]–[9].

For a fixed block header, nonce generation can be viewed as a search over the finite 32-bit nonce space. Under the random-oracle assumption, each distinct nonce produces an independent hash outcome, and the probability of finding a valid PoW solution depends on the number of hash evaluations rather than on the numerical order in which nonce values are tested [6], [8], [9]. Therefore, sequential enumeration provides a natural baseline because it is computationally simple, deterministic, collision-free before wraparound, and straightforward to partition among miners. Nevertheless, alternative deterministic traversal orders can explore the same nonce space according to different sequences without changing the theoretical probability of solving the PoW problem for a fixed number of distinct hash evaluations.

Chaotic dynamical systems have attracted significant atTention for deterministic sequence generation because of their sensitivity to initial conditions, nonlinear dynamics, and mixing properties. Among one-dimensional chaotic systems, the Tent map is particularly attractive due to its simple formulation and low computational complexity. Several modified versions have been proposed to mitigate finite-precision degradation and improve digital implementations [10]–[16]. However, the use of chaotic deterministic traversal mechanisms specifically for PoW nonce search remains comparatively underexplored.

In this work, a perturbed Tent map with a modulo-1 operation is investigated as a deterministic traversal mechanism for the finite 32-bit nonce domain. The proposed formulation introduces an additive perturbation parameter into both branches of the classical Tent map, followed by a modulo-1 operation to preserve the normalized state. The objective is not to improve the cryptographic security of PoW or increase its theoretical success probability, but to investigate whether chaotic dynamics can provide a reproducible, dispersed, and computationally practical ordering of nonce candidates.

The proposed map is analyzed using bifurcation diagrams, Lyapunov exponent estimation, and sensitivity to initial condition analysis. The resulting integer nonce sequences are evaluated using Shannon entropy, Pearson correlation, autocorrelation, nonce-space coverage, unique nonce ratio, collision rate, average jump distance, and search dispersion. Their performance is compared with sequential enumeration, MT19937, Xorshift, and the classical Tent map.

The main contributions are therefore: 1. formulation and dynamical analysis of a modified one-dimensional Tent map with additive perturbation and modulo-1 operation, 2. development of a deterministic chaos-driven traversal mechanism for the 32-bit PoW nonce space, 3. systematic evaluation using statistical and nonce-specific metrics, and 4. comparative analysis of traversal behavior and computational cost, including finite-precision effects and the additional overhead associated with chaotic nonce generation.

2. PROPOSED CHAOS-DRIVEN NONCE TRAVERSAL

2.1. Classical Tent map

The Tent map is one of the most widely studied one-dimensional discrete-time chaotic systems because of its simple piecewise-linear formulation, low computational complexity, and well-established chaotic behavior. These characteristics have led to its extensive application in nonlinear dynamics, pseudo-random sequence generation, optimization, secure communications, and chaos-based cryptographic systems [10]–[16].

The classical Tent map is defined as [10]–[12]

$$x_{n+1} = \begin{cases} \mu x_n, & x_n < 0.5, \\ \mu(1 - x_n), & x_n \geq 0.5. \end{cases} \quad (1)$$

where $x_n \in [0,1]$ denotes the system state at iteration n, and μ is the control parameter governing the system dynamics.

Under infinite-precision arithmetic, the Tent map exhibits fundamental chaotic properties, including sensitivity to initial conditions, topological mixing, ergodicity, and positive Lyapunov exponents for appropriate values of μ [10]–[12], [15]. However, in practical digital implementations, finite-precision arithmetic introduces quantization effects that may degrade the long-term dynamics, leading to relatively short periodic cycles and reduced statistical quality of the generated sequences [10], [11], [13], [16]. To overcome these limitations, numerous enhanced Tent map variants have been proposed, including perturbation techniques, parameter modulation, coupled chaotic systems, hybrid maps, and modulo-based transformations, with the objective of improving finite-precision behavior and enhancing the statistical characteristics of the generated sequences [13]–[16].

Motivated by these developments, this work introduces a modified one-dimensional Tent map designed to generate a deterministic yet dynamically diversified trajectory for alternative traversal of the 32-

bit nonce space. The proposed modification aims to investigate the nonlinear and statistical properties of the resulting traversal sequence while preserving the theoretical characteristics of the underlying PoW protocol.

2.2. Proposed modified Tent map

The proposed modified Tent map is defined as:

$$x_{n+1} = \begin{cases} \mu(x_n + k) \bmod 1, & x_n < 0.5, \\ \mu(1 - x_n + k) \bmod 1, & x_n \geq 0.5. \end{cases} \quad (2)$$

where x_n denotes the system state at iteration *n*, μ is the control parameter, *k* is the perturbation parameter, and the modulo operation confines each generated state to the unit interval.

Unlike the classical Tent map, the proposed formulation introduces a controlled perturbation before the piecewise-linear transformation. The modulo operation preserves bounded evolution while maintaining the simplicity and computational efficiency of the original map. Consequently, the perturbation parameter modifies the sequence of visited states under finite-precision arithmetic without increasing the asymptotic computational complexity of the iterative process.

The proposed map is evaluated in the following sections through nonlinear dynamical analysis, statistical assessment of the generated nonce sequences, and PoW mining experiments to validate its finite-precision behavior, statistical quality, and compatibility with conventional PoW systems. Moreover, the proposed map retains the same computational complexity, *O(1)*, as the classical Tent map, introducing only constant-time arithmetic operations at each iteration, consisTent with standard algorithmic complexity analysis [17].

3. METHOD

This section describes the proposed chaos-driven nonce generation procedure, its integration into the conventional PoW mining process, the experimental setup, and the evaluation metrics used throughout this study.

3.1. Nonce generation procedure

3.1.1. Principle of the proposed traversal strategy

In conventional PoW systems, candidate nonces are typically generated through deterministic sequential enumeration until a valid hash is obtained or the 32-bit nonce space is exhausted [1]–[5]. As discussed in Section 2, under the random-oracle assumption, the traversal order does not affect the theoretical probability of finding a valid PoW solution for a fixed number of distinct nonce evaluations. Accordingly, the proposed approach does not aim to reduce the expected mining complexity but instead investigates an alternative deterministic traversal of the nonce space based on the proposed modified Tent map. The resulting chaotic states are subsequently converted into 32-bit integer nonce candidates, while the remaining mining workflow remains identical to that of conventional PoW systems.

3.1.2. Nonce quantization

The proposed modified Tent map generates real-valued states in the unit interval. Since the nonce field is represented as an unsigned 32-bit integer, each state is converted into a nonce candidate according to

$$Nonce_n = \lfloor 2^{32} x_n \rfloor \quad (3)$$

where x_n denotes the chaotic state at iteration *n*, and $\lfloor \cdot \rfloor$ is the floor operator. Consequently, the generated nonces satisfy

$$0 \leq Nonce_n < 2^{32} \quad (4)$$

This quantization establishes a direct mapping between the unit interval and the 32-bit nonce space without introducing additional scaling factors, following classical principles of integer mapping and numerical sequence generation [18].

3.2. Chaos-driven nonce traversal algorithm

The proposed traversal Algorithm 1 is summarized below.

Algorithm 1: Chaos-Driven Nonce Traversal

Input
- Initial state x_0
- Control parameter μ
- Perturbation parameter k

Output
- Sequence of candidate nonces

Step1: Initialize $x \leftarrow x_0$.
Step2: Repeat
- Compute the next chaotic state : x_{n+1}.
- Generate the nonce $Nonce_n$.
- Construct the candidate block header B_n.
- Compute the hash value $H(B_n)$.
- If $\mathrm{H}(B_n) < \mathrm{T}$, terminate.

Step3: Otherwise continue the iterative process.

The proposed algorithm modifies only the mechanism responsible for generating candidate nonce values. The block format, hash function, validation rule, and consensus protocol remain unchanged.

3.3. PoW integration

The proposed nonce-generation mechanism is fully compatible with conventional PoW systems because it modifies only the nonce generation process while preserving the cryptographic hash function, block structure, validation rule, and consensus protocol [1]–[5], [19]. During mining, each generated nonce is inserted into the candidate block header, whose cryptographic hash is then computed and compared with the network difficulty target. Accordingly, the validation criterion remains

$$H(B_{\mathrm{n}}) < T \tag{5}$$

where the hash function, block structure, and consensus protocol are identical to those of conventional PoW implementations. Consequently, the proposed approach requires no modification to the blockchain protocol.

The proposed approach modifies only the nonce-generation procedure while preserving all standard PoW operations. Consequently, it constitutes an alternative deterministic nonce-traversal strategy that remains fully compatible with existing blockchain protocols.

3.4. Experimental setup

3.4.1. Experimental environment

All experiments were conducted in MATLAB R2009a using IEEE 754 double-precision floating-point arithmetic. MATLAB R2009a was intentionally retained to ensure reproducibility with the original implementation. Unless otherwise specified, the modified Tent map was initialized with the initial state $x_0 \in (0,1)$, while the control parameter μ and perturbation parameter k remained fixed throughout each experiment. To reduce the influence of transient behavior, the initial iterations were discarded before collecting the generated trajectories. The remaining states were used for both nonlinear dynamical analyses and statistical evaluations. All reference generators were evaluated under identical experimental conditions to ensure a fair comparison.

3.4.2. Evaluation method

The proposed approach was evaluated at three complementary levels. First, the nonlinear dynamical behavior of the proposed modified Tent map was analyzed. Second, the statistical properties of the quantized 32-bit nonce sequence were evaluated. Finally, the proposed traversal mechanism was compared with conventional nonce-generation methods under identical experimental conditions. The nonlinear analysis and the statistical evaluation were performed independently to distinguish the behavior of the continuous dynamical system from that of the quantized integer nonce sequence.

3.4.3. Reference generators

The proposed nonce-traversal mechanism was compared with representative deterministic and pseudo-random generators, including sequential enumeration employed in Bitcoin [1], the classical Tent map [11], MT19937 [20], and Xorshift32 [21]. Each method generated the same number of 32-bit nonce candidates under identical experimental conditions. The comparison focused on the statistical properties of the generated sequences, traversal characteristics, and computational cost.

3.5. Evaluation metrics

To evaluate the proposed nonce-traversal mechanism, several quantitative metrics were employed to assess the statistical properties, traversal performance, and computational efficiency of the generated 32-bit nonce sequence.

3.5.1. Shannon entropy

The empirical Shannon entropy is computed as

$$H = -\sum_{i=1}^{M} p_i \log_2 p_i \quad (6)$$

where M denotes the number of histogram bins, and p_i is the empirical probability associated with the ith bin.

Entropy values approaching the theoretical maximum indicate an approximately uniform empirical distribution [22], [23].

3.5.2. Pearson correlation

The linear correlation between two successive nonce values is measured using Pearson's correlation coefficient

$$r = \frac{\sum_{i=1}^{N-1}(x_i-\bar{x})(x_{i+1}-\bar{x})}{\sqrt{\sum_{i=1}^{N-1}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{N-1}(x_{i+1}-\bar{x})^2}} \quad (7)$$

Values close to zero indicate weak empirical linear dependence between consecutive samples [24].

To assess the statistical significance of the difference in Pearson correlation between the classical Tent map and the modified Tent map, a paired Wilcoxon signed-rank test was applied to the correlation coefficients obtained from the 30 independent experimental runs. The significance level was set to α=0.05.

3.5.3. Autocorrelation

The temporal dependence of the generated sequence is evaluated through the normalized autocorrelation function :

$$\mathrm{R}(\tau) = \frac{\mathrm{E}[(\mathrm{x_n}-\mu)(\mathrm{x_{n+\tau}}-\mu)]}{\sigma^2} \quad (8)$$

where τ denotes the lag, μ is the empirical mean, and σ^2 is the empirical variance.

For a well-dispersed traversal sequence, the autocorrelation should rapidly approach zero for non-zero lags [25].

3.5.4. Chi-square goodness-of-fit test

The goodness-of-fit of the generated nonce distribution was evaluated using the Pearson Chi-square statistic, which measures the deviation between the observed and the expected uniform distributions. It is computed as

$$x^2 = \sum_{i=1}^{M} \frac{(O_i-E_i)^2}{E_i} \quad (9)$$

where M is the number of histogram bins, O_i is the observed frequency, and E_i is the expected frequency under a uniform distribution.

Lower Chi-square values indicate better agreement with the ideal uniform distribution and therefore better distribution quality of the generated nonce sequences [24].

3.5.5. Unique nonce ratio

The Unique Nonce Ratio is defined as

$$UNR = \frac{U}{N} \quad (10)$$

where N is the number of generated nonce candidates, and U is the number of distinct candidates.

3.5.6. Duplicate ratio

The duplicate ratio is defined by

$$\mathrm{DR} = 1 - \frac{\mathrm{U}}{\mathrm{N}} \tag{11}$$

This metric quantifies redundant nonce generations. A value close to zero indicates that most generated nonces are distinct.

3.5.7. Domain Coverage

The explored fraction of the entire 32-bit nonce space is

$$\mathrm{DC} = \frac{\mathrm{U}}{2^{32}} \tag{12}$$

Unlike the uniqueness ratio, this quantity measures the proportion of the entire search domain effectively visited.

3.5.8. Execution Time

The average nonce-generation time is measured as

$$\mathrm{T_{nonce}} = \frac{\mathrm{T_{total}}}{\mathrm{N}} \tag{13}$$

where T_{total} is the overall execution time, and *N* is the number of generated nonce candidates.

These metrics provide a comprehensive evaluation of the statistical quality and traversal performance of the proposed nonce-generation mechanism.

4. RESULTS AND DISCUSSION

This section presents the experimental results obtained with the proposed chaos-driven nonce-traversal mechanism. The evaluation is organized into four parts. First, the nonlinear dynamical behavior of the proposed modified Tent map is analyzed. Second, the statistical properties of the generated nonce sequence are investigated. Third, the traversal performance is evaluated using the proposed metrics. Finally, the proposed approach is compared with representative nonce-generation methods.

4.1. Nonlinear dynamic analysis

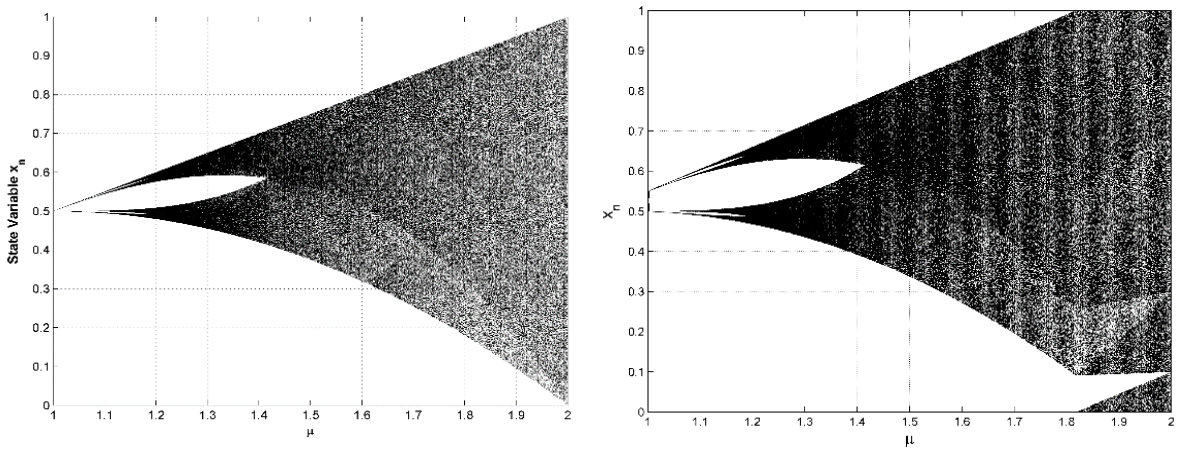
4.1.1. Bifurcation analysis

Bifurcation diagrams are widely used to investigate the qualitative evolution of nonlinear dynamical systems as a control parameter varies, allowing transitions between stable and chaotic behaviors to be identified [10]–[12]. Figure 1 presents the bifurcation diagrams of the classical Tent map and the proposed modified Tent map for $1 \leq \mu \leq 2$. The simulations were performed with $x_0 = 0.321$, *k=0.05*, *1500 iterations*, and *750* discarded transient iterations. As shown in Figure 1(a), the classical Tent map exhibits the expected transition from stable behavior to chaos as the control parameter increases. The trajectories become densely distributed over the state space for larger values of μ\muμ, indicating the chaotic regime.

A comparison of Figures 1(a) and 1(b) shows that the proposed modified Tent map preserves the overall bifurcation structure of the classical map while modifying the distribution of trajectories under finite-precision arithmetic. Although the perturbation parameter changes the organization of the chaotic branches, the map maintains a broad chaotic region over the investigated parameter range, confirming that the essential nonlinear characteristics of the original Tent map are preserved. These observations indicate that the proposed modification influences finite-precision trajectory evolution without suppressing the underlying chaotic dynamics. A quantitative assessment is provided in the following sections through the largest lyapunov exponent (LEE), sensitivity-to-initial-condition analysis, and statistical evaluation of the generated nonce sequence.

4.1.2. Largest lyapunov exponent

The LLE is a fundamental indicator for characterizing chaotic systems as shown in Figure 2. A positive Lyapunov exponent indicates that two trajectories originating from infinitesimally close initial conditions diverge exponentially over time, thereby confirming the presence of chaotic dynamics [26], [27].



(a) (b)

Figure 1. Bifurcation diagrams of (a) the classical Tent map and (b) the proposed modified Tent map

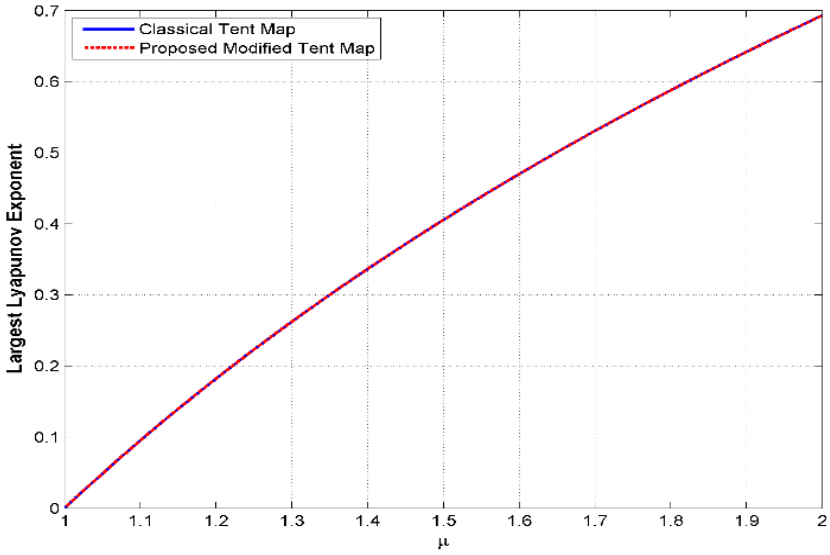


Figure 2. Compares the LLEs of the classical Tent map and the proposed modified Tent map

Tent map over the interval $1 \leq \mu \leq 2$. As observed, the two curves almost completely overlap throughout the investigated parameter range, and both remain positive in the chaotic region. This behavior is expected because the proposed modification preserves the local slope of the classical Tent map. For both maps, the absolute value of the derivative satisfies.

$$| f'(x) | = \mu \tag{14}$$

Except at the discontinuity points. Consequently, the theoretical LLE is approximately

$$\lambda \approx \ln(\mu) \tag{15}$$

which explains the nearly identical evolution of the two curves.

The results indicate that the introduction of the perturbation parameter k does not alter the local exponential divergence of neighboring trajectories. Instead, the proposed modification preserves the intrinsic chaotic behavior of the classical Tent map while influencing the finite-precision evolution of the generated trajectories. Therefore, both maps exhibit comparable chaotic intensity over the investigated parameter range.

4.1.3. Sensitivity to initial conditions

Sensitivity to initial conditions is one of the fundamental characteristics of chaotic systems, describing the exponential divergence of trajectories originating from nearly identical initial states [26], [27]. To evaluate the finite-precision behavior of the proposed modified Tent map, two trajectories were generated using the initial conditions x_0 and $x_0+\delta$, where $\delta=10^{-12}$. The evolution of the separation between the trajectories is illustrated in Figure 3.

As shown in Figure 3(a), the trajectories generated by the classical Tent map eventually collapse to the same state because of finite-precision arithmetic. Once this convergence occurs, the exponential divergence is lost, indicating numerical degradation of the chaotic dynamics. In contrast, Figure 3(b) shows that the proposed modified Tent map maintains the separation between neighboring trajectories throughout the simulation. The perturbation parameter prevents premature convergence, thereby improving the numerical robustness of the map under finite-precision implementation.

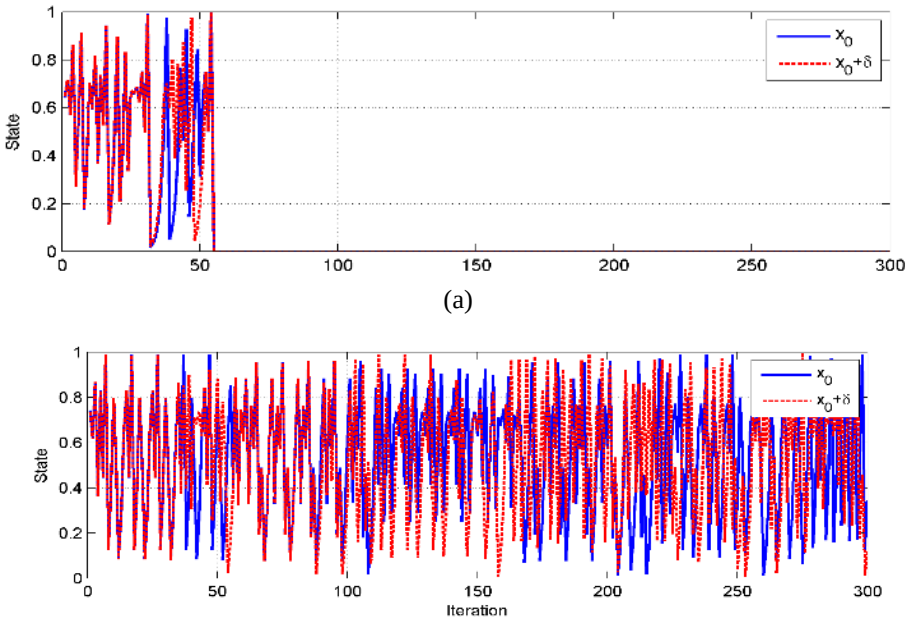


Figure 3. Sensitivity to initial conditions of (a) the classical Tent map and (b) the proposed modified Tent map under finite-precision arithmetic

It should be emphasized that this observation does not indicate a larger LLE, since both maps exhibit nearly identical Lyapunov exponents. Instead, it demonstrates that the proposed modification enhances the preservation of chaotic trajectories in finite-precision arithmetic, which is advantageous for applications requiring long deterministic sequences, such as pseudo-random number generation and PoW nonce traversal.

4.1.4. Finite-precision behavior

Finite-precision arithmetic may significantly affect the long-term evolution of chaotic maps by introducing periodic cycles and reducing the effective state space. To evaluate the numerical robustness of the proposed modified Tent map, the average cycle length and the occurrence of short periodic cycles were measured under both single- and double-precision floating-point arithmetic [10]–[16]. The corresponding results are summarized in Table 1.

Table 1. Numerical robustness comparison under finite-precision arithmetic

Map	Precision	Mean cycle length	Minimum	Maximum	Short-cycle rate (%)
Classical Tent map	Single	3069.01	50	11850	25.00
Proposed modified Tent map	Single	3838.77	1	10636	13.00
Classical Tent map	Double	23042.79	484	49624	3.00
Proposed modified Tent map	Double	24447.79	187	49317	3.00

Under single precision, the modified Tent map increases the mean cycle length from 3069.01 to 3838.77 iterations and reduces the short-cycle rate from 25% to 13%. Under double precision, both maps exhibit substantially longer cycles and the same short-cycle rate of 3%. Although the Modified Tent Map achieves a higher mean cycle length, its minimum cycle length is lower (187 versus 484 iterations), indicating that the improvement is not uniform across all cycle statistics. Thus, the modification improves overall finite-precision behavior under the tested conditions but does not eliminate short periodic cycles. A parameter sensitivity analysis was also conducted by varying μ and k around the selected operating point. The results are summarized in Table 2.

Table 2. Finite-precision sensitivity of the modified Tent map to the control and perturbation parameters

Parameter	Value	Mean cycle	Min cycle	Max cycle	Short rate
M	1.90	5150.00	200	9245	5.00
M	1.95	5259.70	1351	9133	0.00
M	1.97	4779.45	151	9713	5.00
M	1.99	4652.10	229	9208	5.00
M	0.00	4667.10	89	9869	5.00
Parameter	Value	Mean cycle	Min cycle	Max cycle	Short rate
K	0.01	4534.85	798	9587	5.00
K	0.03	5419.25	529	9886	15.00
K	0.05	4652.10	229	9208	5.00
K	0.07	3815.05	334	9627	20.00
K	0.10	3574.85	1	7345	20.00

The results confirm that finite-precision behavior depends on both parameters. The selected values, μ=1.99 and k=0.05, provide a reasonable compromise between mean cycle length and short-cycle rate and were therefore retained for the main experiments rather than considered globally optimal.

Overall, the observed improvement should primarily be interpreted as a numerical and dynamical property of the proposed map rather than as a direct PoW performance advantage. Since the continuous state is ultimately quantized into a 32-bit integer nonce, the improved single-precision robustness does not by itself imply higher mining efficiency or probability of solving the PoW puzzle. Its practical significance lies in supporting longer and more stable deterministic trajectories in digital implementations.

4.2. Statistical evaluation

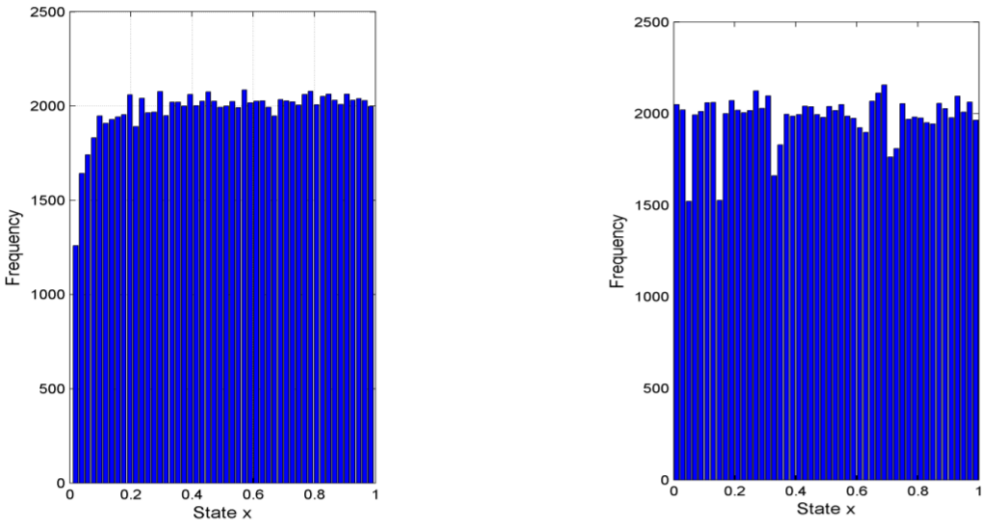
4.2.1. State distribution analysis

The empirical distribution of the generated states provides a useful indication of the statistical behavior of a chaotic system. For applications such as pseudo-random sequence generation and nonce traversal, an approximately uniform distribution over the state space is generally desirable because it promotes a balanced exploration of the available domain [22], [23].

As illustrated in Figure 4, both the classical Tent map and the proposed modified Tent map generate nearly uniform state distributions over the interval [0,1] as shown in Figure 4(a) and 4(b). No significant concentration of states is observed in particular regions of the state space, indicating that both maps preserve a satisfactory level of statistical uniformity. A visual comparison further shows that the proposed modified Tent map exhibits a distribution very similar to that of the classical Tent map. Although slight local fluctuations are present because of the introduced perturbation and finite-precision arithmetic, these variations remain limited and do not noticeably affect the overall distribution of the generated states. These results indicate that the proposed modification preserves one of the fundamental statistical properties of the classical Tent map while introducing an alternative deterministic trajectory. Consequently, the modified map remains suitable for generating uniformly distributed state sequences prior to their conversion into 32-bit nonce candidates.

4.2.2. Shannon entropy

Shannon entropy is widely used to quantify the randomness and uncertainty of a sequence. Entropy values approaching the theoretical maximum indicate a nearly uniform distribution and a high degree of statistical randomness [22], [23]. As shown in Table 3, both the classical Tent map and the proposed modified Tent map produce entropy values that are very close to the theoretical maximum of 8 bits. The corresponding normalized entropy values (0.9994 and 0.9989) indicate that the generated state sequences exhibit an almost ideal statistical distribution.



(a) (b)

Figure 4. Histogram comparison of the state distributions generated by (a) the classical Tent map and (b) the proposed modified Tent map

Table 3. Shannon entropy comparison of the classical and proposed modified Tent maps

Map	Raw entropy	Maximum entropy	Normalized entropy
Classical Tent map	7.9949	8.0000	0.9994
Proposed modified Tent map	7.9909	8.0000	0.9989

A comparison of the two maps reveals only a negligible difference in entropy. Although the proposed modification introduces a perturbation into the system dynamics, it does not significantly affect the randomness of the generated state sequence. The obtained results therefore demonstrate that the proposed modified Tent map preserves the high entropy characteristics of the classical Tent map while providing the improved finite-precision behavior observed in the previous section.

4.2.3. Statistical analysis of quantized nonce sequences

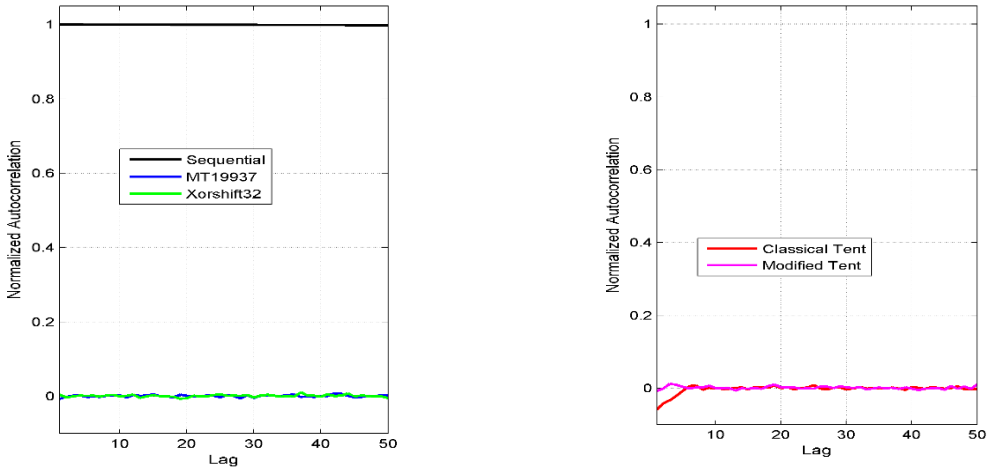
In Figure 5, the statistical properties of the generated 32-bit nonce sequences were evaluated using five representative generators: Sequential Counter, MT19937, Xorshift32, the classical Tent map, and the proposed modified Tent map. Each experiment generated 100,000 nonce candidates and was repeated over 30 independent runs using different initial conditions or generator seeds, as applicable. The chaotic generators were evaluated with μ=1.99, while k=0.05 was used for the Modified Tent map. The reported results correspond to the empirical mean ± standard deviation. The evaluated metrics include Shannon entropy, Pearson correlation, Chi-square statistic, and execution time [22]–[24]. The results are summarized in Table 4, while the normalized autocorrelation functions are shown in Figures 5(a) and 5(b).

Table 4. Statistical analysis of quantized nonce generators (30 Independent Runs)

Generator	Shannon entropy (bits)	Pearson correlation	Chi-square statistic	Execution time (s)
Sequential Counter	8.0000 ± 0.0000	1.00000 ± 0.00000	0.15 ± 0.00	0.0218 ± 0.0144
MT19937	7.9982 ± 0.0001	0.00045 ± 0.00377	249.35 ± 17.34	0.0278 ± 0.0101
Xorshift32	7.9981 ± 0.0002	−0.00123 ± 0.00354	257.30 ± 23.79	0.1922 ± 0.0229
Classical Tent map	7.9982 ± 0.0001	−0.06155 ± 0.00240	247.92 ± 13.45	0.0292 ± 0.0081
Proposed modified Tent map	7.9982 ± 0.0002	−0.00818 ± 0.00314	251.09 ± 22.98	0.0612 ± 0.0091

As shown in Table 4, all pseudo-random and chaotic generators achieve Shannon entropy values close to the theoretical maximum of 8 bits, indicating a high degree of statistical uniformity. The Sequential Counter exhibits a Pearson correlation of 1.00000 ± 0.00000, as expected from its deterministic consecutive-integer traversal, which produces a perfect linear dependence between successive nonce values.The Modified Tent map produces a Pearson correlation of −0.00818±0.00314, substantially closer to zero than the classical Tent map (−0.06155±0.00240). A paired Wilcoxon signed-rank test over the 30 runs yielded W=0 and

p=1.7344×10−6, confirming a statistically significant difference at the 5% level. Thus, the lower correlation magnitude of the Modified Tent map is statistically supported.



(a) (b)

Figure 5. Statistical analysis of quantized nonce generators (30 Independent Runs) (a) conventional nonce generators and (b) chaos-based nonce generators

The Chi-square values indicate satisfactory distribution uniformity, while the execution-time results show the additional computational cost associated with the chaotic generators. The normalized autocorrelation functions in Figure 5 rapidly decay toward zero, indicating weak temporal dependence between successive nonce values [25].

A parameter sensitivity analysis was additionally performed over the selected μ and k ranges using 30 independent runs per configuration. Shannon entropy remained close to 8 bits throughout the investigated ranges, whereas Pearson correlation showed greater parameter dependence. Its magnitude decreased from 0.10673 at μ=1.90 to 0.00703 at μ=1.99, while low magnitudes were obtained for k=0.03 (0.00617) and k=0.05 (0.00795). Accordingly, μ=1.99 and k=0.05 were retained as representative operating parameters, providing near-maximum entropy and low correlation while maintaining acceptable finite-precision behavior.

4.3. Performance evaluation of the proposed nonce traversal strategy

To evaluate the proposed chaos-driven nonce traversal strategy, six traversal metrics were computed for five nonce generation methods over 30 independent runs, each producing 100,000 candidate nonces. The evaluated metrics include the Unique Nonce Ratio (UNR), Duplicate Ratio (DR), Domain Coverage (DC), Collision Count (CC), Expected Collision Count (ECC), and the average Time per Nonce (TPN). The corresponding results are summarized in Table 5.

Table 5. Comparison of nonce traversal metrics (30 independent runs)

Generator	UNR	DR	DC	CC	ECC	TPN
Sequential Counter	1.000000 ± 0.000000	0.000000 ± 0.000000	2.328306×10^{-5} ± 0.000000×10^{-5}	0.00 ± 0.00	1.16 ± 0.00	0.075 ± 0.095
MT19937	0.999989 ± 0.000010	0.000011 ± 0.000010	2.328281×10^{-5} ± 2.32×10^{-10}	1.10 ± 0.99	1.16 ± 0.00	0.172 ± 0.052
Xorshift32	1.000000 ± 0.000000	0.000000 ± 0.000000	2.328306×10^{-5} ± 0.000000×10^{-5}	0.00 ± 0.00	1.16 ± 0.00	1.788 ± 0.046
Classical Tent map	0.999981 ± 0.000028	0.000019 ± 0.000028	2.328261×10^{-5} ± 6.61×10^{-10}	1.93 ± 2.84	1.16 ± 0.00	0.187 ± 0.034
Proposed modified Tent map	0.999986 ± 0.000020	0.000014 ± 0.000020	2.328275×10^{-5} ± 4.64×10^{-10}	1.37 ± 1.99	1.16 ± 0.00	0.519 ± 0.043

As shown in Table 5, all generators achieve a UNR close to one and an extremely low DR, indicating that almost all generated nonces are unique. The proposed modified Tent map attains a UNR of 0.999986 ± 0.000020, which is comparable to the reference generators. The measured DC is approximately

2.328306×10^{-5}, corresponding to only about 0.00233% of the full 32-bit nonce space (2^{32} possible values), as expected for 100 000 generated candidates.

However, the proposed method introduces a higher nonce-generation cost, with an average TPN of 0.519 ± 0.043 μs, compared with 0.187 ± 0.034 μs for the classical Tent map, 0.172 ± 0.052 μs for MT19937, and 0.075 ± 0.095 μs for the Sequential Counter. Thus, the modified Tent map requires approximately 2.8 times the generation time of the classical Tent map and nearly 7 times that of sequential enumeration. This additional cost results from the arithmetic operations and state quantization required by the modified chaotic recurrence. Therefore, the proposed method should be regarded as an alternative deterministic traversal mechanism with favorable statistical characteristics, rather than as a computationally faster nonce-generation method.

In the PoW simulation, the Generation Overhead is defined as the percentage of the total mining time spent generating nonce candidates, i.e.

$$Generation\ Overhead\ = 100 \times \frac{\mathrm{T}_{generation}}{\mathrm{T}_{\mathrm{total}}} \tag{16}$$

It therefore differs from TPN, which measures only the time required to generate an individual nonce. The relatively low generation overhead observed in the PoW experiments results from the fact that the total execution time is dominated by SHA-256 hashing. Consequently, the higher TPN of the modified Tent map does not translate into a proportional increase in total PoW execution time.

4.4. PoW simulation

A PoW simulation was conducted under identical conditions for all generators using a difficulty of 8 leading zero bits. For each 32-bit nonce *n*, a 76-byte block header was constructed in the order Version (4 bytes), Previous Hash (32 bytes), Merkle Root (32 bytes), Timestamp (4 bytes), Bits (4 bytes), and Nonce (4 bytes). Version, Timestamp, and Bits were fixed, while Previous Hash and Merkle Root were generated from the run identifier (*RunID*) using SHA-256. The nonce was encoded as a 32-bit unsigned integer and the resulting header was processed using double SHA-256:

$$H(B_n) = SHA256(SHA256(B_n)) \tag{17}$$

A nonce was considered valid when the first 8 bits of the 256-bit hash were zero, corresponding to a success probability of 1/256 and a theoretical mean of 256 hash evaluations. The simulation comprised 100 independent runs with a maximum of 1,000,000 candidate nonces per run. The results are summarized in Table 6.

Table 6 shows that all evaluated generators require approximately the same number of hash evaluations before discovering a valid nonce, with average values close to the theoretical expectation of 256. The hash rates are also similar (approximately 1.38–1.43 kH/s), while nonce-generation overhead remains below 2.2% for all generators. No duplicate nonces were observed during the 100 PoW simulation runs under the tested conditions.

To verify whether the observed differences in mining performance are statistically significant, the Friedman nonparametric test was selected because it compares multiple related samples without assuming normality of the underlying distributions [28]. The results are reported in Table 7.

Table 6. PoW simulation results (Difficulty = 8 Leading Zero Bits, 100 Independent Runs)

Generator	Hashes until success	Mining time (s)	Hash rate (H/s)	Duplicate count	Generation overhead (%)
Theoretical (2^8)	256	–	–	–	–
Sequential	245.36 ± 216.94	0.1712 ± 0.1512	1428.40 ± 35.37	0.00 ± 0.00	1.49 ± 0.16
MT19937	272.80 ± 264.23	0.1917 ± 0.1846	1403.03 ± 112.98	0.00 ± 0.00	1.63 ± 0.33
Xorshift32	276.76 ± 292.50	0.1968 ± 0.2046	1378.82 ± 114.33	0.00 ± 0.00	2.19 ± 0.94
Classical Tent map	224.11 ± 242.14	0.1564 ± 0.1677	1417.31 ± 48.43	0.00 ± 0.00	1.78 ± 0.26
Proposed modified Tent map	268.43 ± 269.87	0.1895 ± 0.1893	1394.00 ± 72.00	0.00 ± 0.00	1.65 ± 0.48

Table 7. Statistical validation of hashes until success

Metric	Statistical test	p-value	Conclusion
Hashes until success	Friedman test	0.7824	No statistically significant difference among the five nonce generation strategies

All generators required approximately the same number of hash evaluations, with average values close to the theoretical expectation. Hash rates were also similar, while nonce-generation overhead remained below 2.2%. No duplicate nonces were observed during the simulations. To assess statistical differences among the five strategies, the Friedman nonparametric test was applied [28]. The results are reported in Table 7. The obtained p-value of 0.7824 indicates no statistically significant difference. This is consisTent with the random-oracle analysis in Section 2, under which the traversal order does not affect the probability of finding a valid solution for a fixed number of distinct hash evaluations.

The simulation used a deliberately low difficulty of 8 leading zero bits to enable repeated controlled experiments. Therefore, the results demonstrate statistical equivalence of the traversal strategies rather than performance at realistic blockchain difficulty levels. Evaluation under large-scale and hardware-based mining conditions remains future work.

4.5. Comparative discussion

The experimental results demonstrate that the proposed modified Tent map successfully combines the desirable properties of chaotic dynamics with the practical requirements of PoW nonce generation. The nonlinear analyses confirmed that the proposed modification preserves the bifurcation structure and positive LLE of the classical Tent map while improving its robustness against finite-precision degradation. These characteristics make the proposed modified Tent map demonstrate favorable finite-precision characteristics under the tested conditions. The statistical evaluation demonstrated that the generated nonce sequence exhibits high Shannon entropy, low correlation, negligible duplicate generation, and an almost uniform distribution over the 32-bit nonce space. The proposed method also achieves a UNR close to unity, a CC consisTent with theoretical expectations, and computational performance comparable to conventional generators.

The comparison also highlights the different characteristics of the evaluated generators. Sequential enumeration guarantees exhaustive traversal with the lowest computational overhead but follows a predictable search order. MT19937 and Xorshift32 provide strong statistical properties through pseudo-random generation. In contrast, both the classical Tent map and the proposed modified Tent map generate deterministic yet non-sequential traversal sequences based on chaotic dynamics. Compared with the classical Tent map, the proposed modification improves numerical robustness while preserving the statistical quality of the generated nonces.

Regarding parallel deployment, sequential enumeration can be directly partitioned into non-overlapping nonce ranges, whereas the proposed chaos-driven traversal would require an explicit mechanism to assign non-overlapping trajectory segments to parallel mining units. Different initial conditions alone do not guarantee collision-free nonce subsets after quantization. Therefore, trajectory partitioning and hardware implementation for GPU, FPGA, or ASIC architectures remain open issues for future work.

Overall, the proposed modified Tent map should not be interpreted as a mechanism for accelerating blockchain mining, since the expected number of hash evaluations depends solely on the mining difficulty. Instead, its contribution lies in providing an alternative deterministic chaos-driven nonce traversal strategy that preserves the probabilistic behavior of the PoW process while exhibiting favorable nonlinear and statistical properties, while incurring additional nonce-generation cost. The proposed approach therefore represents an alternative deterministic nonce traversal strategy for PoW systems, while its large-scale parallel deployment remains a subject for future investigation.

5. CONCLUSION

This paper proposed a modified Tent map as a deterministic chaos-driven mechanism for generating an alternative traversal of the 32-bit nonce space in PoW systems. Rather than improving mining efficiency, the proposed approach investigates whether nonlinear deterministic dynamics can provide a statistically diversified and reproducible nonce traversal while remaining fully compatible with the conventional PoW mining process. Experimental results demonstrate that the proposed modification preserves the chaotic characteristics of the classical Tent map while improving its robustness under finite-precision arithmetic. After quantization, the generated nonce sequences exhibit high Shannon entropy, low correlation, negligible duplicate generation, and an approximately uniform distribution. Furthermore, the PoW simulations and the Friedman statistical test support the conclusion that the evaluated traversal strategies do not significantly affect the number of hash evaluations required to discover a valid block, consisTent with the theoretical analysis presented in this work.

Therefore, the proposed method should not be regarded as a mining acceleration technique, but rather as an alternative deterministic non-sequential nonce traversal strategy that preserves the probabilistic behavior of PoW while exhibiting favorable nonlinear, statistical, and computational properties. Future work will investigate collision-free chaotic traversal mechanisms, multidimensional chaotic systems, adaptive

perturbation strategies, and a deeper theoretical analysis of finite-precision dynamics. In addition, hardware implementations on GPU, FPGA, and ASIC platforms, as well as integration into practical blockchain systems, will be explored.

REFERENCES

[1] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," 2019, doi: 10.2139/ssrn.3440802.
[2] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, "Bitcoin and cryptocurrency technologies: a comprehensive introduction," *and S. Goldfeder, Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton*, 2016.
[3] D. Yaga, P. Mell, N. Roby, and K. Scarfone, *Blockchain technology overview*. 2018.
[4] A. M. Antonopoulos, "Mastering bitcoin: programming the open blockchain," *3rd ed. Sebastopol, CA, USA: O'Reilly Media*, 2023.
[5] I. Bashir, "Mastering Blockchain," *4th ed. Birmingham, U.K.: Packt Publishing*, 2023.
[6] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. CRC Press, 2018.
[7] D. R. Stinson and M. B. Paterson, "Cryptography: theory and practice," *4th ed. Boca Raton, FL, USA: CRC Press*, 2018.
[8] J. Katz and Y. Lindell, "Introduction to modern cryptography," *3rd ed. Boca Raton, FL, USA: CRC Press*, 2020.
[9] N. I. of S. and T. (NIST), "Secure hash standard (SHS)," *Federal Information Processing Standards Publication (FIPS PUB) 180-4, Gaithersburg, MD, USA*, 2015.
[10] S. H. Strogatz, *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. Chapman and Hall/CRC, 2024.
[11] K. T. Alligood, T. D. Sauer, and J. A. Yorke, *Chaos: an introduction to dynamical systems*. Springer New York, 1996.
[12] D. R. J. Chillingworth, "An introduction to chaotic dynamical systems (2nd edition)," *The Mathematical Gazette*, vol. 74, no. 468, pp. 191–191, 1990, doi: 10.2307/3619398.
[13] L. Kocarev, "Chaos-based cryptography: a brief overview," *IEEE Circuits and Systems Magazine*, vol. 1, no. 3, pp. 6–21, 2001, doi: 10.1109/7384.963463.
[14] G. Chen and T. Ueta, *Chaos in circuits and systems*. WORLD SCIENTIFIC, 2002.
[15] J. C. Sprott, *Chaos and time-series analysis*. Oxford University PressOxford, 2003.
[16] G. Chen and X. Liao, "From chaos to chaos-based cryptography," *Singapore: Springer*, 2022.
[17] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms," *4th ed. Cambridge, MA, USA: MIT Press*, 2022.
[18] J. W. Wrench, "Table errata: The art of computer programming, Vol. 2: Seminumerical algorithms (Addison-Wesley, Reading, Mass., 1969) by Donald E. Knuth," *Mathematics of Computation*, vol. 24, no. 110, p. 504, 1970, doi: 10.1090/s0025-5718-1970-0400642-2.
[19] W. Stallings, "*Cryptography and Network Security: Principles and Practice*: 6th ed. Upper Saddle River, NJ: Prentice Hall, 2013, 752p., $142.40. ISBN: 13: 978-0133354690.," *Information Security Journal: A Global Perspective*, vol. 23, no. 1–2, pp. 49–50, Jan. 2014, doi: 10.1080/19393555.2014.900834.
[20] M. Matsumoto and T. Nishimura, "Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Transactions on Modeling and Computer Simulation*, vol. 8, no. 1, pp. 3–30, Jan. 1998, doi: 10.1145/272991.272995.
[21] G. Marsaglia, "Xorshift RNGs," *Journal of Statistical Software*, vol. 8, no. 14, 2003, doi: 10.18637/jss.v008.i14.
[22] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948, doi: 10.1002/j.1538-7305.1948.tb01338.x.
[23] T. M. Cover and J. A. Thomas, "Elements of information theory," *2nd ed. Hoboken, NJ, USA: John Wiley & Sons*, 2006.
[24] J. Snell, D. C. Montgomery, and G. C. Runger, "Applied statistics and probability for engineers," *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, vol. 158, no. 2, p. 355, 1995, doi: 10.2307/2983314.
[25] A. V. Oppenheim and R. W. Schafer, "Discrete-time signal processing," *3rd ed. Upper Saddle River, NJ, USA: Pearson*, 2010.
[26] A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, "Determining lyapunov exponents from a time series," *Physica D: Nonlinear Phenomena*, vol. 16, no. 3, pp. 285–317, 1985, doi: 10.1016/0167-2789(85)90011-9.
[27] E. Ott, "Chaos in dynamical systems," *2nd ed. Cambridge, U.K.: Cambridge University Press*, 2002.
[28] J. D. Gibbons and S. Chakraborti, "Nonparametric statistical inference," *5th ed. Boca Raton, FL, USA: CRC Press*, 2011.

BIOGRAPHIES OF AUTHORS



Assia Merzoug is a Lecturer and Researcher at the University Center of Barika, Algeria. She received the Ph.D. degree in Cryptography and Information Security from Mostefa Ben Boulaïd University of Batna 2, Algeria. Her research interests include cryptography, chaotic systems, blockchain technology, cybersecurity, artificial intelligence, and secure distributed systems. Her current research focuses on the design of secure cryptographic mechanisms and the application of nonlinear chaotic dynamics to blockchain security, Proof-of-Work consensus protocols, and pseudo-random number generation. She is also interested in the integration of artificial intelligence techniques into cybersecurity applications. She can be contacted at email: assia.merzoug@cu-barika.dz.



Sabah Delenda was born in Batna, Algeria. She is a Lecturer and Researcher at the University Center of Barika, Algeria. She received the Ph.D. degree in Computer Science in 2018 from Mostefa Ben Boulaïd University of Batna 2, Algeria. Her research interests include computer science, artificial intelligence, cryptography, information security, image analysis, and intelligent computing systems. Her current research focuses on the development of artificial intelligence techniques for secure computing, image processing, cryptographic applications, and information security. She can be contacted at email: sabah.delenda@cu-barika.dz.



Hana Ali Pacha is a lecturer and researcher at Quartz Laboratory ECAM-EPMI. She holds a Ph.D. in Telecommunications, specializing in cryptography and data security, from the University of Science and Technology of Oran Mohamed-Boudiaf (USTO-MB). Her research focuses on developing new encryption systems incorporating confusion and diffusion techniques to enhance the security of digital infrastructures. She is particularly interested in information system security, blockchain, and data traceability, as well as the integration of cryptographic systems into distributed architectures and cloud computing. She also contributes to collaborative research projects aimed at improving secret sharing protocols and enhancing system resilience against cyber threats. She can be contacted at email: alipachana@gmail.com.